

How to use NGC Containers on AWS

Scott Ellis & Jeff Weiss GTC 2018 S8276



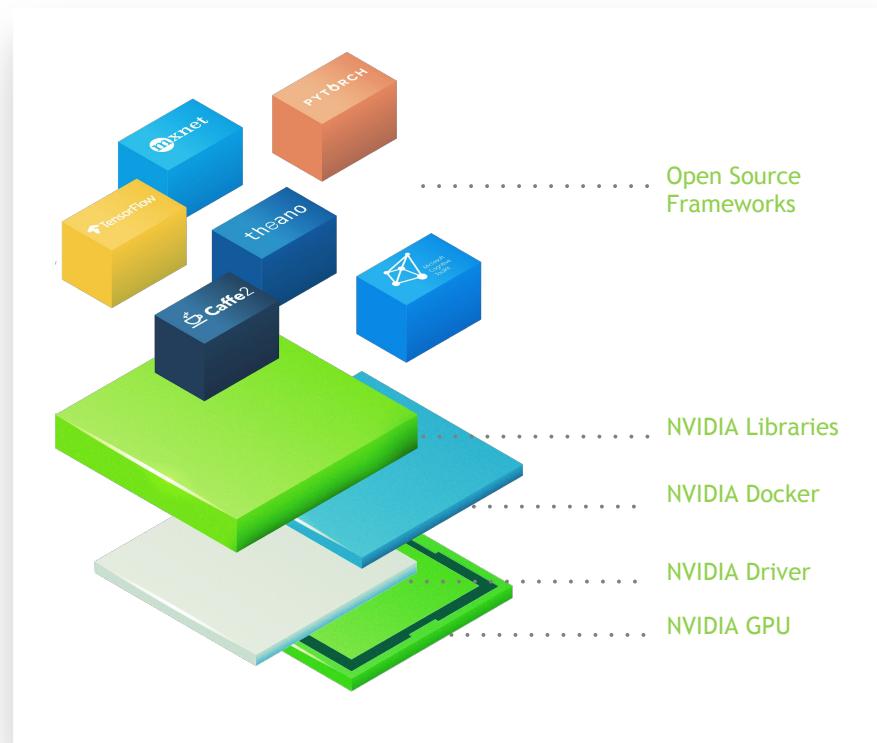
What is NGC? Why is NGC?

Challenges with Complex Software

Current DIY GPU-accelerated AI and HPC deployments are **complex** and **time consuming** to build, test and maintain

Development of software frameworks by the community is moving **very fast**

Requires high level of **expertise** to manage driver, library, framework dependencies



GPU-Accelerated Deep Learning Containers

Deep Learning Everywhere, for Everyone

Innovation for Every Industry

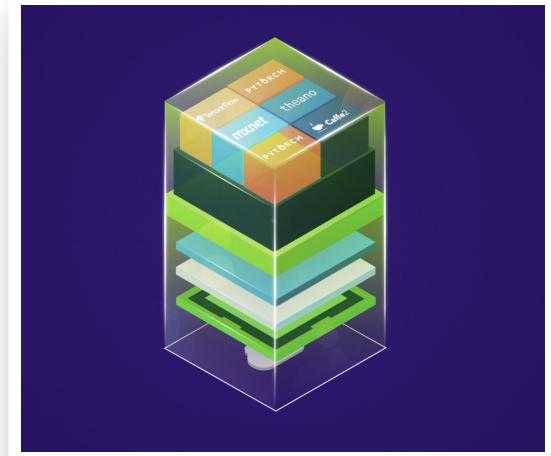
Quickly tap into the power of NVIDIA AI, from automotive, to healthcare, to fintech, and more

Say Goodbye to DIY

Deep learning software containers, tuned, tested, and certified by NVIDIA

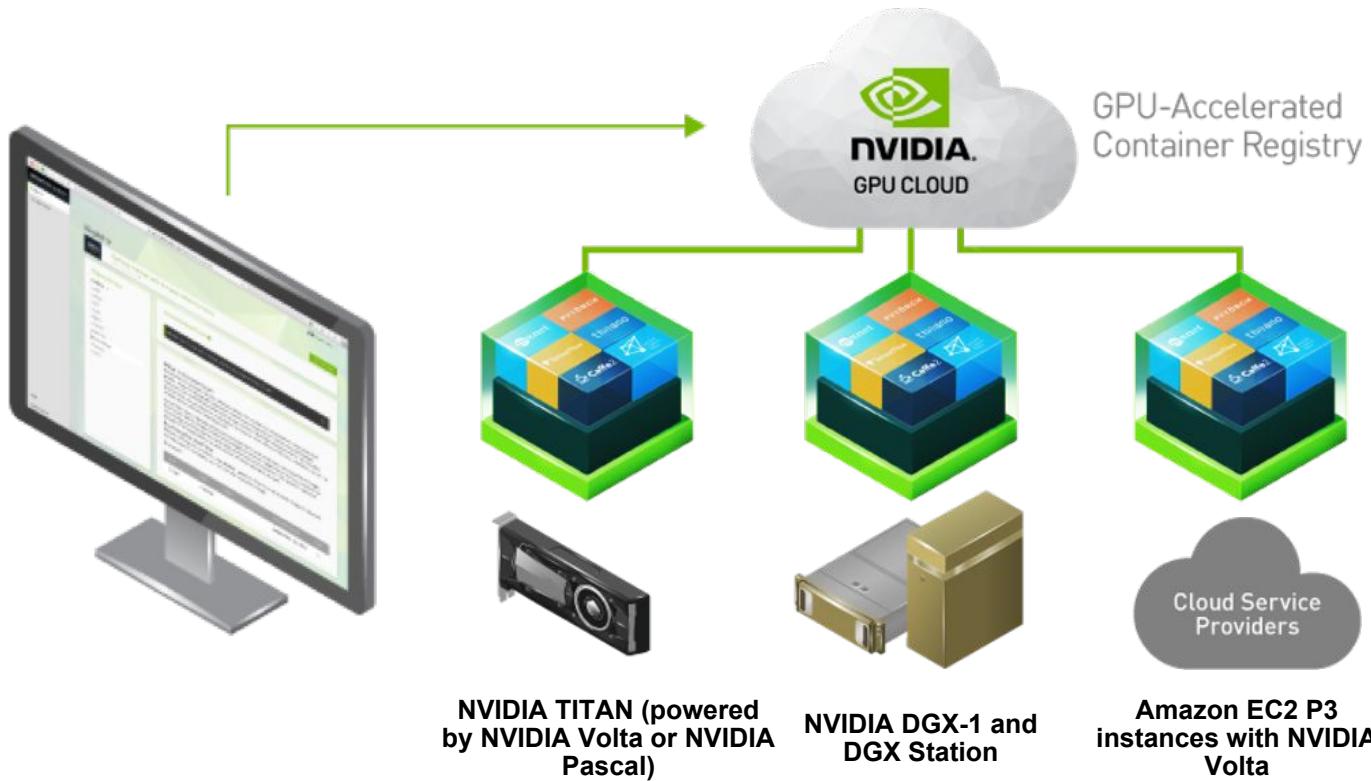
Stay Up To Date

Monthly updates to deep learning containers



NVIDIA GPU Cloud integrates GPU-optimized deep learning frameworks, runtimes, libraries, and OS into a ready-to-run container, available at no charge

Deep Learning Across Platforms



Three Steps To Deep Learning On Amazon EC2 with NGC

SIGN UP

To get an NGC account, go to:

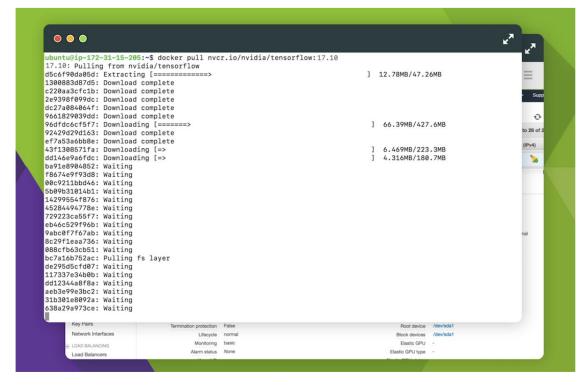
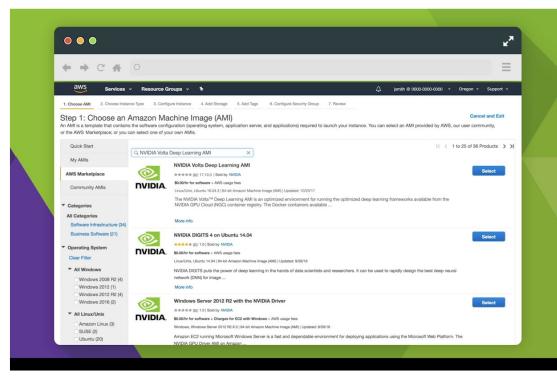
www.nvidia.com/ngcsignup

DEPLOY IMAGE

On Amazon EC2, choose a P3 instance and deploy the **NVIDIA Volta Deep Learning AMI for NGC**

PULL CONTAINER

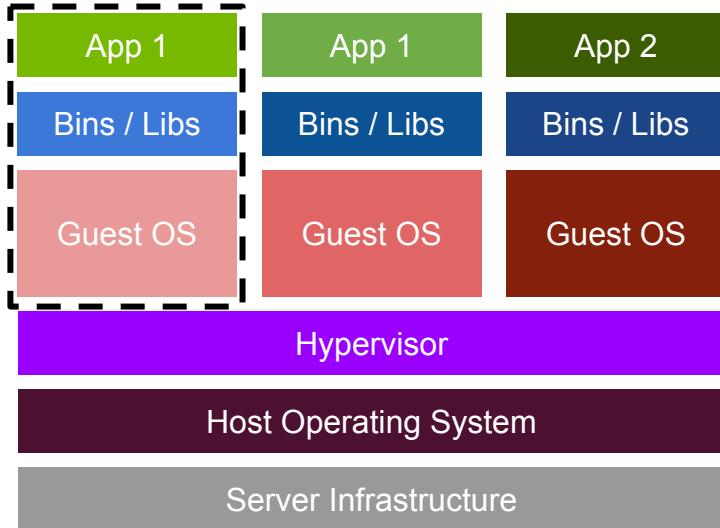
Pick your desired framework (TensorFlow, PyTorch, MXNet, etc.), and pull the container into your instance



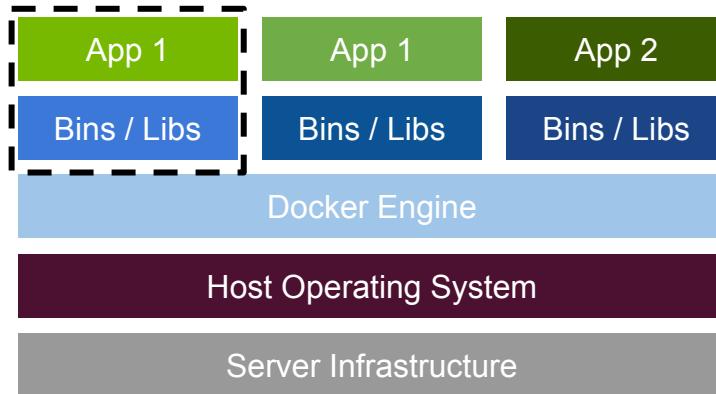
What is this container thing you speak of?

Virtual Machine vs. Container

Not so similar

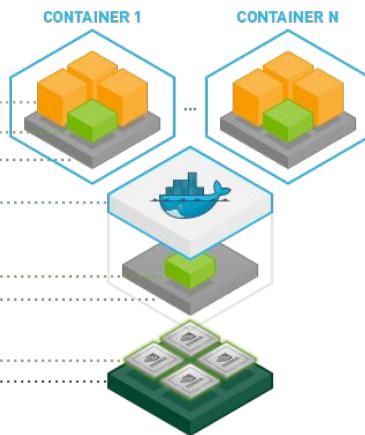


Virtual Machines



Containers

NVIDIA Container Runtime



- Colloquially called “nvidia-docker”
- Docker containers are *hardware-agnostic* and *platform-agnostic*
- NVIDIA GPUs are specialized hardware that require the NVIDIA driver
- Docker does not natively support NVIDIA GPUs with containers
- NVIDIA Container Runtime makes the images agnostic of the NVIDIA driver
 - ◆ Required character devices and driver files are mounted when starting the container on the target machine
 - ◆ This makes Docker images portable while still leveraging NVIDIA GPUs

<https://github.com/NVIDIA/nvidia-docker>

Docker Terms

Definitions

Image

Docker images are the basis of [containers](#). An Image is an ordered collection of root filesystem changes and the corresponding execution parameters for use within a container runtime. An image typically contains a union of layered filesystems stacked on top of each other. An image does not have state and it never changes.

Container

A container is a runtime instance of a [docker image](#).

A Docker container consists of

- A Docker image
- Execution environment
- A standard set of instructions

<https://docs.docker.com/engine/reference/glossary/>

Managing Images and Containers

Common Commands

List Images:

```
docker images
```

Remove an Image:

```
docker rmi imageID
```

```
docker rmi tensorRT
```

List Containers:

```
docker ps -a
```

Stop a running Container:

```
docker stop containerID
```

Remove a Container:

```
docker rm containerID
```

Note: imageID and containerID can be either a hash or a name

Running Containers

docker run and option

docker run Options

- `--runtime=nvidia` enable GPU capabilities
- `--rm` remove the container after it exits
- `-i -t` or `-it` interactive, and connect a "tty"
- `-d --detach` run in the background
- `--name` give the container a name
- `-p 8080:80` port map from host to container
- `-v ~/data:/data` map storage volume from host to container (bind mount) i.e. bind the `~data` directory in your home directory to `/data` in the container

Starts Tensorflow with ports, volumes, and console (All 1 line):

```
docker run
```

```
--runtime=nvidia  
--rm -it  
--name MyCoolContainer  
-p 8888:80  
-v ~/data:/data  
nvcr.io/nvidia/tensorflow:18.01-py2  
examples/nvcnn.py
```

Navigating the NGC WebUI

NVIDIA GPU Cloud

How do we actually use it?

Our challenge:

- Sign up for a *free* NGC account at www.nvidia.com/ngcsignup
- Login to the WebUI
- Generate an API key for Docker to use

What is an API Key?

(And why do you need one?)

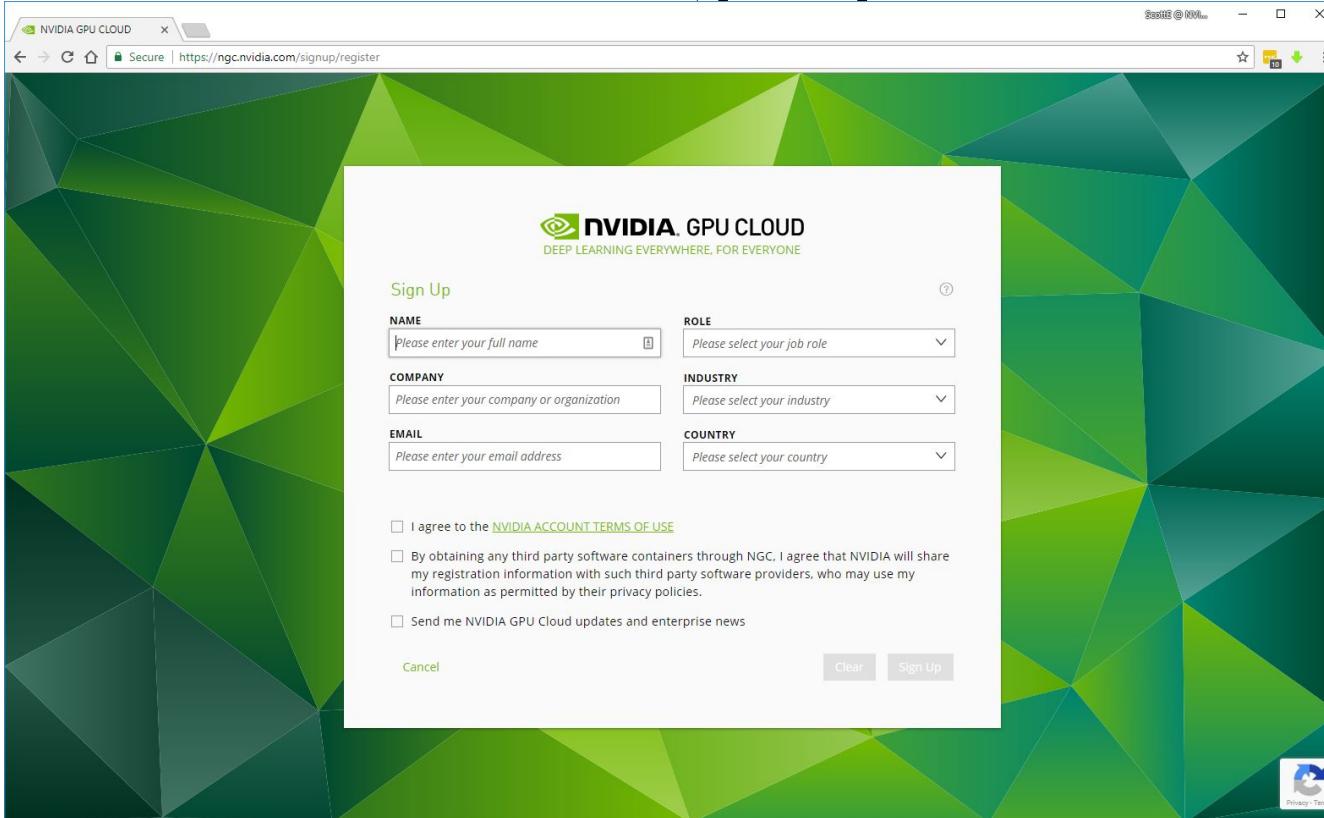
Your API key represents your credentials

- Used for programmatic interaction (e.g., docker, REST API, etc.)
- Uniquely identifies you (think “Username & Password”)
- There can be only one (regenerating your API key invalidates the old one)

