

# An Introduction to Cloud Computing

Marty Kandes, Ph.D.

HPC User Services Group  
San Diego Supercomputer Center  
University of California, San Diego

Student Cluster Competition Training Session  
Friday, April 26th, 2019  
1:00PM - 3:00PM PT

## About Me

- ▶ B.S. in Physics + Applied Mathematics (Michigan); undergraduate research: experimental physics
- ▶ M.S. in Physics, Ph.D. Computational Science (SDSU); graduate research: computational/theoretical physics
- ▶ Started working in Data Center Operations Group @ SDSC while finishing Ph.D.
- ▶ Previously worked for Distributed High-Throughput Computing Group @ SDSC and the Open Science Grid
- ▶ Joined HPC User Services Group @ SDSC in April 2017

# An Overview of Today's Training Session

- ▶ What is Cloud Computing?
- ▶ An Overview of OpenStack + Jetstream
- ▶ Hands-On: Build a compute cluster in the cloud!

# What is Cloud Computing?



# The Definition of Cloud Computing

**Cloud computing** is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

- *NIST Special Publication 800-145*

# The Essential Characteristics of Cloud Computing

- ▶ *On-demand, self-service*: A user can unilaterally provision computing capabilities without the need for human interaction with the service provider.
- ▶ *Broad network access*: Computing capabilities are available over a network and easily accessible.
- ▶ *Resource pooling*: Computing resources are pooled together to serve multiple users, typically within a multi-tenant model.
- ▶ *Rapid elasticity*: Computing capabilities can be elastically provisioned and released, in some cases automatically, to scale up and out or down and in with resource demand.
- ▶ *Measured service*: Resource usage is monitored, controlled, and reported via some form of a metering capability, providing transparency for both the provider and user of the service.

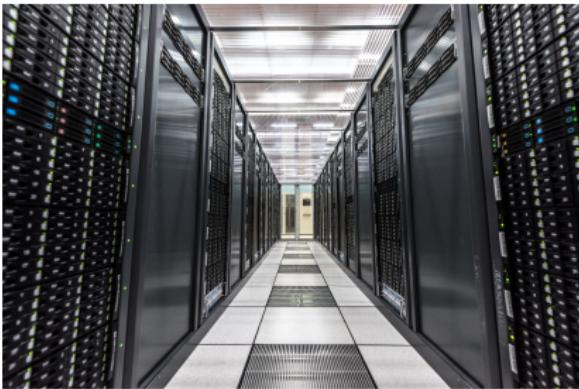
This is cloud computing, right?



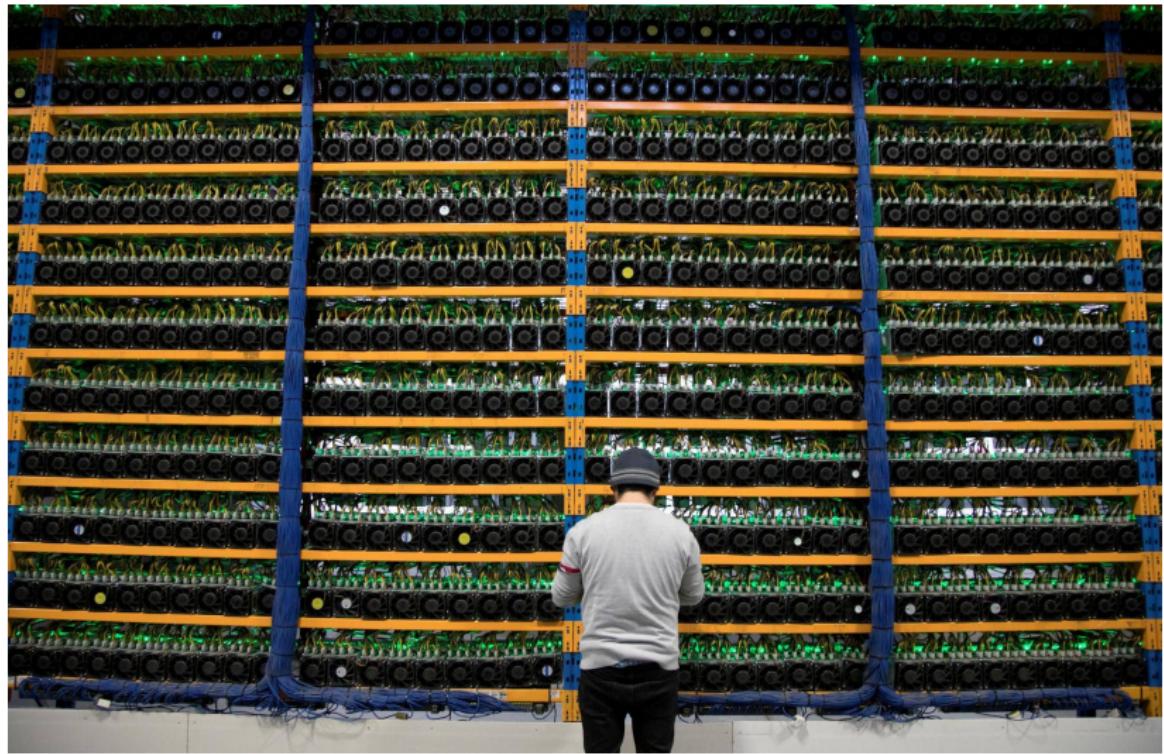
What about this? Is this cloud computing?



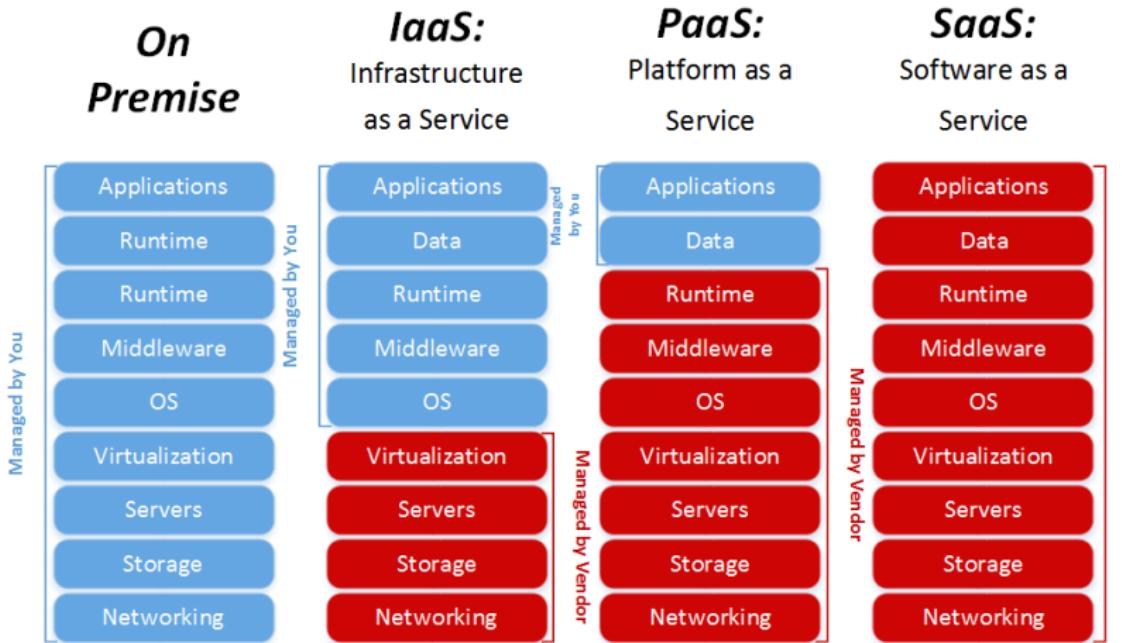
# Is this not cloud computing?



# How about this?



# Service Models for Cloud Computing



# Infrastructure as a Service

**Infrastructure as a Service (IaaS)** provides the capability to provision processing, storage, networks, and other fundamental computing resources where the user is able to deploy and run arbitrary software, which can include operating systems and applications.

The user does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

# Platform as a Service

**Platform as a Service (PaaS)** provides the capability to deploy applications created using programming languages, libraries, services, and tools supported by the provider's cloud infrastructure.

The user does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

# Software as a Service

**Software as a Service (SaaS)** provides the capability to access applications running on a service provider's cloud infrastructure, which are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface.

The user does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

# Deployment Models for Cloud Computing

- ▶ *Private cloud*: Exclusive use by a single organization.
- ▶ *Community cloud*: Exclusive use by a specific community of users from a variety of organizations with a shared mission.
- ▶ *Public cloud*: Open to use by the general public.
- ▶ *Hybrid cloud*: A composition of two or more distinct cloud infrastructures (private, community, or public).