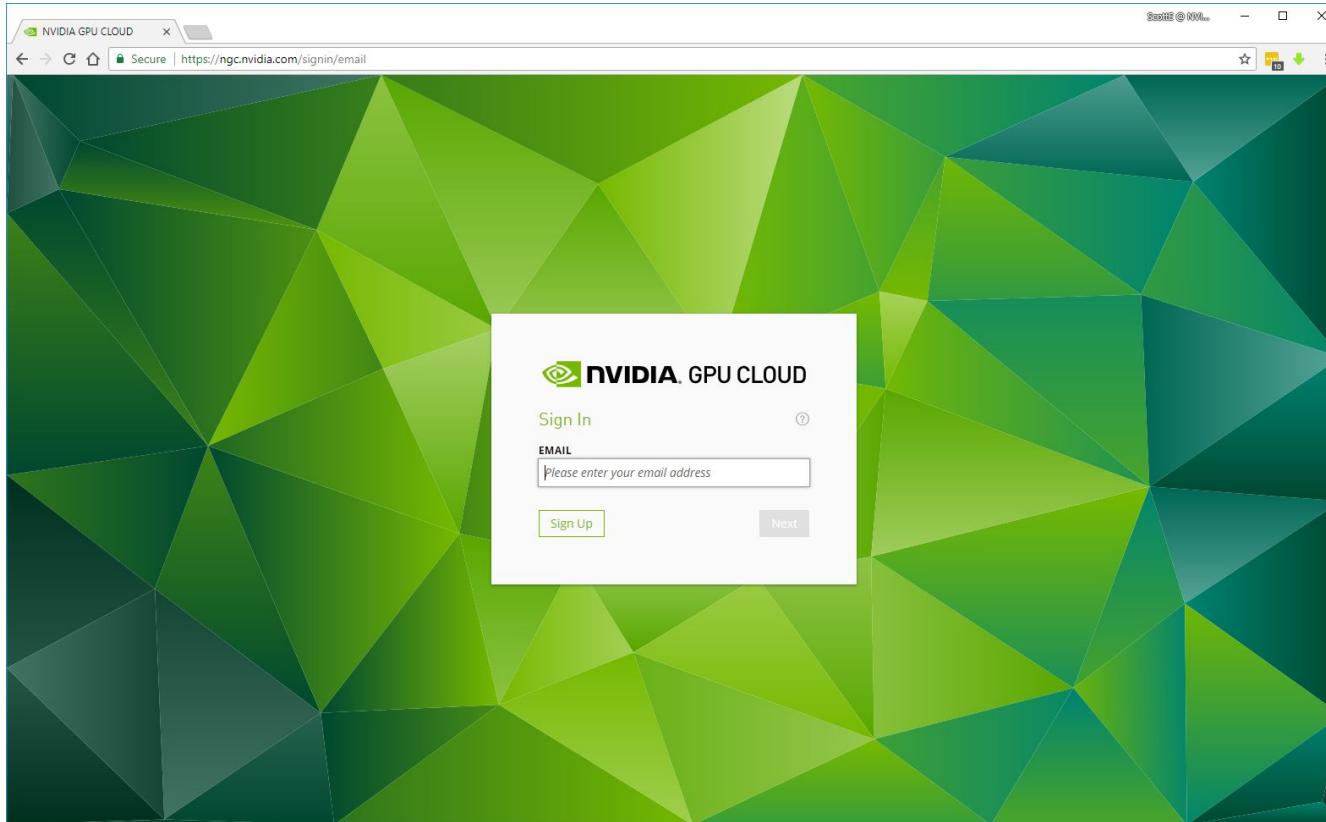
WebUI at ngc.nvidia.com: Use Username & Password

Programmatic interface at nvcr.io: Use API Key

NGC Sign-up



NGC Access

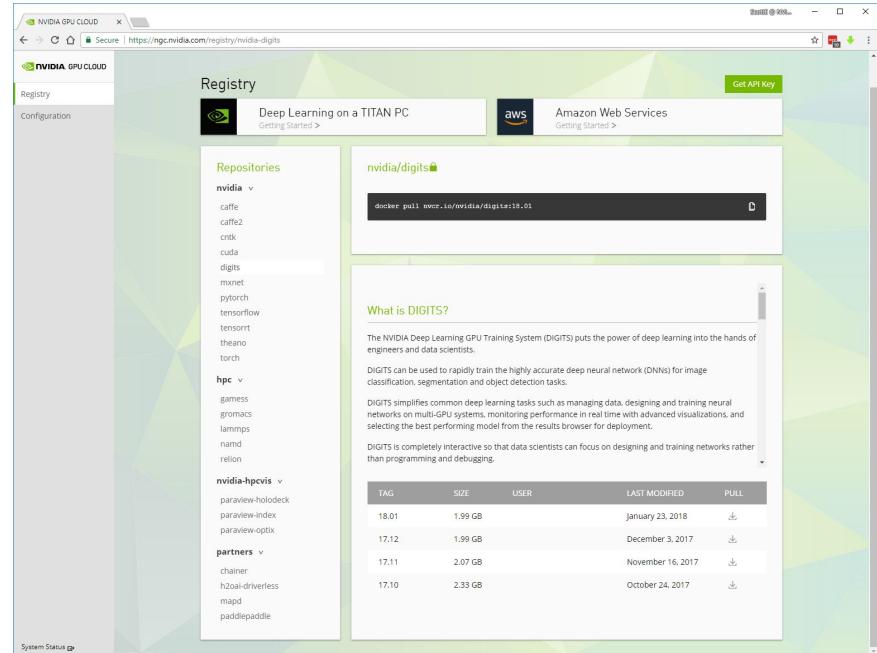


NGC WebUI

Where it all begins

When you login...

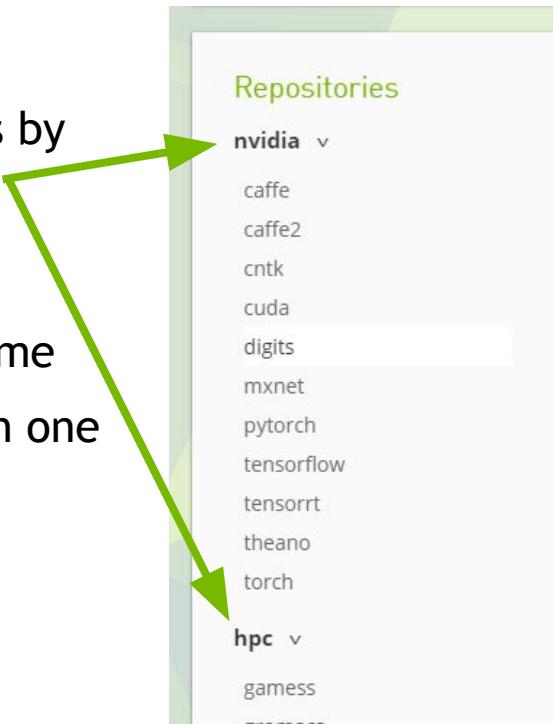
- How-to Pages
- Repository List
- Instructions and Info
- Image specifics
- Docker pull shortcut



NGC WebUI

Repository List

- **Repositories**
 - Used to group containers by functionality
- **Containers**
 - Descriptive container name
 - Details when you click on one



NGC WebUI

Instructions and Information

When you select a container

- Description on what the image contains
- Usually examples on running it
- Often has links for more information and tutorials

The screenshot shows a web browser window with a light gray header bar. The main content area has a white background with a thin green border on the left. At the top, there is a green header section containing the title "What is DIGITS?". Below this, there is a horizontal line and some descriptive text. A vertical scroll bar is visible on the right side of the content area.

What is DIGITS?

The NVIDIA Deep Learning GPU Training System (DIGITS) puts the power of deep learning into the hands of engineers and data scientists.

DIGITS can be used to rapidly train the highly accurate deep neural network (DNNs) for image classification, segmentation and object detection tasks.

DIGITS simplifies common deep learning tasks such as managing data, designing and training neural networks on multi-GPU systems, monitoring performance in real time with advanced visualizations, and selecting the best performing model from the results browser for deployment.

DIGITS is completely interactive so that data scientists can focus on designing and training networks rather than programming and debugging.

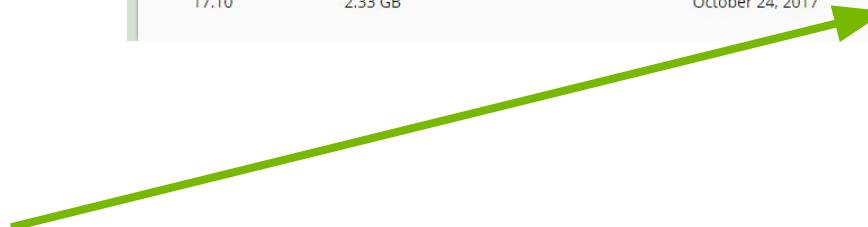
NGC WebUI

Image Specifics

List of images for that container

- **Tag (nvidia/caffe:18.01)**
 - Follows YY.MM format
- **Creation date**
 - Updated monthly
- **Shortcut to copy docker pull command to clipboard**

TAG	SIZE	USER	LAST MODIFIED	PULL
18.01	1.99 GB		January 23, 2018	
17.12	1.99 GB		December 3, 2017	
17.11	2.07 GB		November 16, 2017	
17.10	2.33 GB		October 24, 2017	

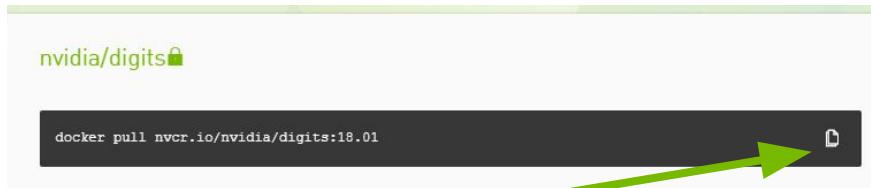


NGC WebUI

Docker pull shortcut

Shortcut to the latest at the top

- Shows full image name
(nvcr.io/nvidia/digits:18.01)
- Icon to copy to clipboard
 - Same as in image details

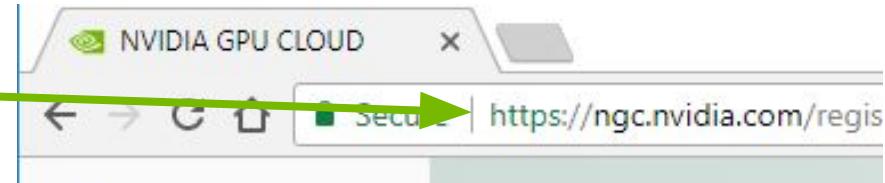


NGC vs. NVCR

Why are there two FQDNs?

ngc.nvidia.com

- NGC = NVIDIA GPU Cloud
- Used for administrative tasks
 - (a.k.a. The WebUI)



nvcr.io

- NVIDIA Container Repository
- Used for Docker tasks

```
docker pull nvcr.io/nvidia/digits:18.01
```

NGC API Key

The screenshot shows the 'Configuration > API Key' page of the NVIDIA GPU CLOUD interface. The left sidebar has 'Registry' and 'Configuration' tabs, with 'Configuration' selected. The main content area is titled 'API' and contains sections for 'API Information', 'Usage', and 'Docker™'. A green 'Generate API Key' button is highlighted with a red box. At the bottom, there's a terminal-style code block showing Docker login command syntax.

Secure | https://ngc.nvidia.com/configuration/api-key

Scott Ellis

Configuration > API Key

Generate API Key

API

API Information

Your API Key authenticates your use of NGC service when using NGC CLI or the Docker client. Anyone with this API Key has access to all services, actions, and resources on your behalf.

Click Generate API Key to create your own API Key. If you have forgotten or lost your API Key, you can come back to this page to create a new one at any time.

Usage

Use your API key to log in to the NGC registry as follows.

Docker™

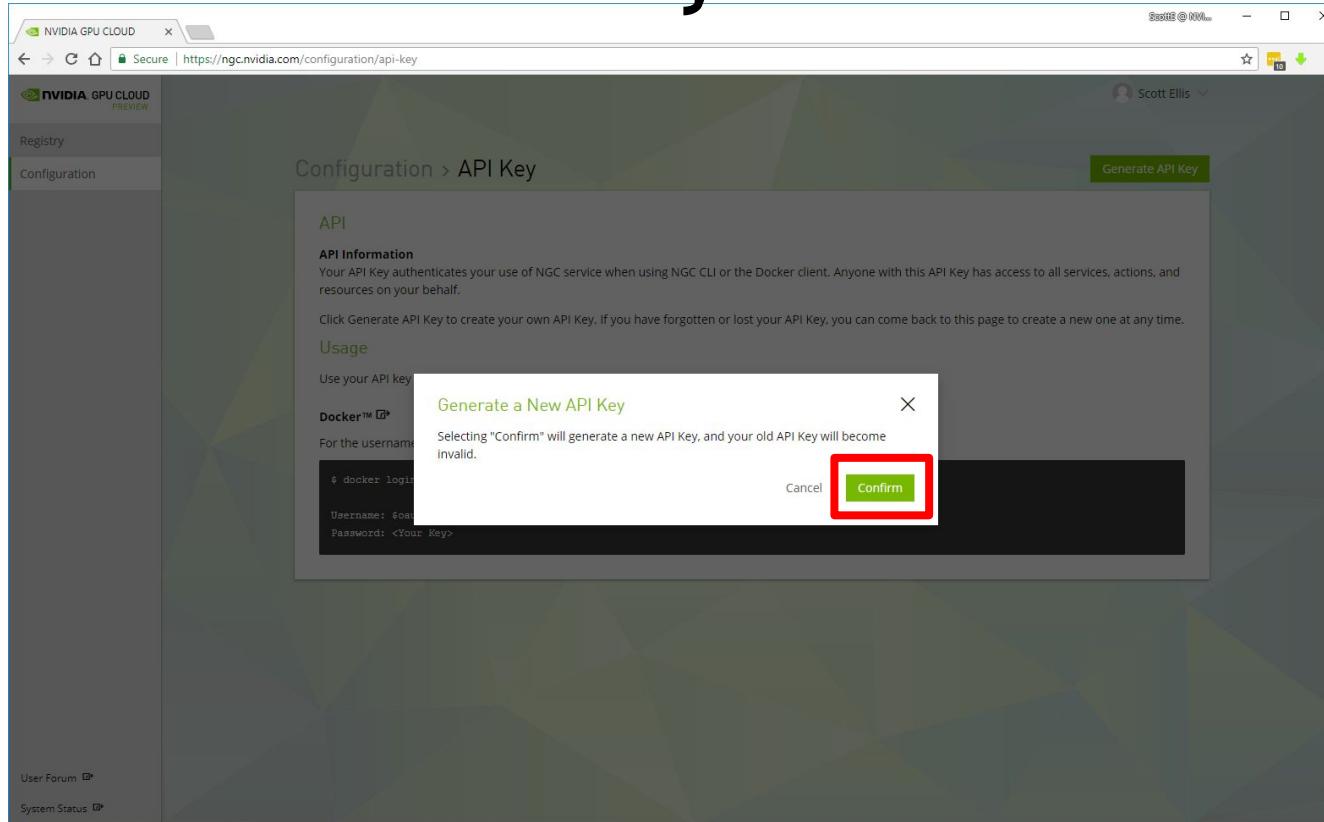
For the username, enter '\$oauthtoken' exactly as shown. It is a special authentication token for all users.

```
$ docker login nvcr.io
Username: $oauthtoken
Password: <Your Key>
```

User Forum

System Status

NGC API Key Generate



NGC API Key Save

The screenshot shows the "Configuration > API Key" page of the NVIDIA GPU CLOUD PREVIEW website. The left sidebar has "Configuration" selected. The main content area is titled "API". It contains sections for "API Information" (describing the API key's purpose), "Usage" (instructions for logging in to the NGC registry), and "Docker™" (instructions for using Docker). A "Generate API Key" button is located at the top right of the main content area. Below it, a message states: "API Key generated successfully. This is the only time your API Key will be displayed. Keep your API Key secret." A red box highlights the generated API key: "a2VmckWkxZG1iNHE2amMyNj1zZGY2MXUwOWM6Zjg4ZGQzMGMtNGViYy0ONTAyLThhY2EtMmQ3ODIjMWFhMmQ4". At the bottom left, there are links for "User Forum" and "System Status".

AWS Account Setup

Execution Environment

How does Amazon figure into this?

AWS is where we'll run NGC containers

...could also be your NVIDIA Titan-based workstation

...could also be your DGX server or workstation

...could also be almost anywhere with a GPU and docker

AWS

Getting Started

Before launching an Amazon instance (for the first time) we need to:

- Secure login access to our instance with an SSH key
- Secure network access with a set of connection rules

AWS

Getting Started

Our challenge:

- Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>
- [Create Key Pair](#)
- [Create Security Group](#)

If AWS is managed for you, this might not be visible to you

(We'll use the WebUI, but there are hundreds of other ways to do this!)

AWS Account Setup

The screenshot shows the AWS Management Console home page. A red box highlights the "Services" dropdown menu in the top navigation bar. Another red box highlights the "Oregon" region selection in the top right corner. The main content area displays various service links and promotional sections like "Helpful tips" and "Explore AWS".

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services: EC2, EFS, Storage Gateway

All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine (With EC2 -2-3 minutes)

Build a web app (With Elastic Beanstalk ~6 minutes)

Build using virtual servers (With Lightsail ~1-2 minutes)

Connect an IoT device (With AWS IoT ~5 minutes)

Start a development project (With CodeStar ~5 minutes)

Register a domain (With Route 53 ~3 minutes)

See more

Learn to build

Learn to deploy your solutions through step-by-step guides, labs, and videos.

See all

Websites: 3 videos, 3 tutorials, 3 labs

DevOps: 6 videos, 2 tutorials, 3 labs

Backup and recovery: 3 videos, 2 tutorials, 3 labs

Helpful tips

Manage your costs: Get real-time billing alerts based on your cost and usage budgets. [Start now](#)

Create an organization: Use AWS Organizations for policy-based management of multiple AWS accounts. [Start now](#)

Explore AWS

Amazon Relational Database Service (RDS): RDS manages and scales your database for you. RDS supports Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. [Learn more](#)

Real-Time Analytics with Amazon Kinesis: Stream and analyze real-time data, so you can get timely insights and react quickly. [Learn more](#)

Get Started with Containers on AWS: Amazon ECS helps you build and scale containers for any size application. [Learn more](#)

AWS Marketplace: Discover, procure, and deploy popular software products that run on AWS. [Learn more](#)

AWS Account Setup

The screenshot shows the AWS Management Console home page. The top navigation bar includes the AWS logo, a 'Services' dropdown, a 'Resource Groups' dropdown, user information (scotta @ 6115-2050-7156, Oregon), and a 'Support' link. A search bar at the top center contains the placeholder text 'Find a service by name or feature (for example, EC2, S3 or VM, storage.)'. Below the search bar is a grid of service categories and their sub-services. A red box highlights the 'EC2' service under the 'Compute' category. Other visible categories include Developer Tools, Machine Learning, AR & VR, Application Integration, Customer Engagement, Business Productivity, and Desktop & App Streaming.

Category	Sub-Services
Compute	EC2, Lambda, Batch, Elastic Beanstalk
Storage	S3, EFS, Glacier, Storage Gateway
Database	Relational Database Service, DynamoDB, ElastiCache, Amazon Redshift
Migration	AWS Migration Hub, Application Discovery Service, Database Migration Service, Server Migration Service, Snowball
Developer Tools	CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, X-Ray
Machine Learning	Amazon SageMaker, Amazon Comprehend, AWS DeepLens, Amazon Lex, Machine Learning, Amazon Polly, Rekognition, Amazon Transcribe, Amazon Translate
AR & VR	Amazon Sumerian
Application Integration	Step Functions, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF
Customer Engagement	Amazon Connect, Pinpoint, Simple Email Service
Business Productivity	Alexa for Business, Amazon Chime, WorkDocs, WorkMail
Media Services	Elastic Transcoder, Kinesis Video Streams, MediaConvert, MediaLive, MediaPackage, MediaStore
Security, Identity & Compliance	IAM, Cognito, GuardDuty, Inspector, Amazon Macie, AWS Single Sign-On
Desktop & App Streaming	WorkSpaces, AppStream 2.0

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. On the left, a sidebar menu lists various services: EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security, Security Groups, Elastic IPs, Key Pairs (which is highlighted with a red box), Load Balancing, Target Groups, and Auto Scaling.

The main content area displays the following information:

- Resources**: You are using the following Amazon EC2 resources in the US West (Oregon) region:
 - 2 Running Instances
 - 5 Elastic IPs
 - 0 Dedicated Hosts
 - 3 Snapshots
 - 18 Volumes
 - 0 Load Balancers
 - 50 Key Pairs
 - 92 Security Groups
 - 0 Placement Groups
- Create Instance**: A button labeled "Launch Instance".
- Service Health**:
 - Service Status:** US West (Oregon): This service is operating normally.
 - Availability Zone Status:**
 - us-west-2a: Availability zone is operating normally.
 - us-west-2b: Availability zone is operating normally.
 - us-west-2c: Availability zone is operating normally.
- Scheduled Events**: US West (Oregon): No events.
- Account Attributes**:
 - Supported Platforms: VPC
 - Default VPC: vpc-bffdd9d9
 - Resource ID length management
- Additional Information**:
 - Getting Started Guide
 - Documentation
 - All EC2 Resources
 - Forums
 - Pricing
 - Contact Us
- AWS Marketplace**: Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular AMIs:
 - Barracuda CloudGen Firewall for AWS - PAYG
 - Matillion ETL for Snowflake

At the bottom, there are links for Feedback, English (US), Copyright notice (© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, and Terms of Use.

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. The left sidebar navigation menu is visible, with the 'Key Pairs' option highlighted under the 'Services' category. In the main content area, there is a 'Create Key Pair' button, which is highlighted with a red box. Below it is a table header with columns for 'Key pair name' and 'Fingerprint'. A search bar at the top of the table allows filtering by keyword. The message 'Select a key pair' is displayed below the table. The bottom of the screen includes standard AWS footer links for Feedback, Language selection (English (US)), and legal notices.

AWS Account Setup

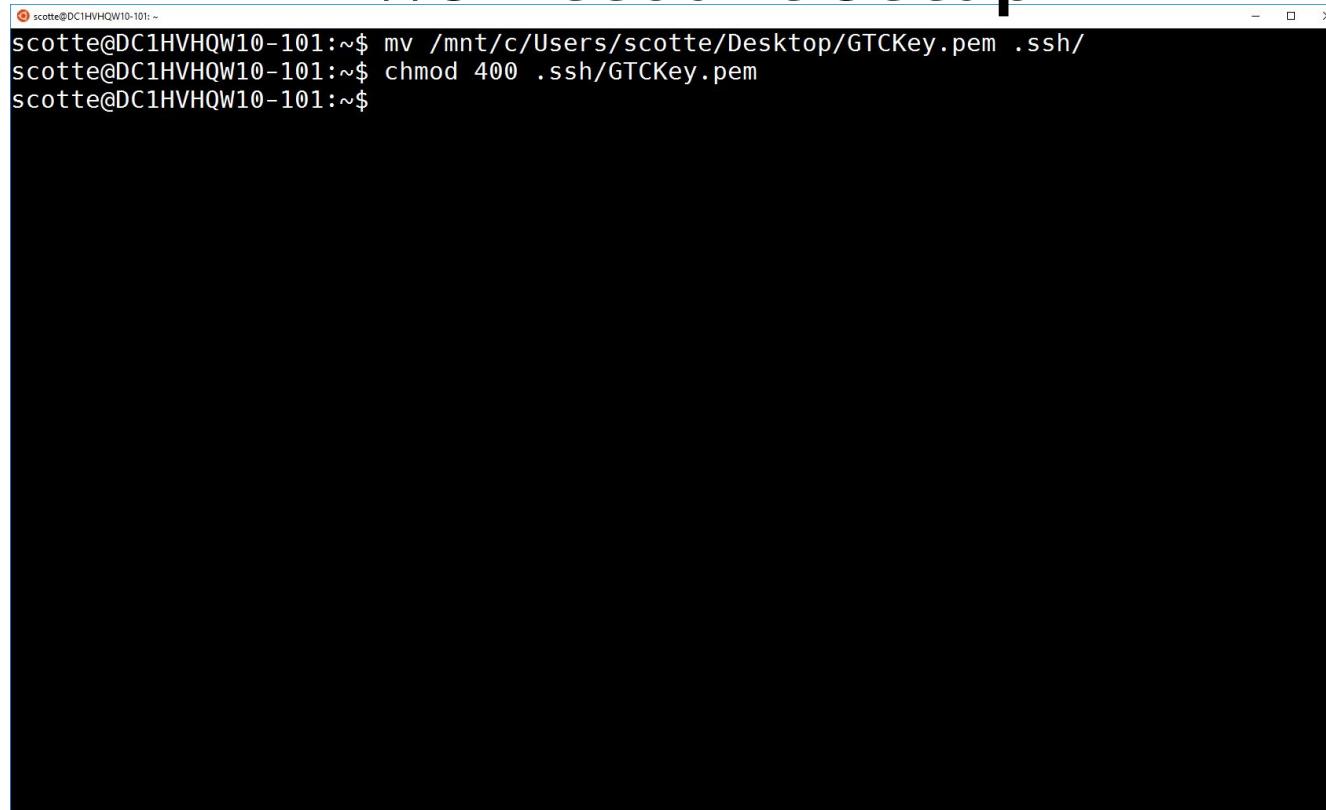
The screenshot shows the AWS EC2 Management Console interface. On the left, a sidebar lists various services: EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, AMIs, Bundle Tasks, Volumes, Snapshots, Security Groups, Elastic IPs, Placement Groups, Key Pairs (which is selected and highlighted in orange), and Load Balancers. The main content area shows a list of existing key pairs, with a search bar at the top. A modal dialog box titled "Create Key Pair" is open in the center, prompting for a "Key pair name" (with "GTCKey" entered) and providing "Cancel" and "Create" buttons. The "Create" button is highlighted with a red box. The status bar at the bottom indicates the URL as https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#KeyPairs:sort=keyName.

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. The left sidebar navigation menu includes categories like EC2 Dashboard, Instances, Images, Elastic Block Store, Network & Security, Key Pairs (which is currently selected), and Load Balancing. The main content area displays a table of key pairs. A search bar at the top of the table allows filtering by name or fingerprint. One entry is visible: "GTCKey" with the fingerprint "36 4d ea ef 58 0c 9e d1 56 63 fc ce 1f 1f d1 cc 75 07 dc 59". Below the table, there is a section titled "Select a key pair" with three small icons.

Key pair name	Fingerprint
GTCKey	36 4d ea ef 58 0c 9e d1 56 63 fc ce 1f 1f d1 cc 75 07 dc 59

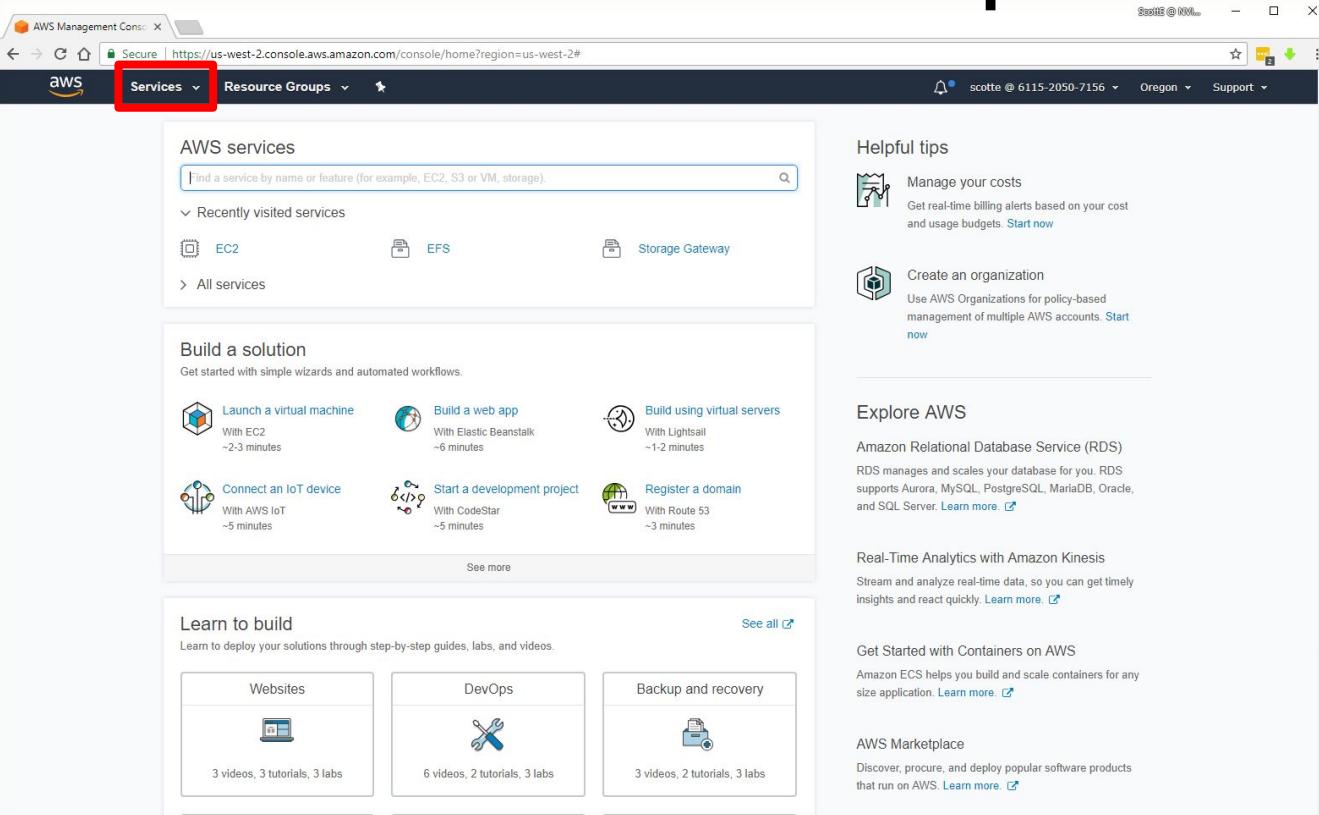
AWS Account Setup



The image shows a terminal window with a black background and white text. At the top left, there is a small red circular icon with a white question mark. The window title bar contains the text "scotte@DC1HVHQW10-101: ~". The terminal prompt is "\$ ". Below the prompt, three commands are displayed:

```
scotte@DC1HVHQW10-101:~$ mv /mnt/c/Users/scotte/Desktop/GTCKey.pem .ssh/
scotte@DC1HVHQW10-101:~$ chmod 400 .ssh/GTCKey.pem
scotte@DC1HVHQW10-101:~$
```

AWS Account Setup



The screenshot shows the AWS Management Console home page. A red box highlights the "Services" dropdown menu in the top navigation bar. The main content area displays various service categories and quick-start options:

- AWS services:** A search bar and a list of recently visited services (EC2, EFS, Storage Gateway) with a "All services" link.
- Build a solution:** Simple wizards for building solutions like virtual machines, web apps, or databases.
- Learn to build:** Step-by-step guides for Websites, DevOps, and Backup and recovery.
- Helpful tips:** Links to "Manage your costs" and "Create an organization".
- Explore AWS:** Information on Amazon RDS, Amazon Kinesis, and AWS Marketplace.

AWS Account Setup

The screenshot shows the AWS Management Console home page. The top navigation bar includes the AWS logo, a 'Services' dropdown, a 'Resource Groups' dropdown, user information (scotta @ 6115-2050-7156, Oregon), and a 'Support' link. A search bar at the top center contains the placeholder text 'Find a service by name or feature (for example, EC2, S3 or VM, storage.)'. Below the search bar is a grid of service categories and their sub-services. The 'EC2' service is highlighted with a red rectangular box. Other visible services include Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, X-Ray), Machine Learning (Amazon SageMaker, Amazon Comprehend, AWS DeepLens, Amazon Lex, Machine Learning, Amazon Polly, Rekognition, Amazon Transcribe, Amazon Translate), AR & VR (Amazon Sumerian), Application Integration (Step Functions, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF), Customer Engagement (Amazon Connect, Pinpoint, Simple Email Service), Business Productivity (Alexa for Business, Amazon Chime, WorkDocs, WorkMail), Desktop & App Streaming (WorkSpaces, AppStream 2.0), and various Storage, Database, Migration, Media Services, and Security, Identity & Compliance services.

AWS Management Console | https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2#

Services ▾ Resource Groups ▾

scotta @ 6115-2050-7156 Oregon Support

History

Console Home

EC2

EFS

Storage Gateway

Find a service by name or feature (for example, EC2, S3 or VM, storage.)

Group A-Z

EC2

Developer Tools

- CodeStar
- CodeCommit
- CodeBuild
- CodeDeploy
- CodePipeline
- Cloud9
- X-Ray

Machine Learning

- Amazon SageMaker
- Amazon Comprehend
- AWS DeepLens
- Amazon Lex
- Machine Learning
- Amazon Polly
- Rekognition
- Amazon Transcribe
- Amazon Translate

AR & VR

- Amazon Sumerian

Application Integration

- Step Functions
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

Customer Engagement

- Amazon Connect
- Pinpoint
- Simple Email Service

Business Productivity

- Alexa for Business
- Amazon Chime
- WorkDocs
- WorkMail

Desktop & App Streaming

- WorkSpaces
- AppStream 2.0

Storage

- S3
- EFS
- Glacier
- Storage Gateway

Management Tools

- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks

Analytics

- Athena
- EMR
- CloudSearch
- Elasticsearch Service
- Kinesis
- QuickSight
- Data Pipeline
- AWS Glue

Database

- Relational Database Service
- DynamoDB
- ElastiCache
- Amazon Redshift

Migration

- AWS Migration Hub
- Application Discovery Service
- Database Migration Service
- Server Migration Service
- Snowball

Media Services

- Elastic Transcoder
- Kinesis Video Streams
- MediaConvert
- MediaLive
- MediaPackage
- MediaStore

Security, Identity & Compliance

- IAM
- Cognito
- GuardDuty
- Inspector
- Amazon Macie
- AWS Single Sign-On

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. The left sidebar lists various services: EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security (with 'Security Groups' highlighted by a red box), Load Balancing, and Auto Scaling.

The main content area displays the 'Resources' section, which shows the following Amazon EC2 resources in the US West (Oregon) region:

Category	Count
Running Instances	2
Dedicated Hosts	0
Volumes	18
Key Pairs	50
Placement Groups	0
Elastic IPs	5
Snapshots	3
Load Balancers	0
Security Groups	92

Below this, there's a 'Create Instance' section with a 'Launch Instance' button and a note about launching instances in the US West (Oregon) region. The 'Service Health' section shows the status of various services, including 'US West (Oregon)' which is operating normally. The 'Scheduled Events' section indicates 'No events'.

The right side of the screen contains 'Account Attributes' (Supported Platforms: VPC, Default VPC: vpc-bffdd9d9), 'Resource ID length management', 'Additional Information' (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us), and sections for 'AWS Marketplace' (Barracuda CloudGen Firewall for AWS - PAYG, Matillion ETL for Snowflake).

At the bottom, there are links for Feedback, Language selection (English (US)), and legal notices: © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, and Terms of Use.

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. The left sidebar navigation menu is visible, with the 'Security Groups' option under the 'NETWORK & SECURITY' section highlighted. A red box highlights the 'Create Security Group' button located at the top center of the main content area. The main content area displays a table header for security groups, including columns for Name, Group ID, Group Name, VPC ID, and Description. Below the header, a message reads 'Select a security group above'. The browser address bar shows the URL: <https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#SecurityGroups:sort=vpcl>.

AWS Account Setup

The screenshot shows the AWS EC2 Management Console with the 'Create Security Group' dialog open. The left sidebar shows various AWS services like EC2 Dashboard, Instances, and Security Groups. The main area displays the 'Create Security Group' form. The security group is named 'GTCSecurityGroup' with a description 'GTCSecurityGroup'. It is associated with VPC 'vpc-bffdd9d9 (default)'. Under the 'Inbound' tab, three rules are defined:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	SSH
Custom TCP	TCP	6006	Anywhere	TensorBoard
Custom TCP	TCP	8888	Anywhere	Jupyter Notebook

A large green speech bubble with the text 'Danger Danger! Use your local subnet!' is overlaid on the right side of the dialog. A red box highlights the 'Create' button at the bottom right of the dialog.

Danger Danger!
Use your local subnet!

Security group name: GTCSecurityGroup
Description: GTCSecurityGroup
VPC: vpc-bffdd9d9 (default)

Inbound

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	SSH
Custom TCP	TCP	6006	Anywhere	TensorBoard
Custom TCP	TCP	8888	Anywhere	Jupyter Notebook

Add Rule

Create

AWS Account Setup

The screenshot shows the AWS EC2 Management Console interface. The left sidebar navigation menu includes links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security (with Security Groups selected), Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, and Target Groups. The main content area displays a table of security groups. A search bar at the top of the table allows filtering by tags and attributes or searching by keyword. The table columns are Name, Group ID, Group Name, VPC ID, and Description. One row is highlighted, showing the details for the security group 'sg-02eb047c' which is named 'GTCSecurityGroup' and belongs to VPC 'vpc-bffdd9d9'. Below the table, a section titled 'Security Group: sg-02eb047c' provides a summary of the group's configuration. It includes tabs for Description, Inbound, Outbound, and Tags. The 'Description' tab shows the group name 'GTCSecurityGroup' and group ID 'sg-02eb047c'. The 'Tags' tab shows the VPC ID 'vpc-bffdd9d9'. The 'Inbound' and 'Outbound' tabs are currently inactive.

Name	Group ID	Group Name	VPC ID	Description
sg-02eb047c		GTCSecurityGroup	vpc-bffdd9d9	GTCSecurityGroup

Security Group: sg-02eb047c

Description

Inbound

Outbound

Tags

Group name: GTCSecurityGroup
Group ID: sg-02eb047c

Group description: GTCSecurityGroup
VPC ID: vpc-bffdd9d9

AWS Launching an Instance

AWS AMI and Instances

What is an AMI? What is an Instance?

- Amazon Machine Instance (AMI)
 - Software to run in a Virtual Machine on AWS
 - Runs on an AWS EC2 Instance
- Instance
 - Hardware for running an AMI
 - Specifies CPU, GPU, RAM, Storage, Network, ...

Instance Size	GPUs - Tesla V100	GPU Peer to Peer	GPU Memory (GB)	vCPUs	Memory (GB)	Network Bandwidth	EBS Bandwidth
p3.2xlarge	1	N/A	16	8	61	Up to 10 Gbps	1.5 Gbps

AWS AMI and Instances

What does NGC provide?

NGC Containers are best on...

- NVIDIA Volta Deep Learning AMI
 - Machine image with docker configured for GPUs
 - Automatic login to NGC on launch
 - Works with P3 instances (1x, 2x, or 8x Volta)

Details: <http://docs.nvidia.com/ngc/ngc-ami-release-notes/>

AWS AMI and Instances

Let's do this

Our challenge:

- Launch an [NVIDIA Volta Deep Learning AMI](#)
- SSH into the instance
- Authenticate with NGC

AWS Instance Launching

The screenshot shows the AWS Management Console home page. A red box highlights the 'Services' button in the top navigation bar. The page displays various AWS services and solutions, including EC2, EFS, Storage Gateway, and several build solutions like launching a virtual machine or connecting an IoT device. It also features sections for 'Helpful tips', 'Explore AWS', and 'Learn to build'.