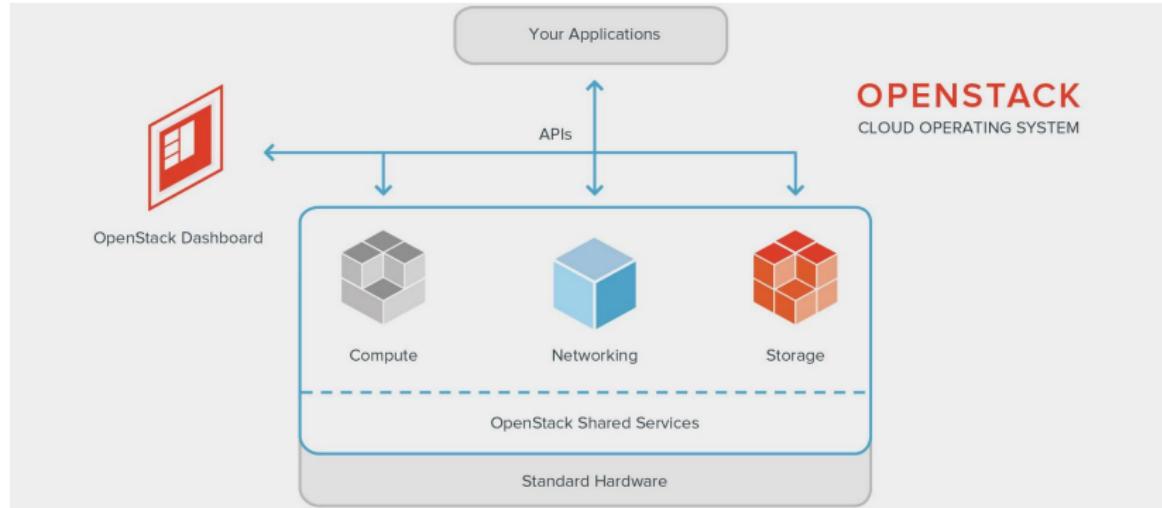
# An Overview of OpenStack + Jetstream



openstack®

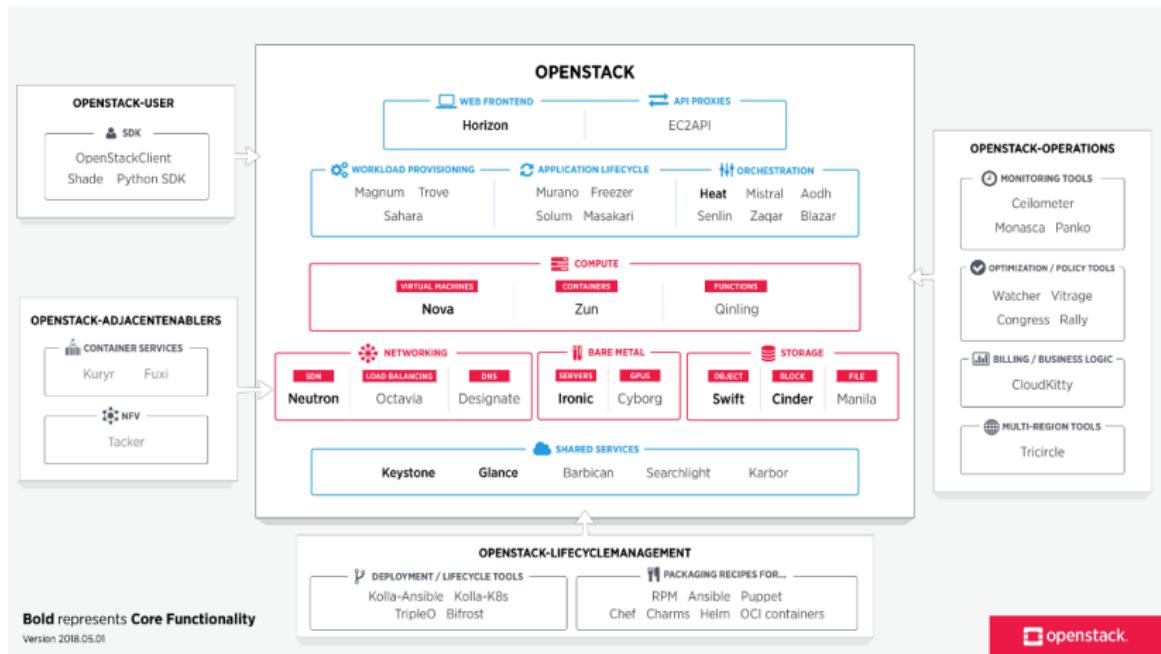
*Jetstream*

# What is OpenStack?



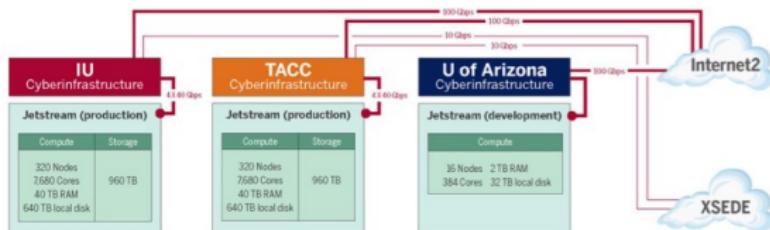
**OpenStack** is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed and provisioned through APIs with common authentication mechanisms.

# The OpenStack Ecosystem



# What is Jetstream?

## Jetstream System Overview



- Geographically distributed cloud; 0.5 PetaFLOPS
- Globus for large-scale file transfer, authentication



**Jetstream** is XSEDE's OpenStack-based cloud computing environment designed to give researchers access to interactive computing and data analysis resources on demand, whenever and wherever they want to analyze their data.

# Hands-on!

Let's build a compute cluster on Jetstream!

# Login to Jetstream @ IU via Horizon GUI

The screenshot shows a web browser window with the title "Login - OpenStack Dash". The address bar displays the URL <https://iu.jetstream-cloud.org>. The main content area features the OpenStack logo (a red square icon) and the word "openstack.®". Below the logo is a "Log in" button. The login form consists of three fields: "Domain" (set to "TACC"), "User Name" (set to "mckandes"), and "Password" (represented by a series of dots). To the right of the password field is an "eye" icon for password visibility. At the bottom right of the form is a blue "Sign In" button. The browser interface includes standard navigation buttons (back, forward, search, etc.) and a vertical scrollbar on the right side.

Log in

Domain

TACC

User Name

mckandes

Password

.....

Sign In

# Login to Jetstream @ IU via Horizon GUI

- ▶ URL: <https://iu.jetstream-cloud.org>
- ▶ Domain: TACC
- ▶ User Name: *Your TACC username*
- ▶ Password: *Your TACC password*

# OpenStack's Horizon GUI

The screenshot shows the OpenStack Horizon GUI interface. The browser title bar reads "Workbooks - OpenStack X". The URL in the address bar is <https://iu.jetstream-cloud.org/mistral/workbooks/>. The top navigation bar includes links for Home, Help, Logout, and a user dropdown for "mckandes". The main header has the "openstack." logo and the text "tacc • TG-CCR190013".

The left sidebar menu is collapsed, showing "Project", "Identity", and "Workflow". The "Workflow" item is expanded, showing "Workbooks", "Workflows", "Actions", "Workflow Executions", "Task Executions", "Action Executions", and "Cron Triggers".

The main content area is titled "Workbooks". It features a search bar with "Filter" and "Search" icons, and buttons for "+ Create Workbook" and "Update Workbook". A table header with columns "Name", "Tags", "Created", "Updated", and "Actions" is shown. Below the table, a message says "No items to display.".

The bottom of the screen contains a footer with navigation icons for back, forward, search, and other system functions.

# OpenStack's Horizon GUI (Identity)

The screenshot shows the OpenStack Horizon Identity interface. The browser address bar displays <https://u.jetstream-cloud.org/identity/>. The main navigation bar includes links for Projects, Identity, and Users. The current view is under the Identity category, specifically the Projects section. The page title is "Projects". On the left, there are navigation links for Project, Identity, and Application Credentials. The main content area displays a table of projects with columns: Name, Description, Project ID, Domain Name, Enabled, and Actions. There are four entries in the table:

Name	Description	Project ID	Domain Name	Enabled	Actions
TG-CCR 190013		6cb7d506aa71457e94f7597cca58459e	tacc	Yes	
TG-DDM 160003		442e4cb6552943218d23e377a54475d6	tacc	Yes	<button>Set as Active Project</button>
TG-STA 120004S		4f88c34c72134b9099a7f3bde0454a0c	tacc	No	
TG-STA 160003		cb2f8e43c2c747669fc03c8121e75954	tacc	Yes	<button>Set as Active Project</button>

Below the table, it says "Displaying 4 items". At the bottom of the page are standard browser navigation buttons.

# OpenStack's Horizon GUI (Compute)

Instance Overview - OpenStack

https://ui.jetstream-cloud.org/project/

openstack. tacc • TG-CCR190013 mckandes

Project API Access Compute Overview

Instances Images Key Pairs Server Groups Volumes Container Infra Network Orchestration Database Data Processing Object Store Share Identity Workflow

## Overview

### Limit Summary

#### Compute

Resource	Used	Total
Instances	0	25
VCPUs	0	132
RAM	0 Bytes	361.3GB
Volume	0	5
Volume Snapshots	0	100
Volume Storage	0 Bytes	1TB
Floating IPs	0	50
Security Groups	1	10
Security Group Rules	4	100

Instances: Used 0 of 25

VCPUs: Used 0 of 132

RAM: Used 0Bytes of 361.3GB

Volume: Used 0 of 5

Volume Snapshots: Used 0 of 100

Volume Storage: Used 0Bytes of 1TB

Floating IPs: Allocated 0 of 50

Security Groups: Used 1 of 10

Security Group Rules: Used 4 of 100

Navigation: Back, Forward, Home, Stop, Refresh, Help, More

# OpenStack's Horizon GUI (Volumes)

The screenshot shows the OpenStack Horizon GUI interface for managing volumes. The browser address bar displays the URL <https://u.jetstream-cloud.org/project/volumes/>. The top navigation bar includes links for Home, Help, and Logout, along with a search icon and a user dropdown for mckandes.

The main navigation sidebar on the left lists various services: Project, API Access, Compute, Volumes, Snapshots, Groups, Group Snapshots, Container Infra, Network, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. The "Volumes" link under the "Compute" section is currently selected, indicated by a blue background.

The central content area is titled "Volumes". It features a search bar with "Filter" and "Search" icons, and two buttons: "+ Create Volume" and "Accept Transfer". Below the search bar is a table header with columns: Name, Description, Size, Status, Group, Type, Attached To, Availability Zone, Bootable, Encrypted, and Actions. A message "No items to display." is shown below the table.

The bottom of the screen contains a footer with navigation icons for back, forward, search, and other system functions.

# OpenStack's Horizon GUI (Network)

The screenshot shows the Network Topology page in the OpenStack Horizon GUI. The URL is [https://iu.jetstream-cloud.org/project/network\\_topology/](https://iu.jetstream-cloud.org/project/network_topology/). The left sidebar shows various project categories like Compute, Volumes, Container Infra, and Network. The Network category is expanded, and Network Topology is selected. The main area displays a network topology diagram with three vertical bars representing segments: a blue bar for 'public' (IP range 149.165.156.0/23, 149.165.168.0/21), an orange bar for 'qnappc-net' (IP range 10.0.0.0/8), and a green bar for 'manila\_testing' (IP range 10.255.0.0/16). Buttons for Launch Instance, Create Network, and Create Router are visible at the top right.

Project / Network / Network Topology

Network Topology

Launch Instance    Create Network    Create Router

Topology    Graph

Small    Normal

Networks

Routers

Security Groups

Floating IPs

Orchestration

Database

Data Processing

Object Store

Share

Identity

Workflow

public  
149.165.156.0/23, 149.165.168.0/21

qnappc-net  
10.0.0.0/8

manila\_testing  
10.255.0.0/16

# OpenStack's Horizon GUI (Object Store)

The screenshot shows the OpenStack Horizon GUI interface for the Object Store. The browser title bar reads "Containers - OpenStack". The URL in the address bar is "https://u.jetstream-cloud.org/project/containers/". The top navigation bar includes links for Home, Help, Logout, and a user dropdown for "mckandes". The main header has the "openstack" logo and the text "tacc • TG-CCR190013".

The left sidebar, titled "Project API Access", lists several categories: Compute, Volumes, Container Infra, Network, Orchestration, Database, Data Processing, and Object Store. The "Object Store" category is expanded, showing "Containers" as the selected item, indicated by a blue background.

The main content area displays the "Containers" page. The breadcrumb navigation shows "Project / Object Store / Containers". A large callout box in the center says "Select a container to browse." with a small info icon. Below it, there is a search bar with a magnifying glass icon and the placeholder text "Click here for filter". A message below the search bar states "No items to display."

At the bottom of the page, there are navigation icons for back, forward, search, and refresh, along with a "Help" link.

# OpenStack's Horizon GUI (Shares)

The screenshot shows the OpenStack Horizon interface for the Shares service. The URL is https://iu.jetstream-cloud.org/project/shares/. The left sidebar has a 'Project' dropdown set to 'openstack'. Under 'API Access', the 'Compute' section is expanded, showing 'Shares' (selected), 'Volumes', 'Container Infra', and 'Network'. Below 'Shares' are 'Share Snapshots', 'Share Networks', 'Security Services', 'Share Groups', and 'Share Group Snapshots'. The main content area shows a table titled 'Shares' with columns: Name, Description, Metadata, Size, Status, Protocol, Visibility, Share Network, Share Group, and Actions. A message 'No items to display.' is shown. There is a 'Create Share' button at the top right of the table area.

Name	Description	Metadata	Size	Status	Protocol	Visibility	Share Network	Share Group	Actions
No items to display.									

Shares

Share Snapshots

Share Networks

Security Services

Share Groups

Share Group Snapshots

Identity

**WARNING**

# Our Jetstream Project is a Shared Account!

- ▶ We all have full access to the same project account on Jetstream.
- ▶ Please do not delete/modify/access any resources you have not created yourself without permission from either myself or the user who created them.



# Jetstream Public Wiki

Jetstream Public Wiki - x +

https://iujetstream.atlassian.net/wiki/spaces/JWT/overview

Jetstream Public Wiki

Jetstream

Jetstream is a user-friendly cloud computing environment for researchers based on Atmosphere and OpenStack. It is designed to provide configurable cyberinfrastructure that gives researchers access to interactive computing and data analysis resources on demand, whenever and wherever they want to analyze their data. For a more in-depth description please see the [System Overview](#).