AWS services

- EC2
- EFS
- Storage Gateway

Build a solution

- Launch a virtual machine
With EC2
~2-3 minutes
- Build a web app
With Elastic Beanstalk
~6 minutes
- Build using virtual servers
With Lightsail
~1-2 minutes
- Connect an IoT device
With AWS IoT
~5 minutes
- Start a development project
With CodeStar
~5 minutes
- Register a domain
With Route 53
~3 minutes

Learn to build

- Websites: 3 videos, 2 tutorials, 3 labs
- DevOps: 6 videos, 2 tutorials, 3 labs
- Backup and recovery: 3 videos, 2 tutorials, 3 labs

Helpful tips

- Manage your costs: Get real-time billing alerts based on your cost and usage budgets. [Start now](#)
- Create an organization: Use AWS Organizations for policy-based management of multiple AWS accounts. [Start now](#)

Explore AWS

- Amazon Relational Database Service (RDS): RDS manages and scales your database for you. RDS supports Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. [Learn more](#)
- Real-Time Analytics with Amazon Kinesis: Stream and analyze real-time data, so you can get timely insights and react quickly. [Learn more](#)
- Get Started with Containers on AWS: Amazon ECS helps you build and scale containers for any size application. [Learn more](#)
- AWS Marketplace: Discover, procure, and deploy popular software products that run on AWS. [Learn more](#)

AWS Instance Launching

The screenshot shows the AWS Management Console homepage. The top navigation bar includes the AWS logo, a 'Services' dropdown, a 'Resource Groups' dropdown, user information (scotta @ 6115-2050-7156, Oregon), and a 'Support' link. A search bar at the top right contains the placeholder text 'Search @ NV...'. Below the navigation is a sidebar with links to 'History', 'Console Home', 'EC2', 'EFS', and 'Storage Gateway'. The main content area displays a grid of service icons and names. The 'EC2' service icon is highlighted with a red box. Other services listed include Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, X-Ray), Machine Learning (Amazon SageMaker, Amazon Comprehend, AWS DeepLens, Amazon Lex, Machine Learning, Amazon Polly, Rekognition, Amazon Transcribe, Amazon Translate), AR & VR (Amazon Sumerian), Application Integration (Step Functions, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF), Customer Engagement (Amazon Connect, Pinpoint, Simple Email Service), Business Productivity (Alexa for Business, Amazon Chime, WorkDocs, WorkMail), Desktop & App Streaming (WorkSpaces, AppStream 2.0), and Security, Identity & Compliance (IAM, Cognito, GuardDuty, Inspector, Amazon Macie, AWS Single Sign-On). A large search bar at the top center allows users to find specific services by name or feature.

AWS Instance Launching

The screenshot shows the AWS Management Console EC2 Dashboard. On the left, there's a navigation sidebar with various services like EC2 Dashboard, Instances, Images, Elastic Block Store, Network & Security, Load Balancing, and Auto Scaling. The main area displays EC2 resources in the US West (Oregon) region, including 2 Running Instances, 5 Elastic IPs, 18 Volumes, 50 Key Pairs, 3 Snapshots, 0 Dedicated Hosts, 0 Load Balancers, 0 Placement Groups, and 92 Security Groups. A prominent 'Create Instance' section features a large blue 'Launch Instance' button, which is highlighted with a red box. Below it, a note says 'To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.' To the right, sections for Account Attributes (Supported Platforms: VPC), Additional Information (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us), and AWS Marketplace (free software trial products) are visible.

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Home

EC2 Dashboard

Events
Tags
Reports
Limits

INSTANCES

Instances
Launch Templates
Spot Requests
Reserved Instances
Dedicated Hosts
Scheduled Instances

IMAGES

AMIs
Bundle Tasks

ELASTIC BLOCK STORE

Volumes
Snapshots

NETWORK & SECURITY

Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces

LOAD BALANCING

Load Balancers
Target Groups

AUTO SCALING

Feedback English (US)

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

2 Running Instances	5 Elastic IPs
0 Dedicated Hosts	3 Snapshots
18 Volumes	0 Load Balancers
50 Key Pairs	92 Security Groups
0 Placement Groups	

Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the [EC2 Videos](#).

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your instances will launch in the US West (Oregon) region

Service Health

Service Status:

- US West (Oregon): This service is operating normally

Availability Zone Status:

- us-west-2a: Availability zone is operating normally
- us-west-2b: Availability zone is operating normally
- us-west-2c: Availability zone is operating normally

Scheduled Events

US West (Oregon): No events

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Account Attributes

Supported Platforms: VPC

Default VPC: vpc-bffdd9d9

Resource ID length management

Additional Information

Getting Started Guide
Documentation
All EC2 Resources
Forums
Pricing
Contact Us

AWS Marketplace

Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular AMIs:

Barracuda CloudGen Firewall for AWS - PAYG
Provided by Barracuda Networks, Inc.
Rating ★★★★
Starting from \$0.60/hr or from \$4,599/yr (12% savings) for software + AWS usage fees
View all Infrastructure Software

Matillion ETL for Snowflake
Provided by Matillion
Rating ★★★★
Starting from \$1.37/hr or from \$9,950/yr (17% savings) for software + AWS usage fees

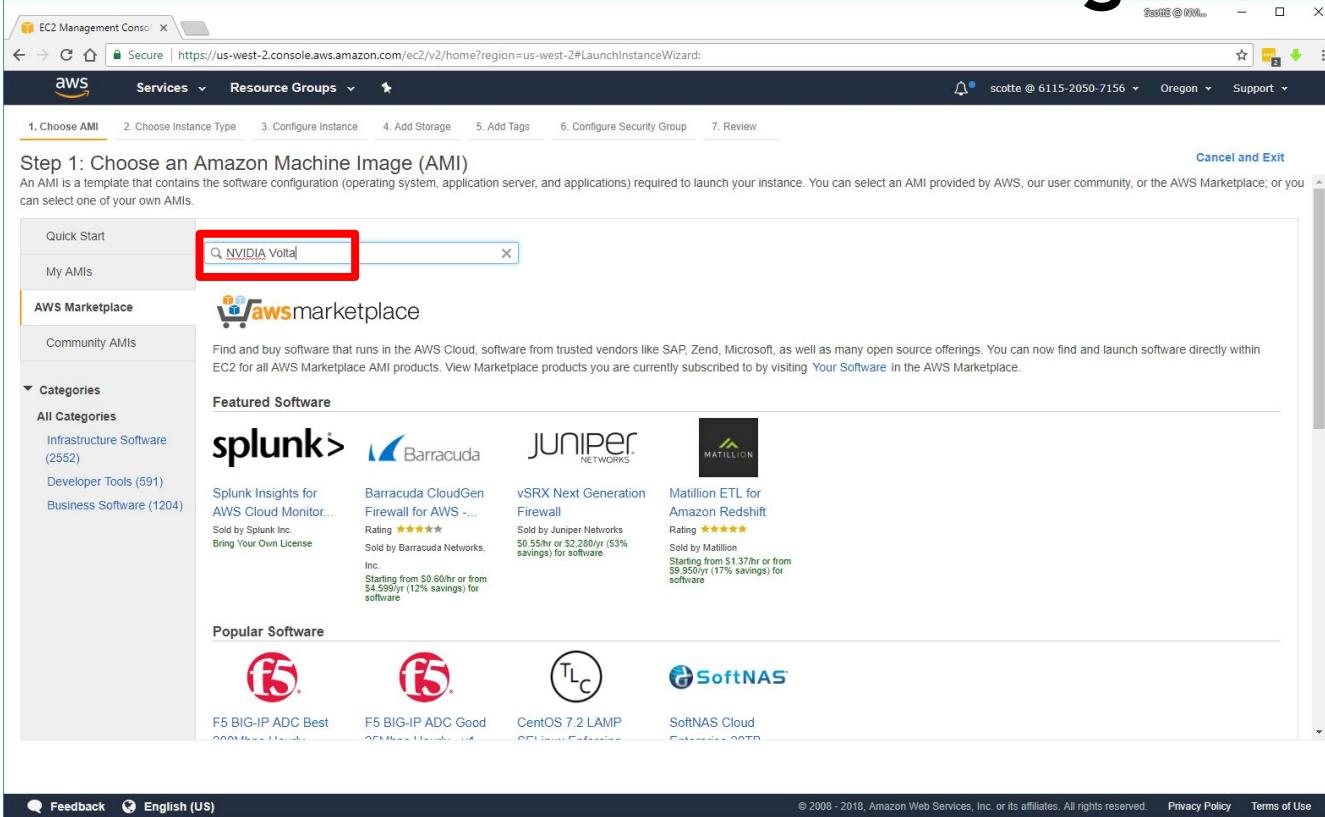
AWS Instance Launching

The screenshot shows the AWS Management Console EC2 Instance Launch Wizard. The current step is "Step 1: Choose an Amazon Machine Image (AMI)". The left sidebar shows navigation options: Quick Start, My AMIs, AWS Marketplace (which is highlighted with a red box), and Community AMIs. The right pane lists several AMI options:

- Amazon Linux AMI 2017.09.1 (HVM), SSD Volume Type - ami-d874e0a0**
Amazon Linux
Free tier eligible
The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select (button) 64-bit
- Amazon Linux 2 LTS Candidate AMI 2017.12.0 (HVM), SSD Volume Type - ami-7f43f307**
Amazon Linux
Free tier eligible
Amazon Linux 2 is the next generation of Amazon Linux. It includes the latest LTS kernel (4.9) tuned for enhanced performance on Amazon EC2, systemd support, newer versions of glibc, gcc and binutils, and an additional set of core packages for performance and security improvements.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select (button) 64-bit
- SUSE Linux Enterprise Server 12 SP3 (HVM), SSD Volume Type - ami-6bc56f13**
SUSE Linux
Free tier eligible
SUSE Linux Enterprise Server 12 Service Pack 3 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select (button) 64-bit
- Red Hat Enterprise Linux 7.4 (HVM), SSD Volume Type - ami-223f945a**
Red Hat
Free tier eligible
Red Hat Enterprise Linux version 7.4 (HVM), EBS General Purpose (SSD) Volume Type
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select (button) 64-bit
- Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-79873901**
Ubuntu
Free tier eligible
Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select (button) 64-bit

At the bottom of the page, there are links for Feedback, English (US), and a footer with copyright information: © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use.

AWS Instance Launching



The screenshot shows the AWS EC2 Management Console launching an instance. The user is on Step 1: Choose an Amazon Machine Image (AMI). A red box highlights the search bar where 'NVIDIA Volta' has been typed. The search results show various AMIs, including Splunk, Barracuda, Juniper Networks, and Matillion. Below the search results, there are sections for Featured Software and Popular Software, featuring F5 BIG-IP ADC, CentOS 7.2 LAMP, and SoftNAS Cloud.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

My AMIs

AWS Marketplace

Community AMIs

Categories

All Categories

- Infrastructure Software (2552)
- Developer Tools (591)
- Business Software (1204)

NVIDIA Volta

splunk Barracuda JUNIPER NETWORKS Matillion

Splunk Insights for AWS Cloud Monitor... Rating ★★★★☆ Sold by Splunk Inc. Bring Your Own License

Barracuda CloudGen Firewall for AWS ... Rating ★★★★☆ Sold by Barracuda Networks, Inc.

vSRX Next Generation Firewall Rating ★★★★☆ Sold by Juniper Networks

Matillion ETL for Amazon Redshift Rating ★★★★☆ Sold by Matillion Starting from \$1.37/hr or from \$9.95/yr (17% savings) for software

F5 BIG-IP ADC Best Rating ★★★★☆ Sold by F5 Networks

F5 BIG-IP ADC Good Rating ★★★★☆ Sold by F5 Networks

CentOS 7.2 LAMP Rating ★★★★☆ Sold by CentOS

SoftNAS Cloud Rating ★★★★☆ Sold by SoftNAS

Feedback English (US)

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard:

Cancel and Exit

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

My AMIs

AWS Marketplace

NVIDIA Volta

NVIDIA Volta Deep Learning AMI
★★★★★ (1) | 18.03.0 | Sold by NVIDIA
\$0.00/hr for software + AWS usage fees
Linux/Ubuntu, Ubuntu 16.04 | 64-bit Amazon Machine Image (AMI) | Updated: 3/7/18
The NVIDIA Volta Deep Learning AMI is an optimized environment for running the deep learning frameworks available from the NVIDIA GPU Cloud (NGC) container registry. The Docker ...
More info

Select

Deep Learning AMI with Source Code (CUDA 9, Amazon Linux)
★★★★★ (3) | 5.0 | Sold by Amazon Web Services
\$0.023 to \$41.944/hr incl EC2 charges + other AWS usage fees
Linux/Ubuntu, Amazon Linux 2017.09 | 64-bit Amazon Machine Image (AMI) | Updated: 2/27/18
Comes with deep learning frameworks custom built from source to enable advanced optimizations. Apache MXNet, TensorFlow, PyTorch, Keras 2.0 and Caffe2 configured with CUDA 9, cuDNN ...
More info

Select

Deep Learning AMI with Source Code (CUDA 9, Ubuntu)
★★★★★ (4) | 5.0 | Sold by Amazon Web Services
\$0.023 to \$41.944/hr incl EC2 charges + other AWS usage fees
Linux/Ubuntu, Ubuntu 16.04 | 64-bit Amazon Machine Image (AMI) | Updated: 2/27/18
Comes with deep learning frameworks custom built from source to enable advanced optimizations. Apache MXNet, TensorFlow, PyTorch, Keras 2.0 and Caffe2 configured with

Select

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

EC2 Management Console x

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard:

Services Resource Groups

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

NVIDIA Volta Deep Learning AMI

 NVIDIA Volta Deep Learning AMI

The NVIDIA Volta Deep Learning AMI is an optimized environment for running the deep learning frameworks available from the NVIDIA GPU Cloud (NGC) container registry. The Docker containers available on the NGC container registry are tuned, tested, and certified by NVIDIA to take full advantage of NVIDIA Volta Tensor Cores, the new driving force ...

[More info](#)

View Additional Details in AWS Marketplace

Pricing Details

Hourly Fees

Instance Type	Software	EC2	Total
GPU Compute 2 Extra Large	\$0.00	\$3.06	\$3.06/hr
GPU Compute 16 Extra Large	\$0.00	\$24.48	\$24.48/hr
GPU Compute 8 Extra Large	\$0.00	\$12.24	\$12.24/hr

EBS General Purpose (SSD) volumes
\$0.10 per GB-month of provisioned storage

You will not be charged until you launch this instance.

Sold by NVIDIA

Customer Rating ★★★★☆ (1)
Latest Version 18.03.0

Base Operating System Linux/Unix, Ubuntu 16.0.4
Delivery Method 64-bit Amazon Machine Image (AMI)
License Agreement End User License Agreement
On Marketplace Since 10/24/17
AWS Services Required EC2, EBS

Highlights

- Provides AI researchers with fast and easy access to NVIDIA Volta GPUs in the

Cancel Continue

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console launching an instance. The process is at Step 2: Choose an Instance Type. A tooltip for the GPU compute instance type is displayed, providing a detailed description of its capabilities. The 'Review and Launch' button is highlighted with a red box.

Step 2: Choose an Instance Type

GPU instances

Instance Type	Memory (GiB)	Cores	Storage (GB)	Local NVMe (GB)	Network Interface	Support	
g2.8xlarge	32	60	2 x 120 (SSD)	-	10 Gigabit	-	
GP	GPU			EBS only	Yes	High	Yes
GP	GPU			EBS only	Yes	10 Gigabit	Yes
GP	GPU			EBS only	Yes	25 Gigabit	Yes
GPU compute	p3.2xlarge	8	61	EBS only	Yes	Up to 10 Gigabit	Yes
GPU compute	p3.6xlarge	32	244	EBS only	Yes	10 Gigabit	Yes
GPU compute	p3.16xlarge	64	488	EBS only	Yes	25 Gigabit	Yes
Memory optimized	r4.large	2	15.25	EBS only	Yes	Up to 10 Gigabit	Yes
Memory optimized	r4.xlarge	4	30.5	EBS only	Yes	Up to 10 Gigabit	Yes
Memory optimized	r4.2xlarge	8	61	EBS only	Yes	Up to 10 Gigabit	Yes
Memory optimized	r4.4xlarge	16	122	EBS only	Yes	Up to 10 Gigabit	Yes
Memory optimized	r4.8xlarge	32	244	EBS only	Yes	10 Gigabit	Yes
Memory optimized	r4.16xlarge	64	488	EBS only	Yes	25 Gigabit	Yes
Memory optimized	x1.16xlarge	64	976	1 x 1920 (SSD)	Yes	10 Gigabit	Yes
Memory optimized	x1e.xlarge	4	122	1 x 120 (SSD)	Yes	Up to 10 Gigabit	Yes
Memory optimized	x1e.2xlarge	8	244	1 x 240 (SSD)	Yes	Up to 10 Gigabit	Yes

Buttons: Cancel, Previous, **Review and Launch**, Next: Configure Instance Details

Page Footer: Feedback, English (US), © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy, Terms of Use

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console interface for launching an instance. The title bar reads "EC2 Management Console" and the URL is "https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard". The top navigation bar includes "AWS Services" and "Resource Groups". The main content area is titled "Step 3: Configure Instance Details" with the sub-instruction "Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more." Below this, there are several configuration sections:

- Number of Instances:** Set to 1, with an option to "Launch into Auto Scaling Group".
- Purchasing option:** An unchecked checkbox for "Request Spot instances".
- Network:** Set to "vpc-bffdd9d9 (default)". Includes options to "Create new VPC" or "Create new subnet".
- Subnet:** Set to "No preference (default subnet in any Availability Zone)". Includes an option to "Create new subnet".
- Auto-assign Public IP:** Set to "Use subnet setting (Enable)".
- Placement group:** An unchecked checkbox for "Add instance to placement group".
- IAM role:** Set to "None". Includes an option to "Create new IAM role".
- Shutdown behavior:** Set to "Stop".
- Enable termination protection:** An unchecked checkbox for "Protect against accidental termination".
- Monitoring:** An unchecked checkbox for "Enable CloudWatch detailed monitoring". A note states "Additional charges apply".
- EBS-optimized instance:** A checked checkbox with the label "Launch as EBS-optimized instance".
- Tenancy:** Set to "Shared - Run a shared hardware instance". A note states "Additional charges will apply for dedicated tenancy".

At the bottom, there is a "Advanced Details" section with a disclosure arrow. Below that, a navigation bar includes "Cancel", "Previous", "Review and Launch" (which is highlighted with a red box), and "Next: Add Storage". The footer contains links for "Feedback", "English (US)", "Privacy Policy", and "Terms of Use", along with copyright information: "© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved."

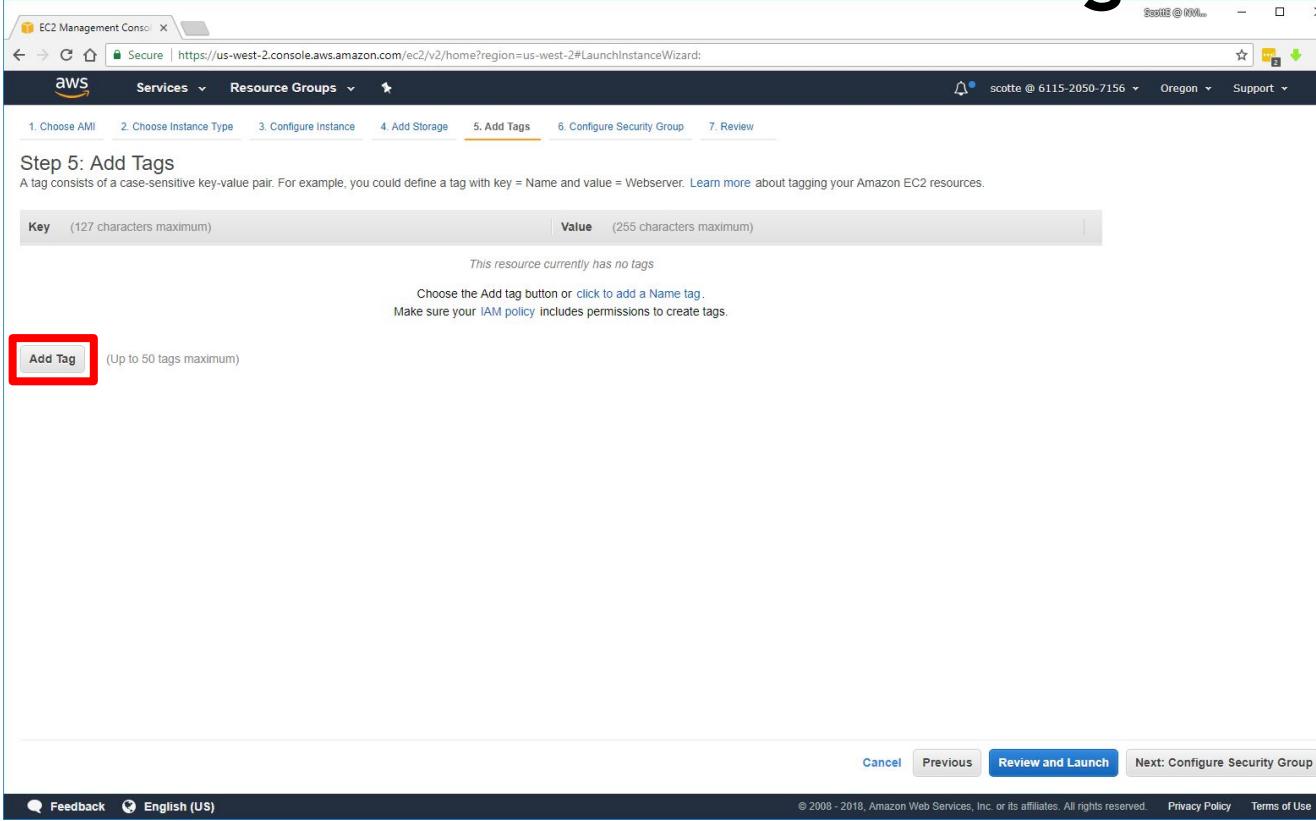
AWS Instance Launching

The screenshot shows the AWS EC2 Management Console interface for launching an instance. The user is currently on Step 4: Add Storage. The page title is "Step 4: Add Storage". Below the title, there is a note: "Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2." A table displays the current storage configuration:

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-019818e0fc2b43d25	32	General Purpose SSD (GP2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Below the table, there is a button labeled "Add New Volume". A callout box contains the text: "Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions." At the bottom of the page, there are navigation buttons: "Cancel", "Previous", "Review and Launch" (which is highlighted with a red box), and "Next: Add Tags".

AWS Instance Launching



The screenshot shows the AWS EC2 Management Console interface for launching a new instance. The user is currently on Step 5: Add Tags. The navigation bar at the top includes links for Choose AMI, Choose Instance Type, Configure Instance, Add Storage, Add Tags (which is the active step), Configure Security Group, and Review.

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tagging your Amazon EC2 resources.

This resource currently has no tags.

Choose the Add tag button or [click](#) to add a Name tag.
Make sure your [IAM policy](#) includes permissions to create tags.

Add Tag (Up to 50 tags maximum)

At the bottom of the screen, there are buttons for Cancel, Previous, Review and Launch (which is highlighted in blue), and Next: Configure Security Group.

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console interface for launching a new instance. The user is currently on Step 5: Add Tags. A single tag is being added, consisting of a key "Name" and a value "GTCInstance". The "Name" field is highlighted with a red box. At the bottom right, the "Next: Configure Security Group" button is also highlighted with a red box.

EC2 Management Console | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard:

AWS Services Resource Groups

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)
Name	GTCInstance

Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console interface for launching an instance. The user is currently on Step 6: Configure Security Group. A red box highlights the radio button for selecting an existing security group. Another red box highlights the security group 'sg-02eb047c GTCSecurityGroup' in the list. A third red box highlights the 'Review and Launch' button at the bottom right.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

Assign a security group:

- Create a new security group
- Select an **existing** security group

Security	Name	Description
	sg-02eb047c GTCSecurityGroup	GTCSecurityGroup

Inbound rules for sg-02eb047c (Selected security groups: sg-02eb047c)

Type	Protocol	Port Range	Source	Description
Custom TCP Rule	TCP	8888	0.0.0.0/0	Jupyter Notebook
Custom TCP Rule	TCP	8888	::/0	Jupyter Notebook
Custom TCP Rule	TCP	6006	0.0.0.0/0	TensorBoard

Cancel Previous **Review and Launch**

AWS Instance Launching

Danger Danger!
Use your local subnet!

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and continue to the launch process.

AMI Details

NVIDIA Volta Deep Learning AMI
Root Device Type: ebs Virtualization type: hvm

Hourly Software Fees: \$0.00 per hour on p3.2xlarge instance (Additional taxes may apply)
Software charges will begin once you launch this AMI and continue until you terminate the instance.

By launching this product, you will be subscribed to this software and agree that your use of this software is subject to the pricing terms and the seller's End User License Agreement.

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
p3.2xlarge	23.5	8	61	EBS only	Yes	Up to 10 Gigabit

Security Groups

Security Group ID	Name	Description
sa-02eb047c	GTCSecurityGroup	GTCSecurityGroup

Launch

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console Step 7: Review Instance Launch wizard. The main page displays instance details, including the AMI (NVIDIA Volta Deep Learning AMI), instance type (p3.2xlarge), and security group (GTCSecurityGroup). A modal window titled "Select an existing key pair or create a new key pair" is open, showing a dropdown menu with "GTCKey" selected. A red box highlights the "GTCKey" entry in the dropdown and the acknowledgment checkbox below it. The "Launch Instances" button at the bottom right of the modal is also highlighted with a red box.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

NVIDIA Volta Deep Learning AMI
Hourly Software Fees: \$0.00 per hour on p3.2xlarge

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)
p3.2xlarge	23.5	8	61

Security Groups

Security Group ID	Name	Description
sg-02eb047c	GTCSecurityGroup	GTCSecurityGroup

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Choose an existing key pair
Select a key pair
GTCKey
 I acknowledge that I have access to the selected private key file (GTCKey.pem), and that without this file, I won't be able to log into my instance.

Cancel **Launch Instances**

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console with the URL <https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard>. The page displays the 'Launch Status' section, which includes a green box indicating 'Your instances are now launching' with a link to 'View launch log'. A red box highlights the instance ID 'i-0900cf09baa76f6c5'. Below this, there's a section for 'Get notified of estimated charges' and instructions on how to connect to instances. A dropdown menu 'Getting started with your software' is open, showing links to 'View Usage Instructions' and 'Open Your Software on AWS Marketplace'. Further down, there's a list of helpful resources and links for managing security groups and EBS volumes.

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard:

AWS Services Resource Groups

scotte @ 6115-2050-7156 Oregon Support

Launch Status

Your instances are now launching

The following instance launches have been initiated: [i-0900cf09baa76f6c5](#) View launch log

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the Instances screen. [Find out](#) how to connect to your instances.

Getting started with your software

To get started with NVIDIA Volta Deep Learning AMI To manage your software subscription

[View Usage Instructions](#) [Open Your Software on AWS Marketplace](#)

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your instances are launching you can also

Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)

Create and attach additional EBS volumes (Additional charges may apply)

Manage security groups

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console interface. A red box highlights the 'Connect' button in the top navigation bar. The main pane displays a table of instances, with one row selected for 'GTCInstance'. The detailed view on the right shows the instance's configuration, including its name, instance ID, type, state, and network details.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs
GTCInstance	i-0900cf09baa76f6c5	p3.2xlarge	us-west-2c	running	Initializing	None	ec2-54-213-139-71.us-west-2.compute.amazonaws.com	54.213.139.71	-

Instance: i-0900cf09baa76f6c5 (GTCInstance) Public DNS: ec2-54-213-139-71.us-west-2.compute.amazonaws.com

Description	Status Checks	Monitoring	Tags	Usage Instructions
Instance ID: i-0900cf09baa76f6c5	Public DNS (IPv4): ec2-54-213-139-71.us-west-2.compute.amazonaws.com	IPv4 Public IP: 54.213.139.71		
Instance state: running		IPv6 IPs: -		
Instance type: p3.2xlarge		Private DNS: ip-172-31-7-211.us-west-2.compute.internal		
Elastic IPs:		Private IPs: 172.31.7.211		
Availability zone: us-west-2c		Secondary private IPs:		
Security groups: GTCSecurityGroup, view inbound rules		VPC ID: vpc-bffdd9d9		
Scheduled events: No scheduled events		Subnet ID: subnet-773e0b2c		
AMI ID: NVIDIA Volta Deep Learning AMI-46a68101-e56b-41cd-8e32-631ac6e5d02b-ami-e59f6698.4 (ami-2a970052)		Network interfaces: eth0		
Platform: -		Source/dest. check: True		
IAM role: -		T2 Unlimited: -		
Key pair name: GTCKey		Owner: 611520507156		

AWS Instance Launching

The screenshot shows the AWS EC2 Management Console with the 'Instances' section selected. A modal window titled 'Connect To Your Instance' is open, providing instructions for connecting to the instance. The 'Description' tab is selected in the modal's left sidebar. A red box highlights the command: `ssh -i "GTCKey.pem" root@ec2-54-213-139-71.us-west-2.compute.amazonaws.com`. A green callout box points from this command to the text: 'For NVIDIA AMI use "ubuntu" instead of "root"'.

For NVIDIA AMI use “ubuntu” instead of “root”

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Instances:search=i-0900cf09baa76f6c5;sort=instanceId

EC2 Dashboard Services Resource Groups

Launch Instance Connect Actions

Instances

GTCInstance i-0900cf09baa76f6c5

Connect To Your Instance

I would like to connect with:

- A standalone SSH client
- A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to connect using PuTTY)
2. Locate your private key file (GTCKey.pem). The wizard automatically detects the key file you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if you need to change it:
`chmod 400 GTCKey.pem`
4. Connect to your instance using its Public DNS:
`ec2-54-213-139-71.us-west-2.compute.amazonaws.com`

Example:

`ssh -i "GTCKey.pem" root@ec2-54-213-139-71.us-west-2.compute.amazonaws.com`

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Instance Launching

The screenshot shows a terminal window with the following text:

```
scotte@DC1HVHQW10-101:~$ ssh -i .ssh/GTCKey.pem ubuntu@ec2-54-213-139-71.us-west-2.compute.amazonaws.com
The authenticity of host 'ec2-54-213-139-71.us-west-2.compute.amazonaws.com (54.213.139.71)' can't be established.
ECDSA key fingerprint is 22:58:ce:9e:89:d6:93:a3:96:c:b7:4f:4b:4a:ec:6c
Are you sure you want to continue connecting (yes/no)? yes
```

Annotations with green boxes and arrows point to specific parts of the terminal output:

- An arrow points from the text "SSH Key saved earlier" to the command `ssh -i .ssh/GTCKey.pem`.
- An arrow points from the text "FQDN of our new instance" to the IP address and region in the command `ubuntu@ec2-54-213-139-71.us-west-2.compute.amazonaws.com`.
- An arrow points from the text "'ubuntu' is the username" to the word `ubuntu` in the command.

AWS Instance Launching

```
scotte@DC1HVHQW10-101: ~
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1052-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

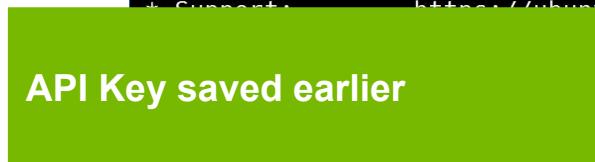
 Get cloud support with Ubuntu Advantage Cloud Guest:
   http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

Welcome to the NVIDIA Volta Deep Learning AMI. This environment is provided to
enable you to easily run the Deep Learning containers from the NGC Registry.
All of the documentation for how to use NGC and this AMI are found at
  http://docs.nvidia.com/deeplearning/ngc

Please enter your NGC APIkey to login to the NGC Registry:
-
```

AWS Instance Launching



```
ubuntu@ip-172-31-7-211:~$ Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1052-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Advantage Cloud Guest:
https://ss/services/cloud

0 updates are secu...ates.

Welcome to the NVIDIA Volta Deep Learning AMI. This environment is provided to
enable you to easily run the Deep Learning containers from the NGC Registry.
All of the documentation for how to use NGC and this AMI are found at
http://docs.nvidia.com/deeplearning/ngc

Please enter your NGC APIkey to login to the NGC Registry.
a2VmWkxZG1iNHE2amMyNjlzZGY2MXUw0WM6Zjg4ZGQxMGMtNGViYy00NTAyLThhY2EtMmQ30DljMWFhMmQ4
Logging into the NGC Registry at nvcr.io.....Login Succeeded
ubuntu@ip-172-31-7-211:~$
```

Run a Container

NGC Container Execution

Quick TensorFlow Run

Our challenge:

- Download a TensorFlow container from NGC
- Run the container in our P3 instance

NGC Container Execution

The screenshot shows the NVIDIA GPU CLOUD Registry interface. A red box highlights the 'Registry' tab in the left sidebar. Another red box highlights the 'tensorflow' entry under the 'nvidia' repository. A third red box highlights the lock icon in the Docker pull command bar.

Registry

Repositories

- nvidia ▾
 - caffe
 - caffe2
 - cntk
 - cuda
 - digits
 - mxnet
- tensorflow
- theano
- torch
- hpc ^
- nvidia-hpcvis ^
- partners ^

User Forum ↗
System Status ↗

nvidia/tensorflow

```
docker pull nvcr.io/nvidia/tensorflow:18.02-py3
```

What is TensorFlow?

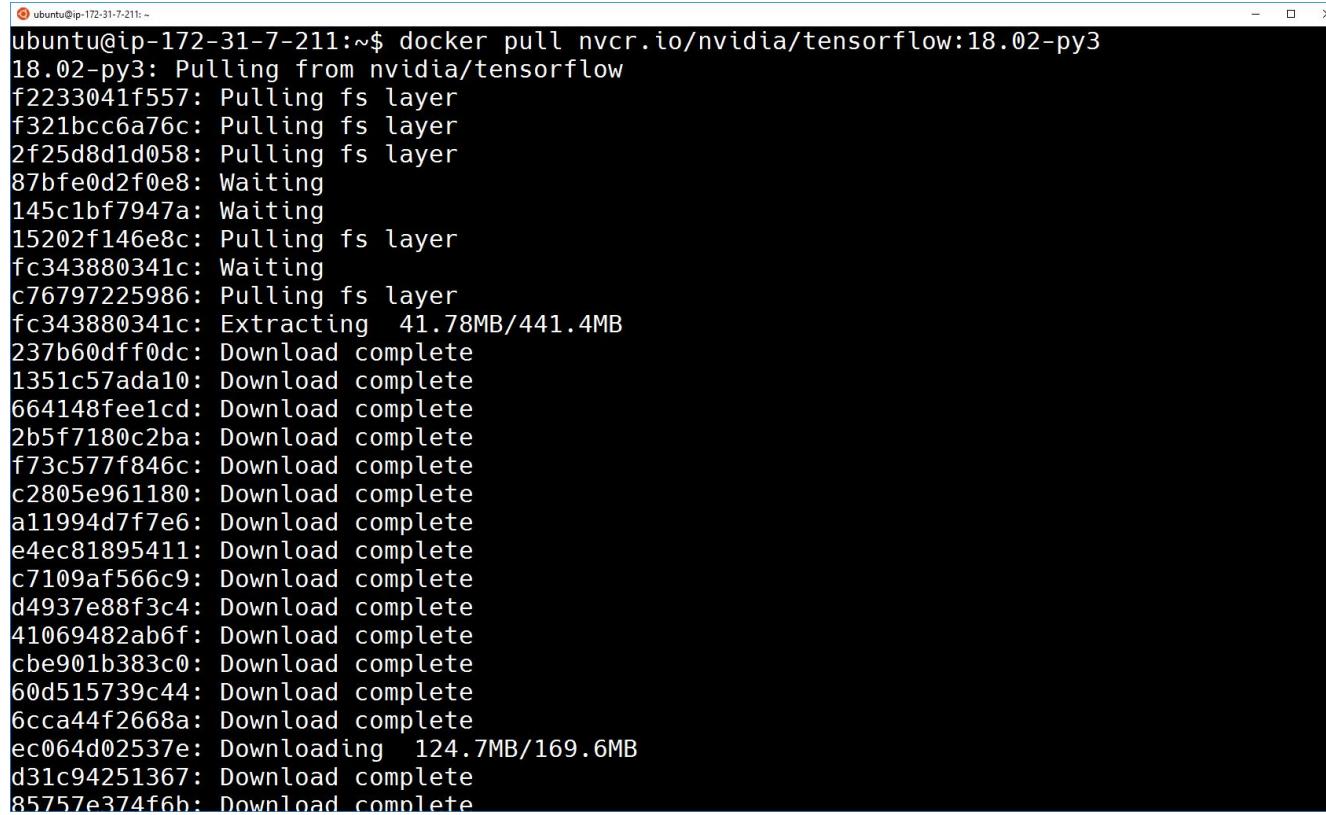
TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture lets you deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device without rewriting code.

TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well.