- First-time users: New users will probably wish to review the [System Overview](#) and then jump in with the [Quick Start Guide](#) for how to obtain an account via the [XSEDE User Portal](#).
- Returning users: You can find a list of our documentation offerings below or you can search the wiki:

Our Troubleshooting and FAQ guide may have helpful suggestions for unusual problems.

?

Overview

Space Settings

SPACE SHORTCUTS

- Jetstream Website
- Jetstream featured images
- Quick Start Guide
- Log in to Jetstream's Web Interface
- XSEDE
- How-to articles
- GitHub Repository
- Galaxy on Jetstream Wiki
- Table of Contents

PAGES

- System Overview
- Known Issues
- Jetstream Allocations
- Jetstream Trial Access Allocation

https://iujetstream.atlassian.net/wiki/spaces/JWT/overview

# Jetstream Public Wiki

Using the OpenStack Horizon Interface

https://iujetstream.atlassian.net/wiki/spaces/JWT/pages/4482663

Jetstream Public Wiki / Using the Jetstream API

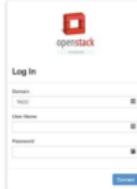
Jetstream Wiki Table of Contents

## Using the OpenStack Horizon GUI Interface

Created by Peg Lindenlaub  
Last updated Feb 06, 2019 by Jeremy Fischer

### Create Private Network

1: Login to Horizon:  
IU: <https://iu.jetstream-cloud.org/dashboard>  
TACC: <https://tacc.jetstream-cloud.org/dashboard>  
Domain: TACC  
User Name: your TACC username  
Password: your TACC password



2: Click on Project → Network → Network Topology → +Create Network



3: Enter a network name, for example, username\_net.  
After giving it a descriptive name, press the blue



<https://iujetstream.atlassian.net/wiki/spaces/JWT/overview>

# Creating a Private Network (1)

The screenshot shows the OpenStack Networks interface. The top navigation bar includes a back button, forward button, refresh button, and a URL field showing <https://lu.jetstream-cloud.org/project/networks/>. The main header says "openstack." and "Project / Network / Networks". A sidebar on the left lists various project categories: Compute, Volumes, Container Infra, Network (selected), Routers, Security Groups, Floating IPs, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. The "Network" section under "Project" is expanded, showing "Displaying 3 items". A table lists three networks:

Network Topology	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
Networks	genapp-net	genapp-net-subnet1 10.0.0.0/8	Yes	No	Active	UP	nova	<a href="#">Edit</a>
Routers	public	public 149.165.156.0/23 public-2 149.165.168.0/21	Yes	Yes	Active	UP	nova	<a href="#">Edit</a>
Security Groups	manila_testing	manilla_testing 10.255.0.0/16	Yes	No	Active	UP	nova	<a href="#">Edit</a>
Floating IPs								

The "Networks" tab is highlighted in blue. The "Create Network" button is located at the top right of the table header. The "Delete Networks" button is also present.

# Creating a Private Network (2)

The screenshot shows a web browser window for 'Networks - OpenStack' at the URL <https://lu.jetstream-cloud.org/project/networks/>. A modal dialog box titled 'Create Network' is open, overlaid on the main OpenStack dashboard.

The modal has three tabs: 'Network' (selected), 'Subnet', and 'Subnet Details'. The 'Network Name' field contains 'sdsc'. Below it are two checked checkboxes: 'Enable Admin State' and 'Create Subnet'. An 'Availability Zone Hints' dropdown menu is open, showing the option 'nova'. At the bottom of the modal are 'Cancel', '« Back', and 'Next »' buttons.

The background dashboard shows a sidebar with 'Project' navigation items: Compute, Volumes, Container Infra, Network (selected), and Network. The main area displays a table of networks with columns 'Name', 'Status', 'Subnets', and 'Actions'. A red button labeled 'Delete Networks' is visible in the Actions column.

# Creating a Private Network (3)

The screenshot shows the OpenStack Networks interface. The main menu on the left includes Compute, Volumes, Container Infra, Network, and Network. The Network tab is selected. The top navigation bar shows the URL <https://lu.jetstream-cloud.org/project/networks/>. A modal dialog box titled "Create Network" is open, with the "Subnet" tab selected. The "Subnet Name" field contains "hpc". The "Network Address Source" dropdown is set to "Enter Network Address manually" and contains the value "10.1.1.0/24". The "IP Version" dropdown is set to "IPv4". The "Gateway IP" field is empty. A checkbox labeled "Disable Gateway" is unchecked. A descriptive text block explains the requirements for creating a subnet. At the bottom of the dialog are "Cancel", "« Back", and "Next »" buttons.

Create Network

Network Subnet Subnet Details

Subnet Name: hpc

Network Address Source: Enter Network Address manually

Network Address\*: 10.1.1.0/24

IP Version: IPv4

Gateway IP:

Disable Gateway

Creates a subnet associated with the network. You need to enter a valid "Network Address" and "Gateway IP". If you did not enter the "Gateway IP", the first value of a network will be assigned by default. If you do not want gateway please check the "Disable Gateway" checkbox. Advanced configuration is available by clicking on the "Subnet Details" tab.

Cancel « Back Next »

# Creating a Private Network (4)

The screenshot shows a web browser window for 'Networks - OpenStack' at the URL <https://lu.jetstream-cloud.org/project/networks/>. The main page lists various project services like Compute, Volumes, Container Infra, and Network. A modal dialog box titled 'Create Network' is open, showing the 'Subnet Details' tab. It includes sections for 'Allocation Pools', 'DNS Name Servers', and 'Host Routes'. There is a checked checkbox for 'Enable DHCP' with a note: 'Specify additional attributes for the subnet.' At the bottom are 'Cancel', 'Back', and 'Create' buttons.

Networks - OpenStack X +

openstack tacc • TG-CCR190013 mckandes

Create Network

Network Subnet Subnet Details

Enable DHCP

Specify additional attributes for the subnet.

Allocation Pools

DNS Name Servers

Host Routes

Cancel Back Create

# Creating a Private Network (5)

The screenshot shows the OpenStack Networks interface. The top navigation bar includes a back button, forward button, refresh button, and a URL field showing <https://lu.jetstream-cloud.org/project/networks/>. The main header says "openstack." and "Project / Network / Networks". A sidebar on the left lists various project categories: Compute, Volumes, Container Infra, Network (selected), Routers, Security Groups, Floating IPs, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. The "Network" section under "Project" is expanded, showing "Displaying 4 items". The table lists four networks:

	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
<input type="checkbox"/>	sdsc	hpc 10.1.1.0/24	No	No	Active	UP	nova	<button>Edit Network</button>
<input type="checkbox"/>	genapp-net	genapp-net-subnet1 10.0.0.0/8	Yes	No	Active	UP	nova	
<input type="checkbox"/>	public	public 149.165.156.0/23 public-2 149.165.168.0/21	Yes	Yes	Active	UP	nova	
<input type="checkbox"/>	manila_testing	manila_testing 10.255.0.0/16	Yes	No	Active	UP	nova	

Below the table, it says "Displaying 4 items". At the bottom right are navigation icons for back, forward, search, and other functions.

# Creating a Router (1)

Network Topology - OpenStack

https://lu.jetstream-cloud.org/project/network\_topology

openstack. mckandes

Project / Network / Network Topology

API Access

Compute >

Volumes >

Container Infra >

Network > Network Topology

Launch Instance + Create Network + Create Router

Topology Graph

Small Normal

Networks

Routers

Security Groups

Floating IPs

Orchestration >

Database >

Data Processing >

Object Store >

Share >

Identity >

Workflow >

149.165.168.0/21, 149.165.156.0/23 public

10.1.1.0/24 sds-c

10.0.0.0/8 genappt-net

10.255.0.0/16 manila-testing

Navigation icons: back, forward, search, etc.

# Creating a Router (2)

Network Topology - OpenStack

openstack. tacc • TG-CCR190013

Project

Compute

Volumes

Container Instances

Networks

Networking

Server Groups

Orchestration

Database

Data Processing

Object Store

Share

Identity

Workflow

sdsc-router

public

sdsc

sdsc-pubnet

smalls testing

192.168.156.0/23, 149.165.168.0/21

10.1.1.0/24

10.0.0.0/8

10.255.0.0/16

Description:

Creates a router with specified parameters.

+ Create Router

Cancel

Create Router

# Creating a Router (3)

Network Topology - OpenStack

https://lu.jetstream-cloud.org/project/network\_topology

openstack. mckandes

Project / Network / Network Topology

API Access

Compute >

Volumes >

Container Infra >

Network > Network Topology

Launch Instance + Create Network + Create Router

Topology Graph

Small Normal

Networks

Routers

Security Groups

Floating IPs

Orchestration >

Database >

Data Processing >

Object Store >

Share >

Identity >

Workflow >

149.165.156.0/23, 149.165.168.0/21 Public

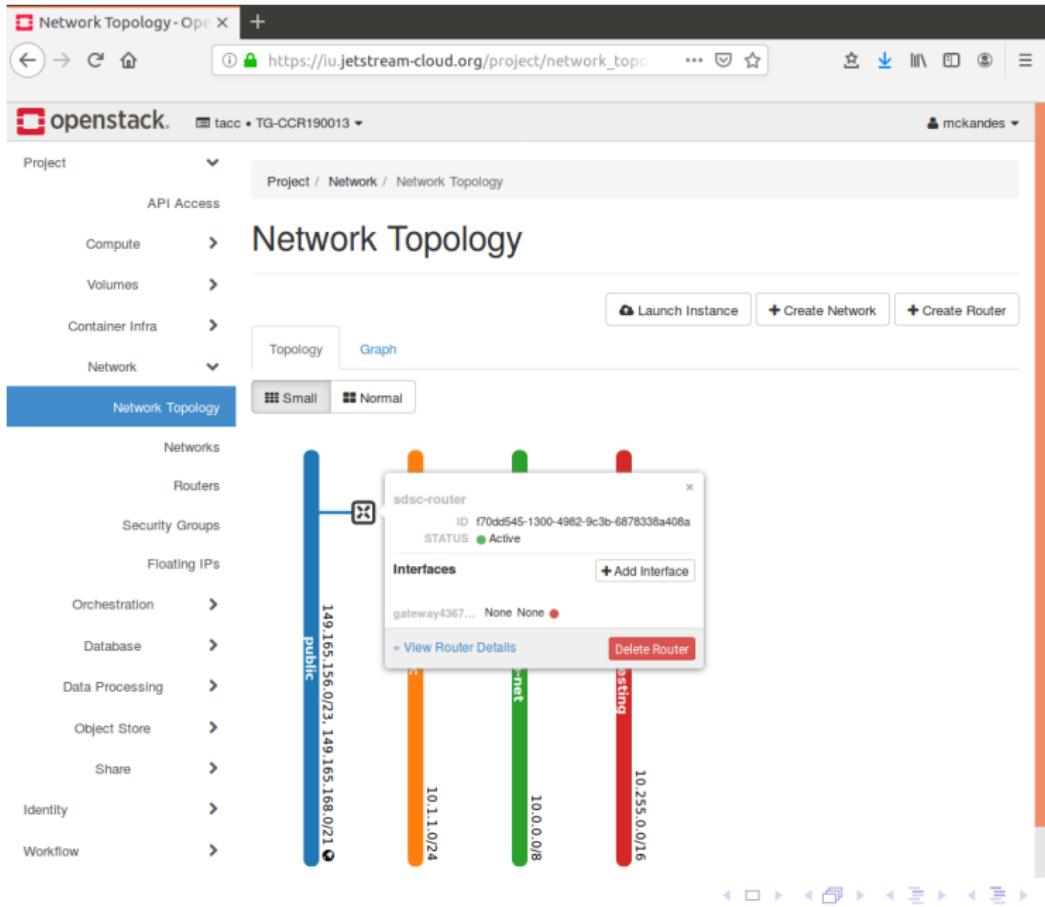
10.1.1.0/24 sds-c

10.0.0.0/8 genappt-net

10.255.0.0/16 manila-testing

Navigation icons: back, forward, search, etc.