Running TensorFlow

TAG	SIZE	USER	LAST MODIFIED	PULL
18.02-py3	1.28 GB		March 2, 2018	↓
18.02-py2	1.28 GB		March 2, 2018	↓
18.01-py2	1.25 GB		January 23, 2018	↓

NGC Container Execution



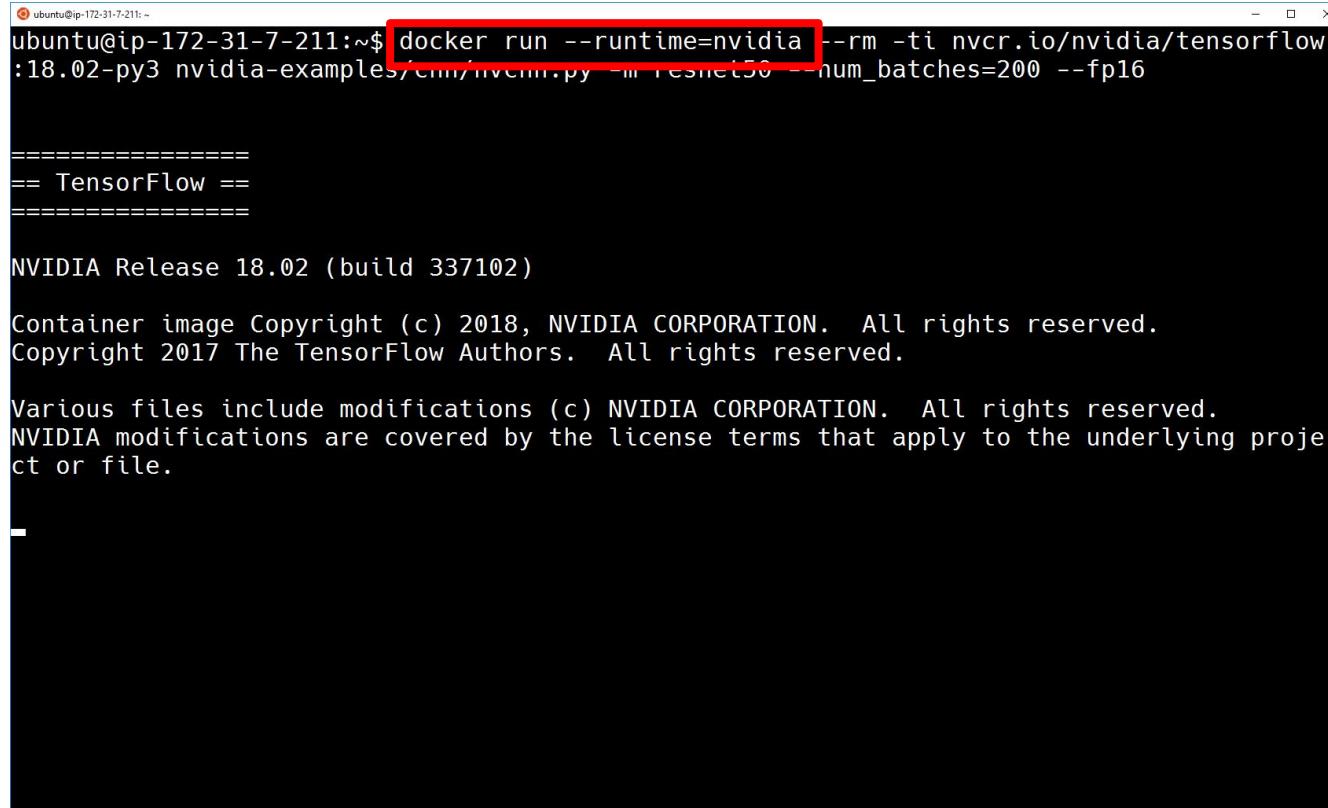
A terminal window titled "ubuntu@ip-172-31-7-211: ~" showing the output of a Docker pull command. The command is "docker pull nvcr.io/nvidia/tensorflow:18.02-py3". The output lists the layers being pulled, their status, and their sizes. Most layers are 441.4MB, while one layer is 169.6MB.

```
ubuntu@ip-172-31-7-211:~$ docker pull nvcr.io/nvidia/tensorflow:18.02-py3
18.02-py3: Pulling from nvidia/tensorflow
f2233041f557: Pulling fs layer
f321bcc6a76c: Pulling fs layer
2f25d8d1d058: Pulling fs layer
87bfe0d2f0e8: Waiting
145c1bf7947a: Waiting
15202f146e8c: Pulling fs layer
fc343880341c: Waiting
c76797225986: Pulling fs layer
fc343880341c: Extracting 41.78MB/441.4MB
237b60dff0dc: Download complete
1351c57ada10: Download complete
664148fee1cd: Download complete
2b5f7180c2ba: Download complete
f73c577f846c: Download complete
c2805e961180: Download complete
a11994d7f7e6: Download complete
e4ec81895411: Download complete
c7109af566c9: Download complete
d4937e88f3c4: Download complete
41069482ab6f: Download complete
cbe901b383c0: Download complete
60d515739c44: Download complete
6cca44f2668a: Download complete
ec064d02537e: Downloading 124.7MB/169.6MB
d31c94251367: Download complete
85757e374f6b: Download complete
```

NGC Container Execution

```
ubuntu@ip-172-31-7-211: ~
2b5f7180c2ba: Pull complete
664148fee1cd: Extracting    472B/472B
664148fee1cd: Download complete
2b5f7180c2ba: Download complete
f73c577f846c: Pull complete
c2805e961180: Pull complete
a11994d7f7e6: Pull complete
e4ec81895411: Pull complete
c7109af566c9: Pull complete
d4937e88f3c4: Pull complete
41069482ab6f: Pull complete
cbe901b383c0: Pull complete
60d515739c44: Pull complete
6cca44f2668a: Pull complete
ec064d02537e: Pull complete
d31c94251367: Pull complete
85757e374f6b: Pull complete
a685c53320ed: Pull complete
f7e832cb61d2: Pull complete
f743b7cb9be2: Pull complete
0c395732af81: Pull complete
7ee97eeb04b4: Pull complete
e8c1d8550a0d: Pull complete
65154325fd45: Pull complete
fb91e851e672: Pull complete
Digest: sha256:899f5407ac404eb94c8277d8ff845e2946e1e5e24639aa3b6e75f15de12a7120
Status: Downloaded newer image for nvcr.io/nvidia/tensorflow:18.02-py3
ubuntu@ip-172-31-7-211:~$
```

NGC Container Execution



A terminal window titled "ubuntu@ip-172-31-7-211: ~" displaying the command and its output. The command is highlighted with a red box.

```
ubuntu@ip-172-31-7-211:~$ docker run --runtime=nvidia --rm -ti nvcr.io/nvidia/tensorflow:18.02-py3 nvidia-examples/cml/nvcnn.py -m ResNet50 --num_batches=200 --fp16
```

=====

== TensorFlow ==

=====

NVIDIA Release 18.02 (build 337102)

Container image Copyright (c) 2018, NVIDIA CORPORATION. All rights reserved.
Copyright 2017 The TensorFlow Authors. All rights reserved.

Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.
NVIDIA modifications are covered by the license terms that apply to the underlying project or file.

-

NGC Container Execution

```
ubuntu@ip-172-31-7-211: ~
178     1  641.4  11.027 0.10000
179     1  640.2  11.050 0.10000
180     1  639.4  11.035 0.10000
181     1  639.4  11.027 0.10000
182     1  639.8  11.014 0.10000
183     1  637.2  11.001 0.10000
184     1  640.5  10.994 0.10000
185     1  641.1  10.986 0.10000
186     1  640.4  10.979 0.10000
187     1  641.6  10.972 0.10000
188     1  642.4  10.965 0.10000
189     1  642.7  10.961 0.10000
190     1  642.5  10.980 0.10000
191     1  640.0  10.992 0.10000
192     1  642.3  10.986 0.10000
193     1  641.6  10.977 0.10000
194     1  641.3  10.976 0.10000
195     1  641.5  10.990 0.10000
196     1  641.7  10.981 0.10000
197     1  642.4  10.975 0.10000
198     1  640.9  10.967 0.10000
199     1  641.0  10.962 0.10000
200     1  641.6  10.957 0.10000
-----
Images/sec: 640.9 +/- 0.1 (jitter = 1.2)
-----
ubuntu@ip-172-31-7-211:~$
```

Wait. What Just Happened?

That was too fast.

1. Logged into NGC and created an API key
2. Logged into AWS and setup our account
 - a. Created SSH Key Pair
 - b. Created Security Group
3. Launched AWS P3 instance with “NVIDIA Volta Deep Learning” AMI
 - a. Logged into the P3 instance with SSH
 - b. Logged into NGC with the API key
4. Downloaded the TensorFlow container from NGC
5. Ran nvcnn.py sample with Resnet-50 with synthetic ImageNet data for 200 epochs
6. Profit!

Exposing Ports

Accessing Container Services

What about things in the container?

Applications in a container are on their own network ('docker0' bridge)

```
ubuntu@ip-172-31-7-161: ~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 0a:8f:c6:32:37:24 brd ff:ff:ff:ff:ff:ff
        inet 172.31.7.161/20 brd 172.31.15.255 scope global ens3
            valid_lft forever preferred_lft forever
        inet6 fe80::88f:c6ff:fe32:3724/64 scope link
            valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:75:48:55:43 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever
ubuntu@ip-172-31-7-161: ~
```

Tell Docker you want to use them at runtime (remember -p ?)

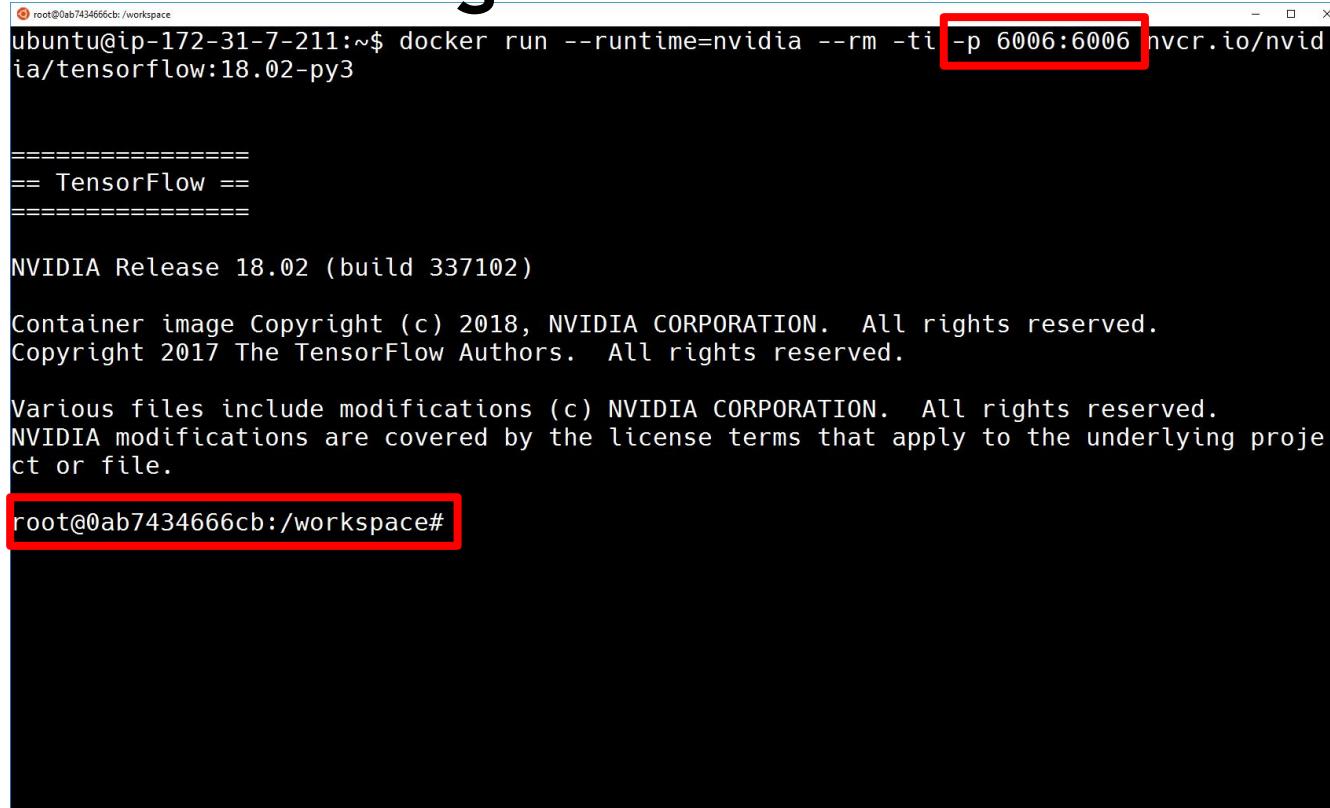
Accessing Container Services

Passing through a TCP port

Our challenge:

- Launch the TensorFlow container
 - Run interactively (`-ti`)
 - Expose Tensorboard port 6006 (`-p`)
- Repeat our prior training run
 - Save log data to `/tmp`
- Run Tensorboard
- Visualize our training run

Accessing Container Services



The screenshot shows a terminal window with a black background and white text. At the top, it displays the command: `ubuntu@ip-172-31-7-211:~$ docker run --runtime=nvidia --rm -ti -p 6006:6006 nvcr.io/nvidia/tensorflow:18.02-py3`. A red box highlights the port mapping part of the command: `-p 6006:6006`. Below the command, the TensorFlow logo and version information are displayed: `=====`, `== TensorFlow ==`, `=====`, `NVIDIA Release 18.02 (build 337102)`, `Container image Copyright (c) 2018, NVIDIA CORPORATION. All rights reserved.`, `Copyright 2017 The TensorFlow Authors. All rights reserved.`, `Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.`, `NVIDIA modifications are covered by the license terms that apply to the underlying project or file.`. At the bottom, the prompt `root@0ab7434666cb:/workspace#` is shown, also highlighted with a red box.

Accessing Container Services

```
root@29eddc293bf8:/workspace# ./nvidia-examples/cnn/nvcnn.py -m resnet50 --num_batches=200 --log_dir=/tmp --fp16
TensorFlow: 1.4.0
This script: ./nvidia-examples/cnn/nvcnn.py v1.4
Cmd line args:
-m
resnet50
--num_batches=200
--log_dir=/tmp
--fp16
Num images: Synthetic
Model: resnet50
Batch size: 64 global
          64.0 per device
Devices: ['/gpu:0']
Data format: NCHW
Data type: fp16
Have NCCL: True
Using NCCL: True
Using XLA: False
Building training graph
```

Accessing Container Services

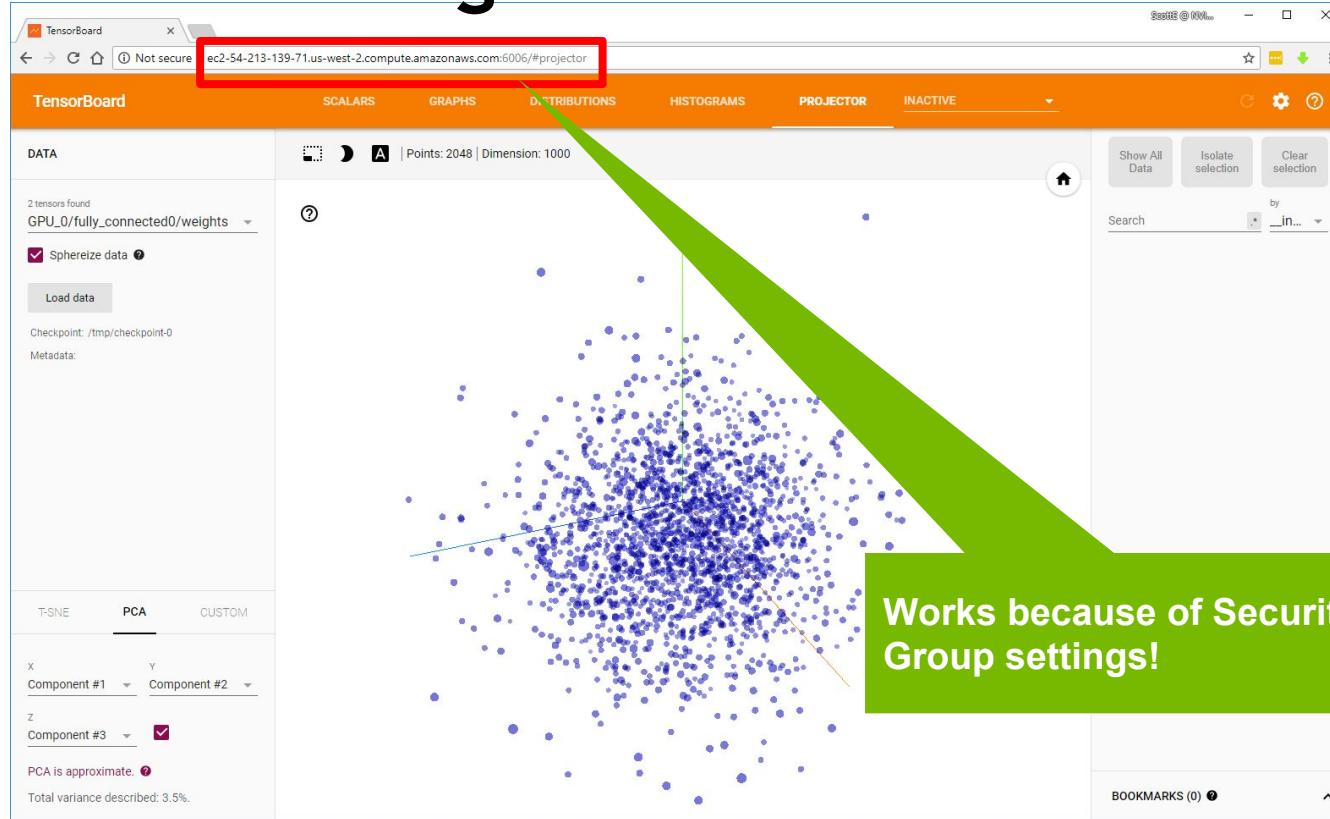
```
root@29eddc293bf8:/workspace
178     1  641.8  10.589 0.10000
179     1  640.5  10.608 0.10000
180     1  641.3  10.598 0.10000
181     1  638.4  10.591 0.10000
182     1  640.4  10.580 0.10000
183     1  638.4  10.569 0.10000
184     1  638.5  10.563 0.10000
185     1  638.1  10.556 0.10000
186     1  642.2  10.552 0.10000
187     1  641.5  10.546 0.10000
188     1  641.1  10.539 0.10000
189     1  640.7  10.535 0.10000
190     1  640.9  10.554 0.10000
191     1  640.9  10.561 0.10000
192     1  642.4  10.556 0.10000
193     1  641.0  10.548 0.10000
194     1  640.9  10.548 0.10000
195     1  640.3  10.559 0.10000
196     1  639.5  10.552 0.10000
197     1  642.4  10.546 0.10000
198     1  641.1  10.539 0.10000
199     1  640.4  10.535 0.10000
200     1  642.1  10.531 0.10000
-----
Images/sec: 640.6 +/- 0.1 (jitter = 1.1)
-----
root@29eddc293bf8:/workspace#
```

Accessing Container Services

```
root@29eddc293bf8:/workspace# tensorboard --logdir=/tmp  
TensorBoard 0.4.0 at http://29eddc293bf8:6006 (Press CTRL+C to quit)  
-  
180 1 641.3 10.598 0.10000  
181 1 638.4 10.591 0.10000  
182 1 640.4 10.580 0.10000  
183 1 638.4 10.569 0.10000  
184 1 638.5 10.563 0.10000  
185 1 638.1 10.556 0.10000  
186 1 642.2 10.552 0.10000  
187 1 641.5 10.546 0.10000  
188 1 641.1 10.539 0.10000  
189 1 640.7 10.535 0.10000  
190 1 640.9 10.554 0.10000  
191 1 640.9 10.561 0.10000  
192 1 642.4 10.556 0.10000  
193 1 641.0 10.548 0.10000  
194 1 640.9 10.548 0.10000  
195 1 640.3 10.559 0.10000  
196 1 639.5 10.552 0.10000  
197 1 642.4 10.546 0.10000  
198 1 641.1 10.539 0.10000  
199 1 640.4 10.535 0.10000  
200 1 642.1 10.531 0.10000  
-  
Images/sec: 640.6 +/- 0.1 (jitter = 1.1)
```