## Creating a Router (4)



# Adding a Network Interface to the Router (1)

Network Topology - OpenStack

https://lu.jetstream-cloud.org/project/network\_topo

openstack. tacc • TG-CCR190013

mckandes

Add Interface

Subnet \*

sds: 10.1.1.0/24 (hpc)

Description:

You can connect a specified subnet to the router.

If you don't specify an IP address here, the gateway's IP address of the selected subnet will be used as the IP address of the newly created interface of the router. If the gateway's IP address is in use, you must use a different address which belongs to the selected subnet.

Create Router

Cancel Submit

Security Groups

Floating IPs

Orchestration >

Database >

Data Processing >

Object Store >

Share >

Identity

Workflow

149.165.156.0/23, 149.165.168.0/21

public

10.1.1.0/24

sds: 10.1.1.0/24

genapnet

10.0.0.0/8

manila testing

10.255.0.0/16

# Adding a Network Interface to the Router (2)

Network Topology - OpenStack

https://lu.jetstream-cloud.org/project/network\_topology

openstack. tacc • TG-CCR190013 mckandes

Project API Access

Compute > Network Topology

Volumes >

Container Infra >

Network >

Network Topology

Launch Instance + Create Network + Create Router

Topology Graph

Small Normal

Networks

Routers

Security Groups

Floating IPs

Orchestration >

Database >

Data Processing >

Object Store >

Share >

Identity >

Workflow >

149.165.156.0/23, 149.165.168.0/21 public

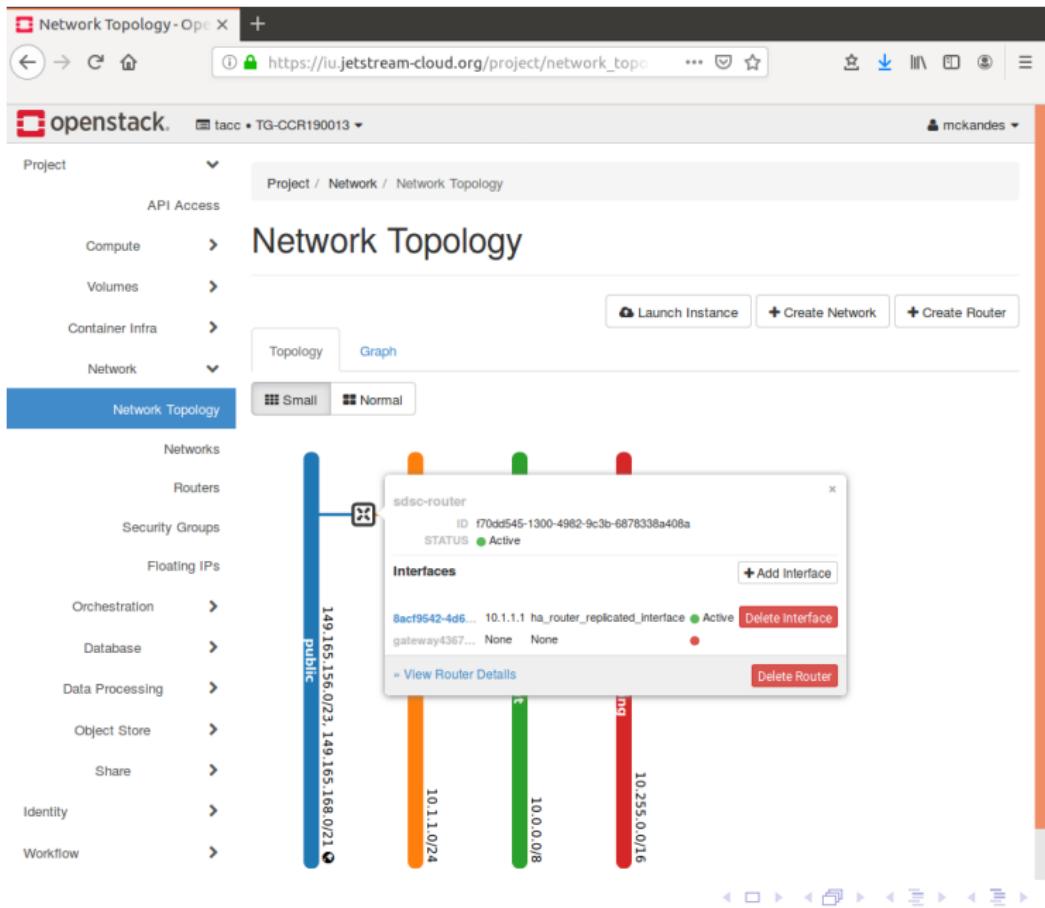
10.1.1.0/24 sdc

10.0.0.0/8 genappt-net

10.255.0.0/16 manila\_testing

Navigation icons: back, forward, search, etc.

## Adding a Network Interface to the Router (3)



# Modifying a Security Group (1)

The screenshot shows the OpenStack Horizon dashboard for a project named "tacc". The main navigation bar includes links for Home, Help, Logout, and a user icon. Below the bar, the URL https://lu.jetstream-cloud.org/project/security\_group is displayed. The left sidebar has a "Project" heading and lists "Compute", "Volumes", "Container Infra", and "Network" (which is expanded to show "Network Topology", "Networks", and "Routers"). The "Network" section is currently selected, indicated by a blue background. Under "Network", the "Security Groups" tab is also selected and highlighted in blue. The main content area is titled "Security Groups" and displays a table with one item:

<input type="checkbox"/>	Name	Security Group ID	Description	Actions
<input type="checkbox"/>	default	96afa9e5-2a53-43a4-b3e2-ff31c4700c1e	Default security group	<button>Manage Rules</button>

Below the table, it says "Displaying 1 item". At the bottom of the page, there are links for "Floating IPs", "Orchestration", "Database", "Data Processing", "Object Store", "Share", "Identity", and "Workflow". The bottom right corner features standard browser navigation icons.

# Modifying a Security Group (2)

The screenshot shows the "Manage Security Group" interface in a web browser. The URL is [https://lu.jetstream-cloud.org/project/security\\_group](https://lu.jetstream-cloud.org/project/security_group). The page title is "Manage Security Group". The navigation bar includes links for "Project", "Compute", "Volumes", "Container Infra", "Network", "Orchestration", "Database", "Data Processing", "Object Store", "Share", "Identity", and "Workflow". The "Network" section is expanded, showing "Network Topology", "Networks", and "Routers". The "Security Groups" section is also expanded, showing a table of security group rules.

**Manage Security Group Rules:**

- default (96afa9e5-2a53-43a4-b3e2-ff31c4700c1e)**

**Displaying 4 items**

	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
<input type="checkbox"/>	Egress	IPv4	Any	Any	0.0.0.0/0	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Egress	IPv6	Any	Any	::/0	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Ingress	IPv4	Any	Any	-	default	-	<button>Delete Rule</button>
<input type="checkbox"/>	Ingress	IPv6	Any	Any	-	default	-	<button>Delete Rule</button>

# Modifying a Security Group (3)

Manage Security Group X +

openstack tacc • TG-CCR190013 mckandes

Add Rule

Rule \* All ICMP

Description

Direction Ingress

Remote \* CIDR 0.0.0.0/0

Description:

Rules define which traffic is allowed to Instances assigned to the security group. A security group rule consists of three main parts:

**Rule:** You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

**Open Port/Port Range:** For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

**Remote:** You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Actions

Delete Rule

Delete Rule

Delete Rule

Delete Rule

Cancel Add

# Modifying a Security Group (4)

Manage Security Group X +

openstack tacc • TG-CCR190013 mckandes

Add Rule

Rule \* SSH

Description ?

Remote \* CIDR

CIDR ? 0.0.0.0/0

Description:

Rules define which traffic is allowed to Instances assigned to the security group. A security group rule consists of three main parts:

**Rule:** You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

**Open Port/Port Range:** For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

**Remote:** You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Actions

Delete Rule

Delete Rule

Delete Rule

Delete Rule

Cancel Add

Displaying 5 Items

# Modifying a Security Group (5)

Using the OpenStack Horizon interface, navigate to the Network tab and click on Router.

The interface shows a list of routers, including "tiny\_router".

**9: Select the subnet you just created from the dropdown list. Click Submit.**

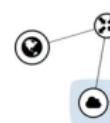
The "Add Interface" dialog box is displayed, showing the selected subnet "Test\_Network\_10.1.1.0/24 (Net, Subnet)".

The network should now be connected to the new router. The end result will look similar to the diagram below.

**Create Security Group**

Each project has its own firewall rules. You will need to add at least an ssh rule to allow yourself to ssh into your instances.

- On the left side bar, click on Network
- Security Groups
- Click +Create Group on the right side of the screen and give your new group a descriptive name and



## Modifying a Security Group (6)

# Modifying a Security Group (7)

The screenshot shows the "Manage Security Group" interface in a web browser. The URL is [https://lu.jetstream-cloud.org/project/security\\_group](https://lu.jetstream-cloud.org/project/security_group). The page title is "Manage Security Group". The left sidebar shows navigation categories like Project, API Access, Compute, Volumes, Container Infra, Network, Network Topology, Networks, Routers, Security Groups, Floating IPs, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. The "Security Groups" section is currently selected. The main content area displays a table titled "Manage Security Group Rules" for a specific security group with ID "default (96afa9e5-2a53-43a4-b3e2-ff31c4700c1e)". The table has columns: Direction, Ether Type, IP Protocol, Port Range, Remote IP Prefix, Remote Security Group, Description, and Actions. There are 8 items displayed. Each row includes a checkbox, the direction (Egress or Ingress), the ether type (IPv4 or IPv6), the IP protocol (Any, ICMP, TCP, UDP), the port range (Any or specific values like 22 (SSH) or 9618), the remote IP prefix (0.0.0.0/0 or ::/0), the remote security group (default), a description field, and a "Delete Rule" button.

Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
Egress	IPv4	Any	Any	0.0.0.0/0	-	-	<button>Delete Rule</button>
Egress	IPv6	Any	Any	::/0	-	-	<button>Delete Rule</button>
Ingress	IPv4	Any	Any	-	default	-	<button>Delete Rule</button>
Ingress	IPv4	ICMP	Any	0.0.0.0/0	-	-	<button>Delete Rule</button>
Ingress	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	-	<button>Delete Rule</button>
Ingress	IPv4	TCP	9618	0.0.0.0/0	-	-	<button>Delete Rule</button>
Ingress	IPv4	UDP	9618	0.0.0.0/0	-	-	<button>Delete Rule</button>

# Importing an SSH Public Key (1)

The screenshot shows the OpenStack dashboard with the "Key Pairs" section selected. The URL in the browser is [https://lu.jetstream-cloud.org/project/key\\_pairs](https://lu.jetstream-cloud.org/project/key_pairs). The main content area displays a table with a single row: "No items to display." At the top right of this table are three buttons: "+ Create Key Pair", "Import Public Key" (highlighted in blue), and "Delete Key Pairs".

Key Pairs

Click here for filters.

+ Create Key Pair Import Public Key Delete Key Pairs

	Name
No items to display.	

Project / Compute / Key Pairs

Project API Access

Compute Instances Images Key Pairs Server Groups Volumes Container Infra Network Orchestration Database Data Processing Object Store Share Identity Workflow

Click here for filters.

+ Create Key Pair Import Public Key Delete Key Pairs

No items to display.

Import Public Key

# What is an SSH key pair?

# How can I create an SSH key pair?

[https:](https://portal.xsede.org/knowledge-base/-/kb/document/aews)

//portal.xsede.org/knowledge-base/-/kb/document/aews

# Importing an SSH Public Key (2)

The screenshot shows the OpenStack Compute interface for managing key pairs. The left sidebar has a 'Key Pairs' tab selected. The main area displays a table with one item, 'mkandes-public-key', which has a red 'Delete Key Pair' button next to it. At the top right of the main area, there is a red 'Import Public Key' button.

Key Pairs

Name
mkandes-public-key

**Import Public Key**

# Importing an SSH Public Key (3)

Manage Security Group X +

openstack. tacc • TG-CCR190013 ▾ mckandes ▾

Project API Access Compute Volumes Container Infra Network

Manage Security Group Rules: default (96afa9e5-2a53-43a4-b3e2-ff31c4700c1e)

Add Rule Delete Rules

Displaying 8 items

	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
<input type="checkbox"/>	Egress	IPv4	Any	Any	0.0.0.0/0	-	-	Delete Rule
<input type="checkbox"/>	Egress	IPv6	Any	Any	::/0	-	-	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	Any	Any	-	default	-	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	ICMP	Any	0.0.0.0/0	-	-	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	-	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	TCP	9618	0.0.0.0/0	-	-	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	UDP	9618	0.0.0.0/0	-	-	Delete Rule

# Launching an Instance (1)

The screenshot shows the 'Instances' page of the OpenStack interface. The URL is https://lu.jetstream-cloud.org/project/instances/. The left sidebar has 'Compute' selected under 'Project'. The main content area shows the 'Instances' tab selected. At the top right are buttons for 'Instance ID =', 'Filter', and 'Launch Instance'. Below is a table with columns: Instance Name, Image Name, IP Address, Flavor, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. A message 'No items to display.' is shown. The bottom navigation bar includes icons for back, forward, search, and other navigation functions.

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
No items to display.										

## Launching an Instance (2)

Instances - OpenStack □ +

https://iu.jetstream-cloud.org/project/instances/

openstack. tacc • TG-CCR190013 ▾ mckandes

## Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

**Details**

**Source \***  
htc-mckandes

**Flavor \***

**Networks \***

**Network Ports**

**Security Groups**

**Key Pair**

**Description**

**Availability Zone**  
Any Availability Zone

**Count \***  
1

Total Instances  
(25 Max)  
4%

0 Current Usage  
1 Added  
24 Remaining

**Cancel** < Back Next > Launch Instance

Share >

Identity >

Workflow >

# Launching an Instance (3)

Instances - OpenStack X +

Instances - OpenStack https://lu.jetstream-cloud.org/project/instances/ mckandes

openstack. tacc • TG-CCR190013

Launch Instance

Details

Source

Flavor \*

Networks \*

Allocated

Name	Updated	Size	Type	Visibility
JS-API-Featured-CentOS7-May-20-2019	5/20/19 7:11 AM	8.00 GB	raw	Public

Security Groups

Key Pair

Available 384 Select one

Configuration

Q JS-API

Server Groups

Scheduler Hints

Metadata

Name	Updated	Size	Type	Visibility
JS-API-Featured-CentOS6-May-14-2019	5/14/19 8:07 PM	8.00 GB	raw	Public
JS-API-Featured-CentOS7-Intel-Developer-May-21-2019	5/21/19 11:32 AM	20.00 GB	raw	Public
JS-API-Featured-Ubuntu16-May-21-2019	5/21/19 1:23 PM	8.00 GB	raw	Public
JS-API-Featured-Ubuntu18-May-22-2019	5/22/19 7:21 PM	8.00 GB	raw	Public
JS-API-Genapp-1-0-alpha-Ubuntu16	3/19/19 11:58 AM	20.00 GB	raw	Public

Instances - OpenStack https://lu.jetstream-cloud.org/project/instances/ mckandes

openstack. tacc • TG-CCR190013

Launch Instance

Details

Source

Flavor \*

Networks \*

Allocated

Name	Updated	Size	Type	Visibility
JS-API-Featured-CentOS7-May-20-2019	5/20/19 7:11 AM	8.00 GB	raw	Public

Security Groups

Key Pair

Available 384 Select one

Configuration

Q JS-API

Server Groups

Scheduler Hints

Metadata

Name	Updated	Size	Type	Visibility
JS-API-Featured-CentOS6-May-14-2019	5/14/19 8:07 PM	8.00 GB	raw	Public
JS-API-Featured-CentOS7-Intel-Developer-May-21-2019	5/21/19 11:32 AM	20.00 GB	raw	Public
JS-API-Featured-Ubuntu16-May-21-2019	5/21/19 1:23 PM	8.00 GB	raw	Public
JS-API-Featured-Ubuntu18-May-22-2019	5/22/19 7:21 PM	8.00 GB	raw	Public
JS-API-Genapp-1-0-alpha-Ubuntu16	3/19/19 11:58 AM	20.00 GB	raw	Public

# Launching an Instance (4)

Instances - OpenStack X +

Instances - OpenStack https://lu.jetstream-cloud.org/project/instances/ mckandes

openstack. tacc • TG-CCR190013

Launch Instance

Details Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Name	VCPUS	RAM	Total Disk	Public	⋮
m1.tiny	1	2 GB	8 GB	Yes	

Networks \*

Available Select one

Network Ports Click here for filters.

Security Groups

Name	VCPUS	RAM	Total Disk	Public	⋮
m1.small	2	4 GB	20 GB	Yes	
m1.quad	4	10 GB	20 GB	Yes	
m1.medium	6	16 GB	60 GB	Yes	
s1.large	10	30 GB	240 GB	Yes	
m1.large	10	30 GB	60 GB	Yes	
s1.xlarge	24	60 GB	480 GB	Yes	
m1.xlarge	24	60 GB	60 GB	Yes	
s1.xxlarge	44	120 GB	960 GB	Yes	

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

⋮

Navigation icons: back, forward, search, etc.

## Launching an Instance (5)

Instances - OpenStack

https://iu.jetstream-cloud.org/project/instances/

Launch Instance

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Cancel

Share

Identity

Workflow

Allocated 1

No

Up

Active

Available 3

Click here for filters.

genapp-net

public

manlia\_testing

Network Shared Admin State Status

sdsc No Up Active

Select at least one network

Launch Instance

# Launching an Instance (6)

Instances - OpenStack X +

Instances - OpenStack https://lu.jetstream-cloud.org/project/instances/ mckandes

openstack. tacc • TG-CCR190013

Launch Instance

Details Ports provide extra communication channels to your instances. You can select ports instead of networks or a mix of both.

Source Allocated Select ports from those listed below.

Flavor

Name	Admin State	Status
Select an item from Available items below		

Networks Networks

Network Ports Available Select one

Name	Admin State	Status
Available		

Security Groups Security Groups

Key Pair Key Pair

Name	Admin State	Status
No available items		

Configuration Configuration

Server Groups Server Groups

Scheduler Hints Scheduler Hints

Metadata Metadata

Cancel Share >

Identity >

Workflow >

Next < Back Launch Instance

Navigation icons: back, forward, search, etc.

# Launching an Instance (7)

Instances - OpenStack X +

Instances - OpenStack https://lu.jetstream-cloud.org/project/instances/ mckandes

openstack. tacc • TG-CCR190013

Launch Instance

Details Select the security groups to launch the instance in.

Allocated ⓘ Name default

Available ⓘ Select one or more Click here for filters.

Name Description

No available items

Key Pair Configuration Server Groups Scheduler Hints Metadata

Cancel Share > Identity Workflow >

Back Next Launch Instance

Share >

Identity >

Workflow >

Back Next Launch Instance

Share >

Identity >

Workflow >

## Launching an Instance (8)

Instances - OpenStack

https://iu.jetstream-cloud.org/project/instances/

openstack • tacc • TG-CCR190013

mckandes

### Launch Instance

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

[+ Create Key Pair](#) [Import Key Pair](#)

**Allocated**

Displaying 1 item

Name
mkandes-public-key

**Available** 0

Select one

Click here for filters.

Displaying 0 items

No items to display.

Displaying 0 items

[Cancel](#) [Back](#) [Next](#) [Launch Instance](#)

## Launching an Instance (9)

Instances - OpenStack

https://iu.jetstream-cloud.org/project/instances/

openstack. tacc • TG-CCR190013

mckandes

### Launch Instance

Details Select the server group to launch the instance in.

Allocated

Source Name

Flavor Select a server group from the available groups below.

Networks Available 0 Select one

Network Ports Filter

Security Groups Name

Key Pair No available items

Configuration

Server Groups

Scheduler Hints

Metadata

Cancel

Share >

Identity >

Workflow >

< Back Next > Launch Instance

## Launching an Instance (10)

Instances - OpenStack X +

https://iu-jetstream-cloud.org/project/instances/ ... ☰ 🔍 🌐

openstack. tacc • TG-CCR190013 mckandes

## Launch Instance

This step allows you to add scheduler hints to your instance.

You can specify scheduler hints by moving items from the left column to the right column. In the left column there are scheduler hint definitions from the Glance Metadata Catalog. Use the "Custom" option to add scheduler hints with the key of your choice.

**Available Scheduler Hints**

Filter

Custom

No available scheduler hints

**Existing Scheduler Hints**

Filter

No existing scheduler hints

Click each item to get its description here.

Details  
Source  
Flavor  
Networks  
Network Ports  
Security Groups  
Key Pair  
Configuration  
Server Groups  
**Scheduler Hints**  
Metadata

# Launching an Instance (11)

The screenshot shows the 'Instances - OpenStack' interface on a web browser. The URL is <https://lu.jetstream-cloud.org/project/instances/>. The user is logged in as 'mckandes'. The main page title is 'openstack' and the tab is 'tacc \* TG-CCR190013'. A modal window titled 'Launch Instance' is open, showing the 'Metadata' step.

The left sidebar lists the following steps:

- Details
- Source
- Flavor** (selected)
- Networks
- Network Ports
- Security Groups
- Key Pair
- Configuration
- Server Groups
- Scheduler Hints
- Metadata** (selected)

The 'Metadata' step contains two columns:

- Available Metadata**: A table with a single row labeled 'Custom'. It includes a 'Filter' input and a magnifying glass icon.
- Existing Metadata**: A table with a single row labeled 'No existing metadata'. It includes a 'Filter' input and a magnifying glass icon.

A note at the bottom of the step says: "Click each item to get its description here."

# Launching an Instance (12)

Instances - OpenStack

https://iu.jetstream-cloud.org/project/instances/

openstack. facc • TG-CCR190013

mckandes

Project API Access Compute

Instances

Overview

Images Displaying 1 item

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
htc-mka-ndes	JS-API-Featured-CentOS-7-May-20-2019	10.1.1.7	m1.tiny	mikandes-public-key	Active	zone-r2	None	Running	2 hours, 1 minute	Create Snapshot

Key Pairs

Server Groups

Volumes >

Container Infra >

Network >

Orchestration >

Database >

Data Processing >

Object Store >

Share >

Identity >

Workflow >

Instance ID Filter Launch Instance Delete Instances More Actions

Instances

Displaying 1 item

htc-mka-ndes JS-API-Featured-CentOS-7-May-20-2019 10.1.1.7 m1.tiny mikandes-public-key Active zone-r2 None Running 2 hours, 1 minute Create Snapshot

Navigation icons: back, forward, search, etc.