Use the P3 instance FQDN instead

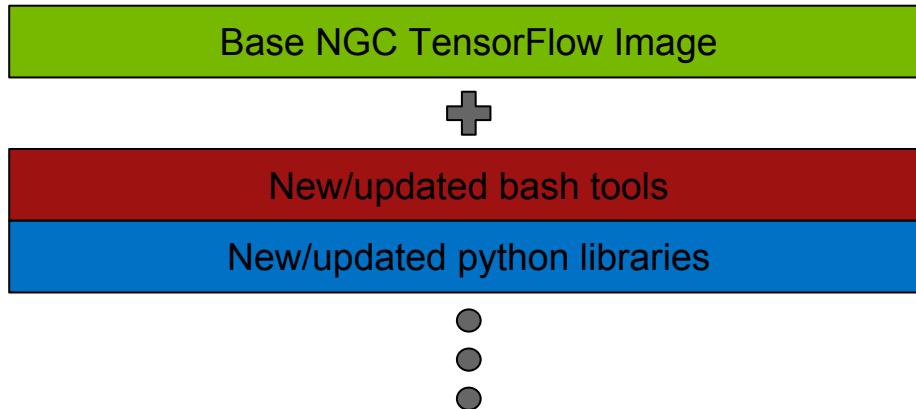
Accessing Container Services



Extending NGC Images

Extending NGC Images

Layers layers layers



- Often the out-of-the-box image is not enough
- Need extra tools/applications
 - Additional layers on top of base image

- Dockerfile allows for building custom images
 - `docker build` command creates new image from set of instructions

Extending NGC Images

A Dockerfile is a script that contains instructions to custom configure a container from a base image

Here are some common commands:

- **FROM** is Mandatory as the first instruction. It denotes the base image to be built from. Use a tag to specify the image.
- **RUN** = Creates a new layer with the output of the specified commands.
- **WORKDIR** = Directory the command will start it
- **CMD** = Default command executed when Docker container is started. Use only one CMD instruction in a Dockerfile.

```
1 FROM nvcr.io/nvidia/tensorflow:18.02-py3
2
3 RUN pip install jupyter
4
5 WORKDIR /notebooks
6
7 CMD jupyter notebook --allow-root --ip=0.0.0.0
```

Best practices for writing Dockerfiles

https://docs.docker.com/engine/userguide/eng-image/dockerfile_best-practices/

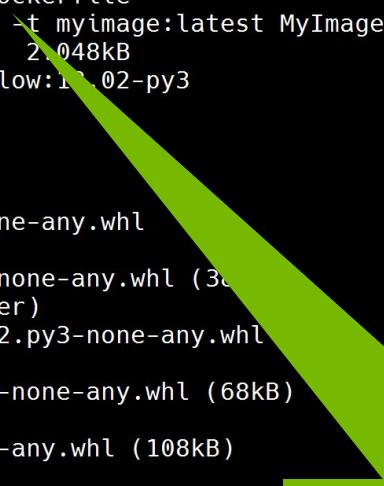
Extending NGC Images

Add Jupyter

Our challenge:

- Add Jupyter to the NVIDIA TensorFlow Image
- Launch jupyter notebook automatically when the container starts
 - By default jupyter listens on port 8888
- Verify it worked!

Extending NGC Images



```
ubuntu@ip-172-31-7-211:~$ mkdir MyImage
ubuntu@ip-172-31-7-211:~$ vi MyImage/Dockerfile
ubuntu@ip-172-31-7-211:~$ docker build -t myimage:latest MyImage
Sending build context to Docker daemon 2.048kB
Step 1/4 : FROM nvcr.io/nvidia/tensorflow:18.02-py3
--> 57ae51ee8b74
Step 2/4 : RUN pip install jupyter
--> Running in b9c14a05670f
Collecting jupyter
  Downloading jupyter-1.0.0-py2.py3-none-any.whl
Collecting nbconvert (from jupyter)
  Downloading nbconvert-5.3.1-py2.py3-none-any.whl (36
Collecting jupyter-console (from jupyter)
  Downloading jupyter_console-5.2.0-py2.py3-none-any.whl
Collecting ipywidgets (from jupyter)
  Downloading ipywidgets-7.1.2-py2.py3-none-any.whl (68kB)
Collecting ipykernel (from jupyter)
  Downloading ipykernel-4.8.2-py3-none-any.whl (108kB)
Collecting notebook (from jupyter)
-
```

Use the example from the prior slide as content

Extending NGC Images

```
ubuntu@ip-172-31-7-211:~$ ipykernel-4.8.2 ipython-6.2.1 ipython-genutils-0.2.0 ipywidgets-7.1.2 jedi-0.11.1 jinja2-2.2.10 jsonschema-2.6.0 jupyter-1.0.0 jupyter-client-5.2.3 jupyter-console-5.2.0 jupyter-core-4.4.0 mistune-0.8.3 nbconvert-5.3.1 nbformat-4.4.0 notebook-5.4.1 pandocfilters-1.4.2 parso-0.1.1 pickleshare-0.7.4 prompt-toolkit-1.0.15 pygments-2.2.0 python-dateutil-2.7.0 pyzmq-17.0.0 qtconsole-4.3.1 simplegeneric-0.8.1 terminado-0.8.1 testpath-0.3.1 tornado-5.0.1 traitlets-4.3.2 wcwidth-0.1.7 widgetsnbextension-3.1.4
You are using pip version 9.0.1, however version 9.0.3 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
Removing intermediate container b9c14a05670f
--> b70170557b8e
Step 3/4 : WORKDIR /notebooks
Removing intermediate container 44a11df4fe6e
--> 79586757db8b
Step 4/4 : CMD jupyter notebook --allow-root --ip=0.0.0.0
--> Running in b91c0a2f389a
Removing intermediate container b91c0a2f389a
--> befe343b36d3
Successfully built befe343b36d3
Successfully tagged myimage:latest
ubuntu@ip-172-31-7-211:~$ docker images
REPOSITORY          TAG           IMAGE ID        CREATED         SIZE
myimage              latest        befe343b36d3   20 seconds ago  3GB
nvcr.io/nvidia/tensorflow  18.02-py3    57ae51ee8b74   5 weeks ago    2.91GB
ubuntu@ip-172-31-7-211:~$
```

Our new image is here!
myimage:latest

Extending NGC Images

```
ubuntu@ip-172-31-7-211:~$ docker run --runtime=nvidia -p 8888:8888 --rm -ti myimage:late
st
=====
== TensorFlow ==
=====

NVIDIA Release 18.02 (build 337102)

Container image Copyright (c) 2018, NVIDIA CORPORATION. All rights reserved.
Copyright 2017 The TensorFlow Authors. All rights reserved.

Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.
NVIDIA modifications are covered by the license terms that apply to the underlying project or file.

[I 21:24:49.047 NotebookApp] Writing notebook server cookie secret to /root/.local/share/jupyter/runtime/notebook_cookie_secret
[I 21:24:49.348 NotebookApp] Serving notebooks from local directory: /notebooks
[I 21:24:49.348 NotebookApp] 0 active kernels
[I 21:24:49.348 NotebookApp] The Jupyter Notebook is running at:
[I 21:24:49.348 NotebookApp] http://0.0.0.0:8888/?token=46edb99a6d0fddd72a0cb463ab5f4bc
ba1334fd100e0fd5
[I 21:24:49.348 NotebookApp] Use Control-C to stop this server and shut down all kernels
(twice to skip confirmation).
[W 21:24:49.349 NotebookApp] No web browser found: could not locate runnable browser.
[ C 21:24:49.349 NotebookApp]
```

Extending NGC Images

```
ubuntu@ip-172-31-7-211: ~
=====
NVIDIA Release 18.02 (build 337102)
Container image Copyright (c) 2018, NVIDIA CORPORATION.
Copyright 2017 The TensorFlow Authors. All rights reserved.

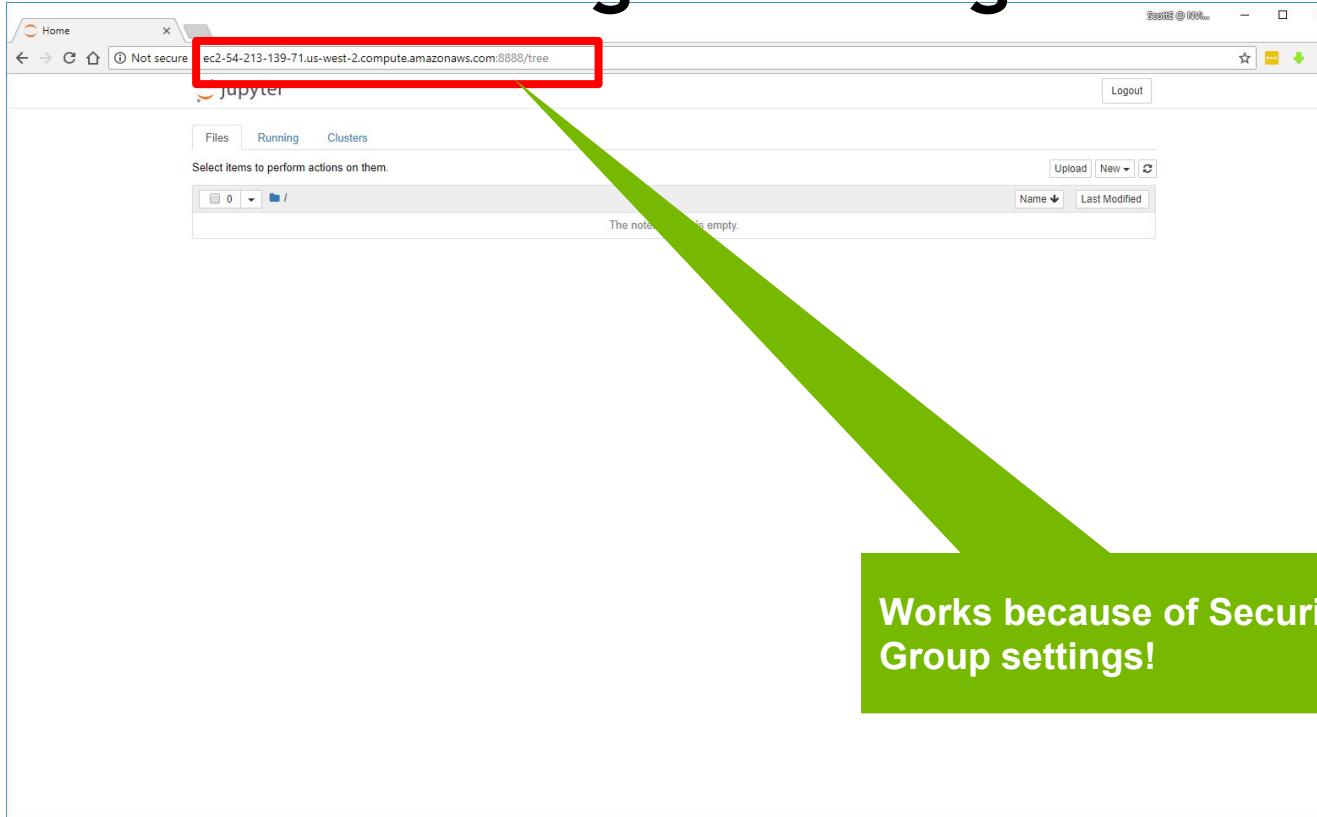
Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.
NVIDIA modifications are covered by the license terms that apply to the underlying project or file.

[I 21:24:49.047 NotebookApp] Writing notebook server cookie secret to /root/.local/share/jupyter/runtime/notebook_cookie_secret
[I 21:24:49.348 NotebookApp] Serving notebooks from local directory: /notebooks
[I 21:24:49.348 NotebookApp] 0 active kernels
[I 21:24:49.348 NotebookApp] The Jupyter Notebook is running at:
[I 21:24:49.348 NotebookApp] http://0.0.0.0:8888/?token=46edb99a6d0fddd72a0cb463ab5f4bcba1334fd100e0fd5
[I 21:24:49.348 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[W 21:24:49.349 NotebookApp] No web browser found: could not locate runnable browser.
[C 21:24:49.349 NotebookApp]

Copy/paste this URL into your browser when you connect for the first time,
to login with a token:
http://0.0.0.0:8888/?token=46edb99a6d0fddd72a0cb463ab5f4bcba1334fd100e0fd5
```

Use the P3 instance FQDN

Extending NGC Images



Works because of Security
Group settings!

Using Your Own Data

Using Your Own Data

How do you access actual data?

Many options for accessing data

- Object Storage via S3 Buckets
- Block Storage via Elastic Block Storage (EBS)
- File Storage via Elastic File Storage (EFS)
- 3rd party storage choices

We'll pick Elastic File Storage (EFS) for simplicity

Using Your Own Data

What to do

- Before we can use EFS we must (this should look familiar)
 - Secure network access with a security profile

This is the same procedure as with our EC2 instance (so we'll skip it)
(Enable NFS as the port)

Using Your Own Data

What to do

Our challenge:

- Mount the storage to our P3 instance
- Launch our container accessing the new data

Using Your Own Data

The screenshot shows the AWS Elastic File System Management console. The left sidebar lists services like EFS, EC2, and Storage Gateway. The main area is a grid of service categories. The 'Storage' category is expanded, and the 'EFS' service is highlighted with a red box. Other storage services listed include Storage Gateway, CloudWatch Metrics, AWS Auto Scaling, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Systems Manager, Trusted Advisor, and Managed Services.

Compute	Developer Tools	Machine Learning	AR & VR
EC2	CodeStar	Amazon SageMaker	Amazon Sumerian
Lightsail	CodeCommit	Amazon Comprehend	
Elastic Container Service	CodeBuild	AWS DeepLens	
Lambda	CodeDeploy	Amazon Lex	
Batch	CodePipeline	Machine Learning	
Elastic Beanstalk	Cloud9	Amazon Polly	
	X-Ray	Rekognition	
		Amazon Transcribe	
		Amazon Translate	
Storage	Management Tools	Analytics	Customer Engagement
EFS	CloudWatch	Athena	Amazon Connect
(CloudWatch Metrics)	AWS Auto Scaling	EMR	Pinpoint
Storage Gateway	CloudFormation	CloudSearch	Simple Email Service
Database	CloudTrail	Elasticsearch Service	
Relational Database Service	Config	Kinesis	
DynamoDB	OpsWorks	QuickSight	
ElastiCache	Service Catalog	Data Pipeline	
Amazon Redshift	Systems Manager	AWS Glue	
	Trusted Advisor		
	Managed Services		
Migration	Media Services	Security, Identity & Compliance	Business Productivity
AWS Migration Hub	Elastic Transcoder	IAM	Alexa for Business
Application Discovery Service	Kinesis Video Streams	Cognito	Amazon Chime
Database Migration Service	MediaConvert	GuardDuty	WorkDocs
Server Migration Service	MediaLive	Inspector	WorkMail
Snowball	MediaPackage	Amazon Macie	
	MediaStore	AWS Single Sign-On	

Using Your Own Data

The screenshot shows the AWS Elastic File System Manager console. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user information (scotta @ 6115-2050-7156, Oregon, Support). The main menu on the left has 'File systems' selected. The central area displays 'File systems' with a 'Create file system' button and an 'Actions' dropdown. A table lists existing file systems, with one row for 'ScottE-GTC-EFS' highlighted by a red box. The 'Other details' section shows Owner ID (611520507156), Life cycle state (Available), Performance mode (General Purpose), and Encrypted (No). The 'Tags' section shows a single tag 'Name: ScottE-GTC-EFS'. The 'File system access' section includes a 'DNS name' (fs-20c54f89.efs.us-west-2.amazonaws.com) and a 'Manage file system access' button highlighted by a red box. The 'Mount targets' section lists three targets across three Availability Zones (us-west-2b, us-west-2a, us-west-2c) with their respective IP addresses, mount target IDs, network interface IDs, security groups, and life cycle states.

Name	File system ID	Metered size	Number of mount targets	Creation date
ScottE-GTC-EFS	fs-20c54f89	22.0 KIB	3	2018-03-06T06:29:53Z

VPC	Availability Zone	Subnet	IP address	Mount target ID	Network interface ID	Security groups	Life cycle state
vpc-bffdd9d9 (default)	us-west-2b	subnet-351e7653 (default)	172.31.31.164	fsmt-c274e46b	eni-32116216	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available
	us-west-2a	subnet-3a69e072 (default)	172.31.36.172	fsmt-c374e46a	eni-8bef73bc	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available
	us-west-2c	subnet-773e0b2c (default)	172.31.6.45	fsmt-c574e46c	eni-f5498cf3	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available

Using Your Own Data

The screenshot shows the AWS Elastic File System Management console. The URL in the browser is <https://us-west-2.console.aws.amazon.com/efs/home?region=us-west-2#/manageaccess/fs-20c54f89>. The page title is "Elastic File System > File systems > Manage file system access for ScottE-GTC-EFS". The left sidebar has "File systems" selected. The main content area shows the "Manage mount targets" section. It explains that instances connect to a file system by using mount targets you create. It recommends creating a mount target in each of your VPC's Availability Zones so that EC2 instances across your VPC can access the file system. A table lists three mount targets:

	Availability Zone	Subnet	IP address	Security groups	Life cycle state
1	us-west-2a	subnet-3a69e072 (default)	172.31.36.172	sg-02eb047c - GTCSecurityGroup x sg-519fee2e - ScottE-EFS x	Available
2	us-west-2b	subnet-351e7653 (default)	172.31.31.164	sg-02eb047c - GTCSecurityGroup x sg-519fee2e - ScottE-EFS x	Available
3	us-west-2c	subnet-773e0b2c (default)	172.31.6.45	sg-02eb047c - GTCSecurityGroup x sg-519fee2e - ScottE-EFS x	Available

The "Security groups" column for the first mount target is highlighted with a red box. At the bottom right of the table are "Cancel" and "Save" buttons.

Using Your Own Data

The screenshot shows the AWS Elastic File System Manager console. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user information (scotta @ 6115-2050-7156, Oregon, Support). The main menu on the left has 'File systems' selected. The central area displays a table of file systems with one entry:

	Name	File system ID	Metered size	Number of mount targets	Creation date
ScottE-GTC-EFS	fs-20c54f89	22.0 KIB	3		2018-03-06T06:29:53Z

Below the table, the 'Other details' section shows:

- Owner ID: 611520507156
- Life cycle state: Available
- Performance mode: General Purpose
- Encrypted: No

The 'Tags' section contains a single tag: Name: ScottE-GTC-EFS.

The 'File system access' section shows the DNS name: fs-20c54f89.efs.us-west-2.amazonaws.com. Below it, the 'Amazon EC2 mount instructions' and 'AWS Direct Connect mount instructions' are listed, with the former being highlighted by a red box.

The 'Mount targets' section lists three targets across three Availability Zones (us-west-2b, us-west-2a, us-west-2c) in a VPC (vpc-bffdd9d9). Each target has a unique IP address and is associated with specific network interface IDs and security groups.