# Associate a Floating (Public) IP with an Instance (1)

The screenshot shows the OpenStack Instances dashboard. On the left, a sidebar lists various project categories: Project, API Access, Compute, Overview, Instances, Images, Key Pairs, Server Groups, Volumes, Container Infra, Network, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. The 'Compute' category is selected. In the main content area, the 'Instances' tab is active. A table displays one instance named 'htc-mka-ndes'. The table columns include Instance Name, Image Name, IP Address, Flavor, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. The 'Actions' column for the instance contains a dropdown menu with options: Associate Floating IP, Attach Interface, Detach Interface, Edit Instance, Attach Volume, Detach Volume, Update Metadata, Edit Security Groups, Edit Port Security Groups, Console, View Log, Pause Instance, Suspend Instance, and Shelve Instance. The 'Associate Floating IP' option is highlighted with a red box. At the bottom of the page, there are navigation icons for back, forward, search, and refresh.

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
htc-mka-ndes	JS-API-Feature-d-CentOS7-May-20-2019	10.1.1.7	m1.tiny	mkandes-public-key	Active	zone-r2	None	Running	2 hours, 1 minute	<ul style="list-style-type: none"><li><a href="#">Create Snapshot</a></li><li><a href="#">Associate Floating IP</a></li><li><a href="#">Attach Interface</a></li><li><a href="#">Detach Interface</a></li><li><a href="#">Edit Instance</a></li><li><a href="#">Attach Volume</a></li><li><a href="#">Detach Volume</a></li><li><a href="#">Update Metadata</a></li><li><a href="#">Edit Security Groups</a></li><li><a href="#">Edit Port Security Groups</a></li><li><a href="#">Console</a></li><li><a href="#">View Log</a></li><li><a href="#">Pause Instance</a></li><li><a href="#">Suspend Instance</a></li><li><a href="#">Shelve Instance</a></li></ul>

# Associate a Floating (Public) IP with an Instance (2)

The screenshot shows the OpenStack Instances interface on a web browser. The URL is https://iu.jetstream-cloud.org/project/instances/. The sidebar on the left lists various project services: API Access, Compute, Overview, Instances, Images, Key Pairs, and Server Groups. Under Compute, the 'Instances' tab is selected. A modal dialog box titled 'Manage Floating IP Associations' is open. Inside the dialog, there is an 'IP Address' dropdown menu with the option 'No floating IP addresses allocated'. Below it is a note: 'Select the IP address you wish to associate with the selected instance or port.' Another dropdown menu labeled 'Port to be associated' contains the entry 'htc-mkandes: 10.1.1.7'. At the bottom right of the dialog are 'Cancel' and 'Associate' buttons. In the background, the main instance list shows one item: 'htc-mkandes' with IP '10.1.1.7', status 'Active', and other details like 'm1.tiny', 'public-key', and 'zone=r2'. The instance was created on '7-May-2019' and has been running for '2 hours, 1 minute'. There is also a 'Create Snapshot' button next to the instance details.

# Associate a Floating (Public) IP with an Instance (3)

The screenshot shows the OpenStack Instances interface on a web browser. The URL is <https://iu.jetstream-cloud.org/project/instances/>. The left sidebar lists various project categories: API Access, Compute, Overview, Instances (selected), Images, Key Pairs, Server Groups, Volumes, Container Infra, Network, Orchestration, Database, Data Processing, Object Store, Share, Identity, and Workflow. A modal dialog box titled "Allocate Floating IP" is open in the center. It contains fields for "Pool" (set to "public"), "Description" (empty), "DNS Domain" (empty), and "DNS Name" (empty). Below these fields is a "Project Quotas" section showing "Floating IP" usage at 0 of 50 Used. At the bottom of the dialog are "Cancel" and "Allocate IP" buttons. The background shows a list of instances with columns for Name, Status, and Actions.

# Associate a Floating (Public) IP with an Instance (4)

The screenshot shows the OpenStack Instances interface on a web browser. The URL is https://iu.jetstream-cloud.org/project/instances/. The sidebar on the left lists various project services: API Access, Compute, Overview, Instances, Images, Key Pairs, and Server Groups. Under Compute, the 'Instances' tab is selected. A modal dialog box titled 'Manage Floating IP Associations' is open. Inside the dialog, there is a form with two fields: 'IP Address' containing '149.165.156.146' and 'Port to be associated' containing 'htc-mkandes: 10.1.1.7'. Below the form is a note: 'Select the IP address you wish to associate with the selected instance or port.' At the bottom right of the dialog are 'Cancel' and 'Associate' buttons. In the background, the main instance list shows one entry: 'htc-mkandes' (JS-API-F featured-7-May-2019), which is currently selected. The instance details show it's running on node 'm1.tiny' in zone 'r2' with a public key. The status bar at the bottom indicates the browser version is 11.0.

# Associate a Floating (Public) IP with an Instance (5)

The screenshot shows the OpenStack Instances page at <https://iu.jetstream-cloud.org/project/instances/>. The navigation bar includes 'Project / Compute / Instances'. The main area displays the 'Instances' table with one item:

Instance ID	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
	htc-mka	JS-API-Feature	10.1.1.7	m1.tiny	mkandas-key	Active	zone-r2	None	Running	2 hours, 7 minutes	<a href="#">Create Snapshot</a>

The table also lists floating IPs: 'Floating IPs: 149.165.156.146'.

# Login to an Instance via SSH

```
mkandes@castlebravo:~$ ssh centos@149.165.156.146
The authenticity of host '149.165.156.146 (149.165.156.146)' can't be established.
ECDSA key fingerprint is SHA256:78++lbs0hrY1/oDGXU79vM9vxPlgee6jPZIfVx0GuAo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '149.165.156.146' (ECDSA) to the list of known hosts.
Last login: Mon May 20 13:57:13 2019 from jbm2.jetstream.iu.edu
[centos@htc-mkandes ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9000
      inet 10.1.1.7 netmask 255.255.255.0 broadcast 10.1.1.255
      inet6 fe80::f816:3eff:feeb:73b1 prefixlen 64 scopeid 0x20<link>
        ether fa:16:3e:eb:73:b1 txqueuelen 1000 (Ethernet)
          RX packets 3462 bytes 24343907 (23.2 MiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 2456 bytes 210670 (205.7 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
          RX packets 19 bytes 1344 (1.3 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 19 bytes 1344 (1.3 KiB)
```

```
ssh centos@<your-instances-floating-ip-address>
ifconfig
```

# Switch to the 'root' user

```
mkandes@castlebravo:~$ ssh centos@149.165.171.142
Last login: Fri May 24 23:16:30 2019 from wireless-169-228-112-11.ucsd.edu
[centos@htc-mkandes ~]$ whoami
centos
[centos@htc-mkandes ~]$ hostname
htc-mkandes.novalocal
[centos@htc-mkandes ~]$ sudo -i
-bash-4.2# bash
[root@htc-mkandes ~]# whoami
root
[root@htc-mkandes ~]# █
```

**sudo -i**

## Okay, now what?

Well, if we want to build a cluster in the cloud, we'll probably want a batch job scheduler to manage job requests and resources ...



HTCondor is a specialized workload management system for compute-intensive jobs. It provides a job queueing mechanism, scheduling policy, priority scheme, resource monitoring, and resource management.

# Install HTCondor on your Instance

```
Verifying : perl-IO-Compress-2.061-2.el7.noarch          6/11
Verifying : perl-Archive-Tar-1.92-2.el7.noarch          7/11
Verifying : condor-procd-8.8.2-1.el7.x86_64            8/11
Verifying : 1:perl-Compress-Raw-Zlib-2.061-4.el7.x86_64 9/11
Verifying : perl-Compress-Raw-Bzip2-2.061-3.el7.x86_64 10/11
Verifying : condor-8.8.2-1.el7.x86_64                  11/11

Installed:
  condor.x86_64 0:8.8.2-1.el7

Dependency Installed:
  condor-classads.x86_64 0:8.8.2-1.el7
  condor-external-libs.x86_64 0:8.8.2-1.el7
  condor-procd.x86_64 0:8.8.2-1.el7
  perl-Archive-Tar.noarch 0:1.92-2.el7
  perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7
  perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7
  perl-IO-Compress.noarch 0:2.061-2.el7
  perl-IO-Zlib.noarch 1:1.10-294.el7_6
  perl-Package-Constants.noarch 1:0.02-294.el7_6
  python2-condor.x86_64 0:8.8.2-1.el7

Complete!
[root@htc-mkandes ~]#
```

You can follow the [installation instructions for HTCondor on CentOS 7](#). Note, however, before you *start* (and *enable*) condor in Step 3, you'll want to provide condor with a configuration file.

# HTCondor Manual

The screenshot shows a web browser window displaying the HTCondor™ Version 8.8.2 Manual. The title bar reads "HTCondor™ Version 8.8.2 Manual". The address bar shows the URL "research.cs.wisc.edu/htcondor/manual/v8.8/index.html". The page content includes the title "HTCondor™ Version 8.8.2 Manual", the subtitle "Center for High Throughput Computing, University of Wisconsin-Madison", the date "April 11, 2019", and a "Contents" section. The "Contents" section is organized into three main sections: "Overview", "Users' Manual", and "Administrators' Manual", each with a list of sub-topics.

**HTCondor™ Version 8.8.2 Manual**

Center for High Throughput Computing, University of Wisconsin-Madison

April 11, 2019

**Contents**

**1 Overview**

- [1.1 High-Throughput Computing \(HTC\) and Its Requirements](#)
- [1.2 HTCondor's Power](#)
- [1.3 Exceptional Features](#)
- [1.4 Current Limitations](#)
- [1.5 Availability](#)
- [1.6 Contributions and Acknowledgments](#)
- [1.7 Contact Information](#)
- [1.8 Privacy Notice](#)

**2 Users' Manual**

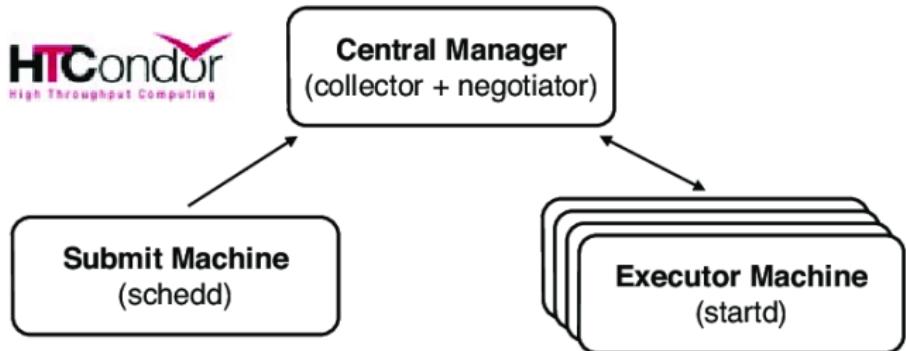
- [2.1 Welcome to HTCondor](#)
- [2.2 Introduction](#)
- [2.3 Matchmaking with ClassAds](#)
- [2.4 Running a Job: the Steps To Take](#)
- [2.5 Submitting a Job](#)
- [2.6 Managing a Job](#)
- [2.7 Priorities and Preemption](#)
- [2.8 Java Applications](#)
- [2.9 Parallel Applications \(Including MPI Applications\)](#)
- [2.10 DAGMan Applications](#)
- [2.11 Virtual Machine Applications](#)
- [2.12 Docker Universe Applications](#)
- [2.13 Time Scheduling for Job Execution](#)
- [2.14 Special Environment Considerations](#)
- [2.15 Potential Problems](#)

**3 Administrators' Manual**

- [3.1 Introduction](#)
- [3.2 Installation, Start Up, Shut Down, and Reconfiguration](#)
- [3.3 Introduction to Configuration](#)
- [3.4 Configuration Templates](#)
- [3.5 Configuration Macros](#)
- [3.6 User Priorities and Negotiation](#)
- [3.7 Policy Configuration for Execute Hosts and for Submit Hosts](#)
- [3.8 Security](#)

The latest version of the [HTCondor Manual](#).

# HTCondor Architecture



A standard HTCondor cluster (or pool) has **three different standard machine types**: a *Submit node*, where users submit their job/resource requests; a *Central Manager node*, where user jobs are matched to compute resources; and *Execute/Worker nodes*, where user jobs are executed.

# HTCondor Configuration Files

HTCondor configuration files allow you to customize the type of HTCondor machine you're running and how they operate.

You'll place your HTCondor configuration files in the following directory on your instance: /etc/condor/config.d

To start, we'll configure your instance with HTCondor installed as a *Personal HTCondor* pool, which runs a complete, self-contained condor cluster (with a submit, central manager, and execute node) all on your one running instance.

## Configure your Personal HTCondor Pool

You can download the `00-jetstream-htcondor-personal.conf` file from the IU OpenStack Object Store container I've created called `sdsc-hpc-intro-to-cloud-computing`.

Once you've downloaded the file, you can either upload it to your instance and install it (as root) in the `/etc/condor/config.d` directory OR you can simply create a new file in that directory and copy and paste the contents of the configuration file there.

## 00-jetstream-htcondor-personal.conf

```
DAEMON_LIST = MASTER, SCHEDD, COLLECTOR, NEGOTIATOR, STARTD

CONDOR_HOST = 127.0.0.1
COLLECTOR_PORT = 9618
COLLECTOR_HOST = $(CONDOR_HOST):$(COLLECTOR_PORT)

BIND_ALL_INTERFACES = False
NETWORK_INTERFACE = 127.0.0.1

SEC_DEFAULT_AUTHENTICATION = REQUIRED
SEC_DEFAULT_AUTHENTICATION_METHODS = FS
SEC_DEFAULT_ENCRYPTION = REQUIRED
SEC_DEFAULT_INTEGRITY = REQUIRED
SEC_DEFAULT_NEGOTIATION = REQUIRED
```

# Start your Personal HTCondor Pool

With configuration file `00-jetstream-htcondor-personal.conf` in `/etc/condor/config.d`, you can now *start* and *enable* HTCondor on your instance by running the following commands (as root):

```
systemctl start condor  
systemctl enable condor
```

If you already ran these commands (as outlined in the HTCondor installation instructions), then you must run the *restart* command to load the new condor configuration file (or when you make any future changes to the configuration file(s)).

```
systemctl restart condor
```

# Check your HTCondor Pool is Running (1)

```
[root@htc-mkandes config.d]# systemctl status condor
● condor.service - Condor Distributed High-Throughput-Computing
  Loaded: loaded (/usr/lib/systemd/system/condor.service; enabled; vendor prese
t: disabled)
  Active: active (running) since Sat 2019-05-25 01:19:42 UTC; 45s ago
    Main PID: 25445 (condor_master)
      Status: "All daemons are responding"
     CGroup: /system.slice/condor.service
             └─25445 /usr/sbin/condor_master -f
                  ├─25488 condor_procd -A /var/run/condor/procd_pipe -L /var/log/con...
                  ├─25489 condor_shared_port -f
                  ├─25490 condor_collector -f
                  ├─25491 condor_schedd -f
                  ├─25492 condor_negotiator -f
                  └─25493 condor_startd -f

May 25 01:19:42 htc-mkandes.novalocal systemd[1]: Started Condor Distributed ...
Hint: Some lines were ellipsized, use -l to show in full.
[root@htc-mkandes config.d]#
```

```
systemctl status condor
```

## Check your HTCondor Pool is Running (2)

```
[centos@htc-mkandes ~]$ condor_q

-- Schedd: htc-mkandes.novalocal : <127.0.0.1:9618?... @ 05/25/19 01:25:54
OWNER BATCH_NAME      SUBMITTED      DONE      RUN      IDLE      HOLD      TOTAL JOB_IDS
Total for query: 0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
Total for centos: 0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
Total for all users: 0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended

[centos@htc-mkandes ~]$ █
```

condor\_q

## Check your HTCondor Pool is Running (3)

```
[centos@htc-mkandes ~]$ condor_status
Name          OpSys      Arch   State       Activity LoadAv Mem  ActvtyT
htc-mkandes.novalocal LINUX      X86_64 Unclaimed Idle        0.000 1838  0+00:04

Machines Owner Claimed Unclaimed Matched Preempting  Drain
X86_64/LINUX      1      0      0      1      0      0      0
Total            1      0      0      1      0      0      0
[centos@htc-mkandes ~]$ █
```

[condor\\_status](#)

## Submit your first HTCondor job (1)

From the IU OpenStack Object Store container, download the `bash_pi.sh` and `bash_pi.htcondor` files and then upload them to the centos user account on your instance.

`bash_pi.sh` is a Bash script that computes an approximation of the number  $\pi$  using a simple Monte carlo method. For example, see this [reference](#).

`bash_pi.htcondor` is an HTCondor job submission script, which will submit the `bash_pi.sh` calculation to your HTCondor pool.

## bash\_pi.sh

```
#!/usr/bin/env bash
#
# Estimate the value of Pi via Monte Carlo

# Read in and parse input variables from command-line arguments
if (( "${#}" > 0 )); then
    while (( "${#}" > 0 )); do
        case "${1}" in
            -b | --bytes ) bytes="${2}" ;;
            -r | --round ) round="${2}" ;;
            -s | --samples ) samples="${2}" ;;
        esac
        shift 2
    done
fi

# Calculate the maximum value for an unsigned integer of size $bytes
maxint=$(echo "2^(8*${bytes})-1" | bc )

# Initialize sample counts; $inside will count the number of samples
# that are located 'inside' the radius of a unit circle, while $outside
# will count the number of samples that are located 'outside' the radius
# of a unit circle.
inside=0
outside=0

for (( i = 1; i <= $samples; i++ )); do
    # Get two uniformly distributed random integers of size $bytes
    j=$(od -An -N ${bytes} -t u${bytes} -v < /dev/urandom)
    k=$(od -An -N ${bytes} -t u${bytes} -v < /dev/urandom)

    # Normalize these random integers with respect to $maxint to obtain
    # two uniformly distributed random real numbers on unit interval
    x=$(bc <<< "scale=${round}; ${j}/${maxint}" )
    y=$(bc <<< "scale=${round}; ${k}/${maxint}" )
```



## bash\_pi.htcondor

```
# A sample HTCondor submit description file
universe = vanilla
executable = bash_pi.sh
arguments = -b 8 -r 5 -s 10000
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
output = bash_pi.out.$(ClusterId).$(ProcId)
error = bash_pi.err.$(ClusterId).$(ProcId)
log = bash_pi.log.$(ClusterId).$(ProcId)
request_cpus = 1
request_memory = 1024
queue 1
```

## Submit your first HTCondor job (2)

```
mkandes@castlebravo:~$ ssh centos@149.165.171.142
Last login: Sat May 25 01:08:58 2019 from rrcs-24-43-78-138.west.biz.rr.com
[centos@htc-mkandes ~]$ ls
bash_pi.htcondor  bash_pi.sh
[centos@htc-mkandes ~]$ condor_submit bash_pi.htcondor
Submitting job(s).
1 job(s) submitted to cluster 1.
[centos@htc-mkandes ~]$ condor_q

-- Schedd: htc-mkandes.novalocal : <127.0.0.1:9618?... @ 05/25/19 02:33:30
OWNER  BATCH_NAME      SUBMITTED      DONE      RUN      IDLE      TOTAL JOB_IDS
centos  ID: 1          5/25 02:33      -         1        -         1 1.0

Total for query: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for centos: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for all users: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended

[centos@htc-mkandes ~]$ █
```

`condor_submit bash_pi.htcondor`

## Submit your first HTCondor job (3)

```
[centos@htc-mkandes ~]$ condor_status
Name          OpSys     Arch   State    Activity LoadAv Mem  ActvtyT
htc-mkandes.novalocal LINUX     X86_64 Claimed   Busy      0.000 1838  0+00:00
                                         Machines Owner Claimed Unclaimed Matched Preempting Drain
                                         X86_64/LINUX 1      0       1        0       0       0       0
                                         Total    1      0       1        0       0       0       0
[centos@htc-mkandes ~]$
```

In addition to using the `condor_q` command to check the status of your job, you can use the `condor_status` command to check the status of the resources (CPUs, Memory, etc) connected to your Personal HTCondor cluster.

## Submit your first HTCondor job (4)

Name	OpSys	Arch	State	Activity	LoadAv	Mem	ActvtyT
htc-mkandes.novalocal	LINUX	X86_64	Unclaimed	Idle	0.080	1838	0+00:00
Machines Owner Claimed Unclaimed Matched Preempting Drain							
X86_64/LINUX	1	0	0	1	0	0	0
Total	1	0	0	1	0	0	0
[centos@htc-mkandes ~]\$ ls -lahtr							
total 3.2M							
-rw-r--r--. 1 centos centos 231 Dec 6 2016 .bashrc							
-rw-r--r--. 1 centos centos 193 Dec 6 2016 .bash_profile							
-rw-r--r--. 1 centos centos 18 Dec 6 2016 .bash_logout							
drwx-----. 2 centos centos 29 Feb 13 2018 .ssh							
drwxr-xr-x. 4 root root 34 May 24 23:53 ..							
-rw-----. 1 centos centos 178 May 25 02:32 .bash_history							
-rw-r--r--. 1 centos centos 1.6K May 25 02:32 bash_pi.sh							
-rw-r--r--. 1 centos centos 357 May 25 02:32 bash_pi.htcondor							
-rw-r--r--. 1 centos centos 1 May 25 02:34 bash_pi.out.1.0							
drwx-----. 3 centos centos 206 May 25 02:34 .							
-rw-r--r--. 1 centos centos 3.2M May 25 02:34 bash_pi.err.1.0							
-rw-r--r--. 1 centos centos 1.1K May 25 02:34 bash_pi.log.1.0							
[centos@htc-mkandes ~]\$ █							

Once your job completes, check the bash\_pi.out, bash\_pi.err, and bash\_pi.log files. What do you find?

## Submit your first HTCondor job (5)

```
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 38: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 41: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 34: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 35: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 38: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 41: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 34: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 35: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 38: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 41: bc: command not foun
d /var/lib/condor/execute/dir_29685/condor_exec.exe: line 49: bc: command not foun
d [centos@htc-mkandes ~]$ █
```

The job failed! We forgot to install the bc package used by the bash\_pi.sh script.

## Submit your first HTCondor job (6)

```
Installing:
bc           x86_64          1.06.95-13.el7          base        115 k

Transaction Summary
=====
Install 1 Package

Total download size: 115 k
Installed size: 215 k
Is this ok [y/d/N]: y
Downloading packages:
bc-1.06.95-13.el7.x86_64.rpm | 115 kB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : bc-1.06.95-13.el7.x86_64          1/1
  Verifying  : bc-1.06.95-13.el7.x86_64          1/1

Installed:
  bc.x86_64 0:1.06.95-13.el7

Complete!
[root@htc-mkandes ~]#
```

Install bc using the yum package manager (as root).

## Submit your first HTCondor job (7)

```
Complete!
[root@htc-mkandes ~]# exit
exit
-bash-4.2# exit
logout
[centos@htc-mkandes ~]$ condor_submit bash_pi.htcondor
Submitting job(s).
1 job(s) submitted to cluster 2.
[centos@htc-mkandes ~]$ condor_q

-- Schedd: htc-mkandes.novalocal : <127.0.0.1:9618?... @ 05/25/19 02:38:04
OWNER  BATCH_NAME   SUBMITTED   DONE    RUN    IDLE  TOTAL JOB_IDS
centos  ID: 2        5/25 02:37      -       1      -      1 2.0

Total for query: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for centos: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for all users: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended

[centos@htc-mkandes ~]$ █
```

After installing bc, resubmit the bash\_pi.htcondor job to your Personal HTCondor pool.

## Submit your first HTCondor job (8)

```
-- Schedd: htc-mkandes.novalocal : <127.0.0.1:9618?... @ 05/25/19 02:38:04
OWNER  BATCH_NAME      SUBMITTED     DONE      RUN      IDLE      TOTAL  JOB_IDS
centos  ID: 2          5/25 02:37      -         1        -         1 2.0

Total for query: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for centos: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
Total for all users: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended

[centos@htc-mkandes ~]$ condor_status
Name           OpSys      Arch   State       Activity LoadAv Mem   ActvtyT
htc-mkandes.novalocal  LINUX      X86_64  Claimed    Busy      0.000 1838  0+00:00

                         Machines Owner  Claimed Unclaimed Matched Preempting  Drain
X86_64/LINUX          1      0      1      0      0      0      0      0
Total                  1      0      1      0      0      0      0      0
[centos@htc-mkandes ~]$
```

Monitor the status of your new job with either the `condor_q` or `condor_status` commands.

## Submit your first HTCondor job (9)

```
X86_64/LINUX      1      0      0      1      0      0      0      0
Total      1      0      0      1      0      0      0      0
[centos@htc-mkandes ~]$ cat bash_pi.out.2.0
3.13640
[centos@htc-mkandes ~]$ ls -lahtr
total 3.2M
-rw-r--r--. 1 centos centos  231 Dec  6  2016 .bashrc
-rw-r--r--. 1 centos centos  193 Dec  6  2016 .bash_profile
-rw-r--r--. 1 centos centos   18 Dec  6  2016 .bash_logout
drwx-----. 2 centos centos  29 Feb 13  2018 .ssh
drwxr-xr-x. 4 root   root    34 May 24 23:53 ..
-rw-----. 1 centos centos  178 May 25 02:32 .bash_history
-rw-r--r--. 1 centos centos 1.6K May 25 02:32 bash_pi.sh
-rw-r--r--. 1 centos centos  357 May 25 02:32 bash_pi.htcondor
-rw-r--r--. 1 centos centos    1 May 25 02:34 bash_pi.out.1.0
-rw-r--r--. 1 centos centos 3.2M May 25 02:34 bash_pi.err.1.0
-rw-r--r--. 1 centos centos 1.1K May 25 02:34 bash_pi.log.1.0
-rw-r--r--. 1 centos centos     8 May 25 02:39 bash_pi.out.2.0
-rw-r--r--. 1 centos centos     0 May 25 02:39 bash_pi.err.2.0
drwx-----. 3 centos centos  275 May 25 02:39 .
-rw-r--r--. 1 centos centos 1.1K May 25 02:39 bash_pi.log.2.0
[centos@htc-mkandes ~]$ █
```

Once the job completes, check if you now get the expected results.

## Reconfigure your Personal HTCondor as Execute Node (1)

We now want to reconfigure our Personal HTCondor instance to be a standalone HTCondor execute node and have it join a larger, shared HTCondor pool.

To start, download the HTCondor configuration file  
20-jetstream-htcondor-execute.conf from the IU Object Store container sdsc-hpc-intro-to-cloud-computing.

Once downloaded, upload the configuration file to your Personal HTCondor instance.

## Reconfigure your Personal HTCondor as Execute Node (2)

```
mkandes@castlebravo:~$ ssh centos@149.165.171.142
Last login: Tue May 28 18:59:51 2019 from wireless-169-228-102-18.ucsd.edu
[centos@htc-mkandes ~]$ sudo -i
-bash-4.2# bash
[root@htc-mkandes ~]# cd /etc/condor/config.d/
[root@htc-mkandes config.d]# ls
00-jetstream-htcondor-personal.conf
[root@htc-mkandes config.d]# systemctl stop condor
[root@htc-mkandes config.d]# systemctl status condor
● condor.service - Condor Distributed High-Throughput-Computing
   Loaded: loaded (/usr/lib/systemd/system/condor.service; enabled; vendor prese
t: disabled)
     Active: inactive (dead) since Tue 2019-05-28 20:01:08 UTC; 6s ago
       Process: 25445 ExecStart=/usr/sbin/condor_master -f (code=exited, status=0/SUC
CESS)
      Main PID: 25445 (code=exited, status=0/SUCCESS)
        Status: "All daemons are responding"

May 25 01:19:42 htc-mkandes.novalocal systemd[1]: Started Condor Distributed ...
May 28 20:01:08 htc-mkandes.novalocal systemd[1]: Stopping Condor Distributed...
May 28 20:01:08 htc-mkandes.novalocal systemd[1]: Stopped Condor Distributed ...
Hint: Some lines were ellipsized, use -l to show in full.
[root@htc-mkandes config.d]# █
```

Next, stop condor on your instance.

## Reconfigure your Personal HTCondor as Execute Node (2)

```
[root@htc-mkandes config.d]# pwd
/etc/condor/config.d
[root@htc-mkandes config.d]# ls
00-jetstream-htcondor-personal.conf
[root@htc-mkandes config.d]# rm 00-jetstream-htcondor-personal.conf
rm: remove regular file '00-jetstream-htcondor-personal.conf'? y
[root@htc-mkandes config.d]# vi 20-jetstream-htcondor-execute.conf
[root@htc-mkandes config.d]# ls
20-jetstream-htcondor-execute.conf
[root@htc-mkandes config.d]# condor_store_cred -c add
Account: condor_pool@novalocal

Enter password:

Operation succeeded.
[root@htc-mkandes config.d]# systemctl restart condor
[root@htc-mkandes config.d]# systemctl status condor
● condor.service - Condor Distributed High-Throughput-Computing
  Loaded: loaded (/usr/lib/systemd/system/condor.service; enabled; vendor prese
t: disabled)
  Active: active (running) since Tue 2019-05-28 20:04:17 UTC; 10s ago
    Main PID: 565170 (condor_master)
   Status: "Problems: STARTD=STOPPED"
```

Then remove your personal HTCondor configuration file  
00-jetstream-htcondor-personal.conf and replace it with  
the new 20-jetstream-htcondor-execute.conf file.

# Reconfigure your Personal HTCondor as Execute Node (3)

```
[root@htc-mkandes config.d]# pwd
/etc/condor/config.d
[root@htc-mkandes config.d]# ls
00-jetstream-htcondor-personal.conf
[root@htc-mkandes config.d]# rm 00-jetstream-htcondor-personal.conf
rm: remove regular file '00-jetstream-htcondor-personal.conf'? y
[root@htc-mkandes config.d]# vi 20-jetstream-htcondor-execute.conf
[root@htc-mkandes config.d]# ls
20-jetstream-htcondor-execute.conf
[root@htc-mkandes config.d]# condor_store_cred -c add
Account: condor_pool@novalocal

Enter password:

Operation succeeded.
[root@htc-mkandes config.d]# systemctl restart condor
[root@htc-mkandes config.d]# systemctl status condor
● condor.service - Condor Distributed High-Throughput-Computing
    Loaded: loaded (/usr/lib/systemd/system/condor.service; enabled; vendor prese
t: disabled)
    Active: active (running) since Tue 2019-05-28 20:04:17 UTC; 10s ago
      Main PID: 565170 (condor_master)
        Status: "Problems: STARTD=STOPPED"
```

Once the new configuration file is in place, add a pool password using the `condor_store_cred` tool. For the password, you should use the dimension of the Hilbert matrix given in the OpenMP and MPI slides.

# Reconfigure your Personal HTCondor as Execute Node (4)

```
[root@htc-mkandes config.d]# pwd  
/etc/condor/config.d  
[root@htc-mkandes config.d]# ls  
00-jetstream-htcondor-personal.conf  
[root@htc-mkandes config.d]# rm 00-jetstream-htcondor-personal.conf  
rm: remove regular file '00-jetstream-htcondor-personal.conf'? y  
[root@htc-mkandes config.d]# vi 20-jetstream-htcondor-execute.conf  
[root@htc-mkandes config.d]# ls  
20-jetstream-htcondor-execute.conf  
[root@htc-mkandes config.d]# condor_store_cred -c add  
Account: condor_pool@novalocal  
  
Enter password:  
  
Operation succeeded.  
[root@htc-mkandes config.d]# systemctl restart condor  
[root@htc-mkandes config.d]# systemctl status condor  
● condor.service - Condor Distributed High-Throughput-Computing  
   Loaded: loaded (/usr/lib/systemd/system/condor.service; enabled; vendor prese  
t: disabled)  
     Active: active (running) since Tue 2019-05-28 20:04:17 UTC; 10s ago  
       Main PID: 565170 (condor_master)  
         Status: "Problems: STARTD=STOPPED"
```

With pool password set, you can then `restart condor`.

## Reconfigure your Personal HTCondor as Execute Node (5)

```
Main PID: 565170 (condor_master)
Status: "Problems: STARTD=STOPPED"
Tasks: 5 (limit: 32767)
Memory: 5.3M
CGroup: /system.slice/condor.service
└─565170 /usr/sbin/condor_master -f
    ├─565201 condor_procd -A /var/run/condor/procd_pipe -L /var/log/co...
    ├─565202 condor_shared_port -f
    ├─565203 condor_startd -f
    └─565214 kflops

May 28 20:04:17 htc-mkandes.novalocal systemd[1]: Started Condor Distributed ...
Hint: Some lines were ellipsized, use -l to show in full.
[root@htc-mkandes config.d]# condor status
Name          OpSys      Arch     State      Activity LoadAv Mem   ActvtyT
htc-mkandes.novalocal LINUX      X86_64 Unclaimed Idle      0.000 1838  0+00:00
                                         Machines Owner Claimed Unclaimed Matched Preempting Drain
                                         X86_64/LINUX      1      0      0      1      0      0      0
                                         Total      1      0      0      1      0      0      0
[root@htc-mkandes config.d]#
```

After restarting condor, your execute node should register with the pool's central manager node. Check that your execute node has registered with the central manager node by running the `condor_status` command.

# Login to Submit Node

```
mkandes@castlebravo:~$ sftp mkandes@149.165.171.19
Connected to 149.165.171.19.
sftp> put bas
bash_pi.htcondor    bash_pi.sh
sftp> put bash_pi.*
Uploading bash_pi.htcondor to /home/mkandes/bash_pi.htcondor
bash_pi.htcondor          100%   357     0.4KB/s  00:00
Uploading bash_pi.sh to /home/mkandes/bash_pi.sh
bash_pi.sh              100% 1611     1.6KB/s  00:00
sftp> exit
mkandes@castlebravo:~$ ssh mkandes@149.165.171.19
Last login: Tue May 28 23:19:54 2019 from wireless-169-228-102-18.ucsd.edu
[mkandes@htc-submit ~]$ condor_submit bash_pi.htcondor
Submitting job(s).
1 job(s) submitted to cluster 2.
[mkandes@htc-submit ~]$ condor_q

-- Schedd: htc-submit.novalocal : <10.1.1.6:9618?... @ 05/28/19 23:20:24
OWNER      BATCH_NAME      SUBMITTED      DONE      RUN      IDLE      TOTAL JOB_IDS
mkandes  ID: 2      5/28 23:20      -          1      -          1 2.0

Total for query: 1 jobs; 0 completed, 0 removed, 0 idle, 1 running, 0 held, 0 suspended
```

If your execute node registered with the central manager, try logging into the HTCondor submit node (using your XSEDE username). If successful, run at least one more `bash_pi.htcondor` job in this HTCondor pool we've built together.

## Login Problems?

If you're not able to login to the HTCondor submit node, contact me ([mkandes@sdsc.edu](mailto:mkandes@sdsc.edu)). I'll configure your user account on the submit node for the pool.

# Questions?