Using Your Own Data

The screenshot shows the AWS Elastic File System Management console. A modal window titled "Amazon EC2 mount instructions" is displayed over the main interface. The modal contains two sections: "Amazon EC2 mount instructions" and "Mounting your file system". The "Mounting your file system" section is highlighted with a red rectangle and contains the following steps:

1. Open an SSH client and connect to your EC2 instance. ([find out how to connect](#))
2. Create a new directory on your EC2 instance, such as "efs".
 - `sudo mkdir efs`
3. Mount your file system using the DNS name. ([Mounting considerations](#))
 - `sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2 fs_20c54f89.efs.us-west-2.amazonaws.com:/ efs`

If you are unable to connect, please see our [troubleshooting documentation](#).

Close

The main interface shows a table of "File systems" with one entry:

VPC	Availability Zone	Subnet	IP address	Mount target ID	Network interface ID	Security groups	Life cycle state
vpc-bffdd9d9 (default)	us-west-2b	subnet-351e7653 (default)	172.31.31.164	fsmt-c274e46b	eni-32116216	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available
	us-west-2a	subnet-3a69e072 (default)	172.31.36.172	fsmt-c374e46a	eni-8bef73bc	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available
	us-west-2c	subnet-773e0b2c (default)	172.31.6.45	fsmt-c574e46c	eni-f5498cf3	sg-519fee2e - ScottE-EFS sg-02eb047c - GTCSecurityGroup	Available

Using Your Own Data

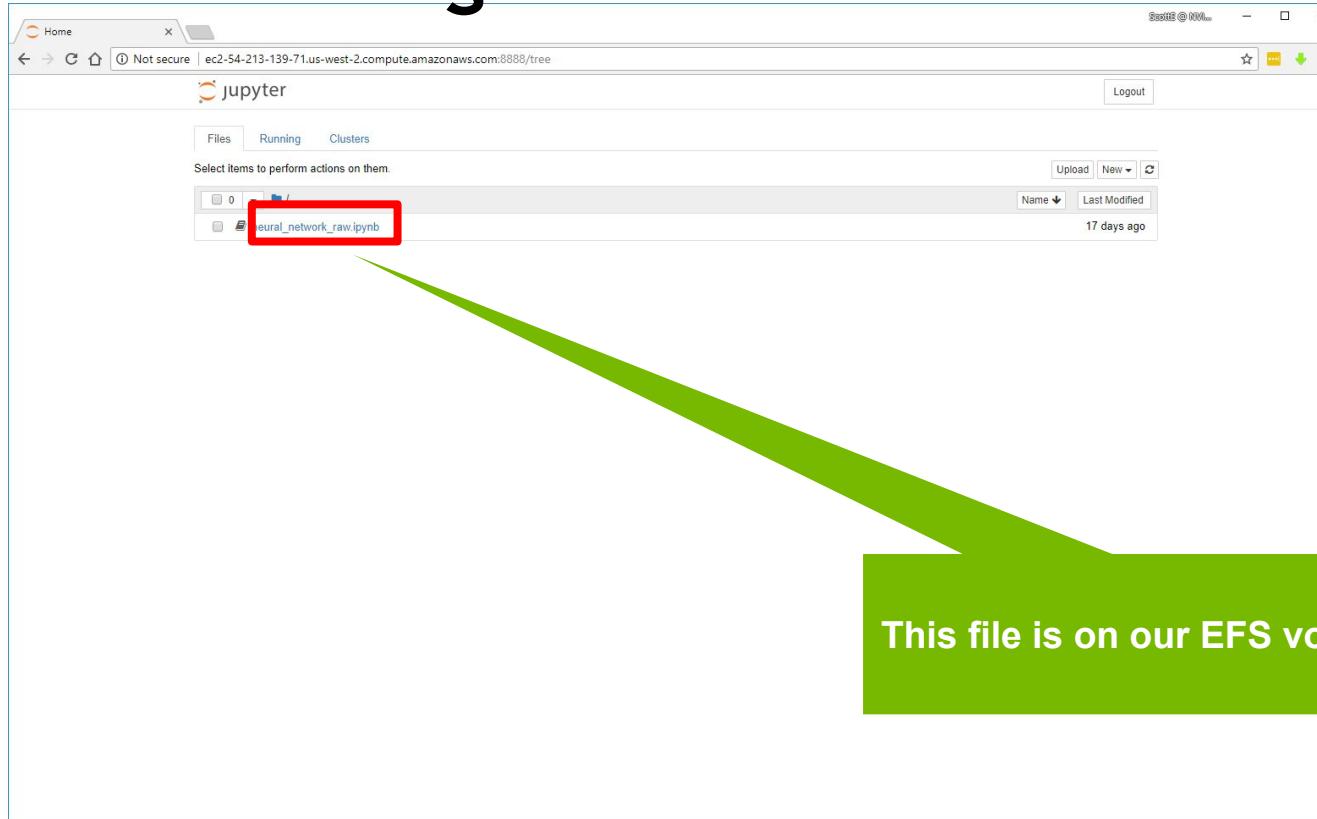
```
ubuntu@ip-172-31-7-211:~$ sudo mkdir /efs
ubuntu@ip-172-31-7-211:~$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,
hard,timeo=600,retrans=2 fs-20c54f89.efs.us-west-2.amazonaws.com:/ /efs
```

Using Your Own Data

```
ubuntu@ip-172-31-7-211:~$ docker run --runtime=nvidia -v /efs:/notebooks -p 8888:8888 --rm -ti myimage:latest  
=====  
= TensorFlow =  
=====  
  
NVIDIA Release 18.02 (build 337102)  
  
Container image Copyright (c) 2018, NVIDIA CORPORATION. All rights reserved.  
Copyright 2017 The TensorFlow Authors. All rights reserved.  
  
Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.  
NVIDIA modifications are covered by the license terms that apply to this file or file.  
  
[I 21:47:25.991 NotebookApp] Writing notebook server cookie secret to /jupyter/runtime/notebook_cookie_secret  
[I 21:47:26.290 NotebookApp] Serving notebooks from local directory  
[I 21:47:26.291 NotebookApp] 0 active kernels  
[I 21:47:26.291 NotebookApp] The Jupyter Notebook is running at: http://0.0.0.0:8888/?token=7b61d40996c066b44  
[I 21:47:26.291 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).  
[W 21:47:26.291 NotebookApp] No web browser found: could not locate runnable browser.  
[C 21:47:26.291 NotebookApp]
```

Remember “/notebooks” from our Dockerfile

Using Your Own Data



Using Your Own Data

neural_network_raw

Not secure | ec2-54-213-139-71.us-west-2.compute.amazonaws.com:8888/notebooks/neural_network_raw.ipynb

jupyter neural_network_raw (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted Python 3

Run

Neural Network Example

Build a 2-hidden layers fully connected neural network (a.k.a multilayer perceptron) with TensorFlow.

- Author: Aymeric Damien
- Project: <https://github.com/aymericdamien/TensorFlow-Examples/>

Neural Network Overview

input layer hidden layer 1 hidden layer 2 output layer

MNIST Dataset Overview

This example is using MNIST handwritten digits. The dataset contains 60,000 examples for training and 10,000 examples for testing. The digits have been size-normalized and centered in a fixed-size image (28x28 pixels) with values from 0 to 1. For simplicity, each image has been flattened and converted to a 1-D numpy array of 784 features (28*28).

Where do we go from here?

Framework Documentation

<https://ngc.nvidia.com/>

The screenshot shows a web interface for managing Docker containers. On the left, a sidebar lists repositories under categories: **nvidia**, **hpc**, **nvidia-hpcvis**, and **partners**. The **nvidia** category is expanded, showing sub-repositories: **caffe**, **caffe2**, **cntk**, **cuda**, **digits**, **mxnet**, **pytorch**, **tensorflow**, **tensorrt**, **theano**, and **torch**. The **tensorflow** repository is selected, indicated by a green border around its card.

nvidia/tensorflow

`docker pull nvcr.io/nvidia/tensorflow:18.01-py2`

What is TensorFlow?

TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture lets you deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device without rewriting code.

TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well.

Running TensorFlow

Container Documentation

<http://docs.nvidia.com/deeplearning/dgx/>

The screenshot shows a web browser displaying the "DEEP LEARNING DGX DOCUMENTATION" page from docs.nvidia.com. The URL in the address bar is docs.nvidia.com/deeplearning/dgx/mxnet-release-notes/rel_18.01.html#rel_18.01. The page header includes the NVIDIA logo and "ACCELERATED COMPUTING". The left sidebar has a navigation menu with links like "Getting Started", "MXNet Release Notes", and "MXNet Release 18.01" (which is currently selected). The main content area starts with a section titled "MXNet Release 18.01". It states that the NVIDIA container image of MXNet, release 18.01, is available. It mentions that the container image version 18.01 is based on [MXNet 1.0.0](#). Below this, there's a "Contents of MXNet" section which lists various components included in the container image. It includes links to Ubuntu 16.04, Python 2.7 and 3.5, NVIDIA CUDA 9.0.176, cuDNN 7.0.5, NCCL 2.1.2, and ONNX exporter for CNN classification models. A note indicates that the ONNX exporter is continuously improved. The page also mentions Driver Requirements (CUDA 9) and Key Features and Enhancements, listing additions like Python 3 support and enhanced convolutional operations.

MXNet Release 18.01

The NVIDIA container image of MXNet, release 18.01, is available.

[MXNet](#) container image version 18.01 is based on [MXNet 1.0.0](#).

Contents of MXNet

This container image contains the complete source of the version of MXNet in `/opt/mxnet`. It is pre-built and The container also includes the following:

- Ubuntu 16.04

Note: Container image 18.01-py2 contains [Python 2.7](#); 18.01-py3 contains [Python 3.5](#).

- NVIDIA CUDA 9.0.176 including [CUDA® Basic Linear Algebra Subroutines library™ \(cuBLAS\)](#) 9.0.282
- [NVIDIA CUDA® Deep Neural Network library™ \(cuDNN\)](#) 7.0.5
- [NCCL](#) 2.1.2 (optimized for [NVLink™](#))
- ONNX exporter for CNN classification models

Note: The ONNX exporter is being continuously improved. You can try the latest changes by pulling from

- Amazon Labs Sockeye sequence-to-sequence framework (for machine translation)

Driver Requirements

Release 18.01 is based on CUDA 9, which requires [NVIDIA Driver](#) release 384 or later.

Key Features and Enhancements

This MXNet release includes the following key features and enhancements.

- Addition of Python 3 package
- Enhanced-performance cuDNN-based batched 1D convolutions (merged to upstream)
- Added [MxNet-to-ONNX](#) exporter for classification of CNN models (tested with LeNet-5, ResNet-50, etc.)
- Added the Sockeye sequence-to-sequence framework, along with a German-to-English translation mode to reproduce the [OpenNMT reference model](#) when trained until convergence.
- Latest version of cuBLAS
- Latest version of cuDNN
- Latest version of NCCL
- Ubuntu 16.04 with December 2017 updates

NGC + AWS

<http://docs.nvidia.com/ngc/ngc-aws-setup-guide/>

The screenshot shows a documentation page with a dark header bar containing the NVIDIA logo and the text "ACCELERATED COMPUTING" and "NVIDIA GPU CLOUD DOCUMENTATION". The main content area has a light background. On the left, there is a sidebar menu with the following structure:

- Using NGC with AWS Setup Guide**
 - 1. Introduction to Using NGC with AWS
 - 2. Preliminary Setup
 - 2.1. Setting Up Your AWS Key Pair
 - 2.2. Setting Up Security Groups for the EC2 Instance
 - 2.3. Setting Up Security Groups for EFS
 - 3. Launching a VM Instance from the AWS Console
 - 3.1. Logging In and Selecting the AWS Zone
 - 3.2. Selecting the NVIDIA Deep Learning AMI
 - 3.3. Selecting an Amazon EC2 P3 Instance Type and Configuring Instance Settings
 - 3.4. Launching Your VM Instance
 - 3.5. Connecting to Your VM Instance
 - 3.6. Starting, Stopping, and Terminating Your VM Instance
 - 4. Launching a VM Instance Using AWS CLI
 - 4.1. Setting Up Environment Variables
 - 4.2. Launching Your VM Instance
 - 4.3. Connecting To Your VM Instance
 - 4.4. Starting, Stopping, and Terminating Your VM Instance
 - 5. Using the Amazon Elastic File System (EFS) for Persistent Data Storage
 - 5.1. Creating an EFS
 - 5.2. Mounting an EFS

Using NGC with AWS Setup Guide

Table of Contents

- [1. Introduction to Using NGC with AWS](#)
- [2. Preliminary Setup](#)
 - [2.1. Setting Up Your AWS Key Pair](#)
 - [2.2. Setting Up Security Groups for the EC2 Instance](#)
 - [2.3. Setting Up Security Groups for EFS](#)
- [3. Launching a VM Instance from the AWS Console](#)
 - [3.1. Logging In and Selecting the AWS Zone](#)
 - [3.2. Selecting the NVIDIA Deep Learning AMI](#)
 - [3.3. Selecting an Amazon EC2 P3 Instance Type and Configuring Instance Settings](#)
 - [3.4. Launching Your VM Instance](#)
 - [3.5. Connecting to Your VM Instance](#)
 - [3.6. Starting, Stopping, and Terminating Your VM Instance](#)
- [4. Launching a VM Instance Using AWS CLI](#)
 - [4.1. Setting Up Environment Variables](#)
 - [4.2. Launching Your VM Instance](#)
 - [4.3. Connecting To Your VM Instance](#)
 - [4.4. Starting, Stopping, and Terminating Your VM Instance](#)
- [5. Using the Amazon Elastic File System \(EFS\) for Persistent Data Storage](#)
 - [5.1. Creating an EFS](#)
 - [5.2. Mounting an EFS](#)

NGC + Real Hardware

<http://docs.nvidia.com/ngc/ngc-titan-setup-guide/>

The screenshot shows a web browser displaying the NVIDIA GPU Cloud Documentation for the NGC-Titan-Setup-Guide. The URL in the address bar is docs.nvidia.com/ngc/ngc-titan-setup-guide/index.html. The page title is "NVIDIA GPU CLOUD DOCUMENTATION". On the left, there's a sidebar with the NVIDIA logo and a navigation menu:

- ACCELERATED COMPUTING
- NVIDIA GPU Cloud (NGC)
- Using NGC with Your NVIDIA TITAN PC Setup Guide** (highlighted)
- 1. Introduction
- ▷ 2. Installing the NVIDIA Driver
- ▷ 3. Installing Docker and the Docker Utility Engine for NVIDIA GPUs
- 4. Using NGC Containers

The main content area starts with a section titled "Using NGC with Your NVIDIA TITAN PC Setup Guide". It includes an "Introduction" paragraph and a "Prerequisites" section. Below that is a "Installing the NVIDIA Driver" section with a "Setting Up the Driver Repository" sub-section. A command-line snippet at the bottom shows how to install the CUDA repository key:

```
sudo apt-get install -y apt-transport-https curl  
cat <<EOF | sudo tee /etc/apt/sources.list.d/cuda.list > /dev/null
```

NGC Everywhere

<http://docs.nvidia.com/ngc/>

The screenshot shows a web browser displaying the NVIDIA GPU Cloud Documentation. The URL in the address bar is docs.nvidia.com/ngc/index.html. The page title is "NVIDIA GPU CLOUD DOCUMENTATION". On the left, there is a navigation sidebar with the following menu items:

- NVIDIA
- ACCELERATED COMPUTING
- NVIDIA GPU Cloud (NGC)
 - Introduction
 - Getting Started
 - NGC with Amazon Web Services (AWS)
 - NVIDIA Volta Deep Learning AMI Release Notes
 - Using NGC with AWS Setup Guide
 - NGC with NVIDIA TITAN PCs
 - Using NGC with Your NVIDIA TITAN PC Setup Guide
 - NGC Container Registry
 - NGC Container User Guide

The "Getting Started" item in the sidebar is highlighted with a dark background. The main content area contains three sections:

- NVIDIA GPU Cloud (NGC)**
 - Introduction**

This introduction provides an overview of NGC and how to use it.
 - Getting Started**

This NGC Getting Started Guide provides step-by-step instructions for how to
- NGC with Amazon Web Services (AWS)**
 - NVIDIA Volta Deep Learning AMI Release Notes**

This document describes the current status, information about included softw
 - Using NGC with AWS Setup Guide**

This Using NGC with AWS Setup Guide explains how to set up an NVIDIA Volt
- NGC with NVIDIA TITAN PCs**
 - Using NGC with Your NVIDIA TITAN PC Setup Guide**

This Setup Guide explains how to set up an NVIDIA TITAN PC for running NG

NGC Status

<https://ngc.statuspage.io/>

The screenshot shows the NGC Statuspage.io interface. At the top, there's a navigation bar with icons for back, forward, refresh, and a lock symbol labeled "Secure". The URL "https://ngc.statuspage.io" is displayed. Below the header, the NVIDIA GPU CLOUD logo is on the left, and a green "SUBSCRIBE TO UPDATES" button is on the right. A green banner at the top states "All Systems Operational" and "Refreshed less than 1 minute ago". To the right of the banner, text indicates "Uptime over the past 90 days. View historical uptime." Below this, a chart titled "Cloud Services" shows a 100% uptime bar from "90 days ago" to "Today". The main content area features a "System Metrics" section with three cards: "Container Registry" (100% uptime), "Authentication Service" (100% uptime), and another partially visible card. Each metric card includes a timeline from 12:00 to 09:00 on 25. Feb. The bottom right corner has the NVIDIA logo.

All Systems Operational
Refreshed less than 1 minute ago

Uptime over the past 90 days. [View historical uptime.](#)

Cloud Services

90 days ago ————— 100.0 % uptime ————— Today

Operational

Container Registry 100%

100
50
0

12.00 15.00 18.00 21.00 25. Feb 03.00 06.00 09.00

Authentication Service 100%

100
--

Day | Week | Month

NVIDIA

Help from Humans

<https://devtalk.nvidia.com/>

The screenshot shows a web browser displaying the NVIDIA GPU Cloud (NGC) Users forum at <https://devtalk.nvidia.com/default/board/200/nvidia-gpu-cloud-ngc-users/>. The page features a navigation bar with links to Home, CUDA ZONE, Forums, Accelerated Computing, NVIDIA GPU Cloud (NGC) Users, Downloads, Training, Ecosystem, and Forums. Below the navigation bar, a breadcrumb trail indicates the current location: Home > CUDA ZONE > Forums > Accelerated Computing > NVIDIA GPU Cloud (NGC) Users. A "Forum Statistics" link is also present. The main content area is titled "NVIDIA GPU Cloud (NGC) Users". It lists several categories with their respective counts of topics and comments:

Category	Topics	Comments
Announcements	4 Topics	4 Comments
FAQ	0 Topics	3 Comments
NGC Account	6 Topics	29 Comments
TITAN	4 Topics	7 Comments
AWS AMI	12 Topics	41 Comments
Docker and NVIDIA Docker	4 Topics	18 Comments
Feature Requests	5 Topics	11 Comments

On the right side, there are two vertical columns: "Popular" and "Latest Topics". The "Popular" column lists recent activity in various forums, while the "Latest Topics" column lists the most recent posts across different categories.

Thank You



Scott Ellis - scotte@nvidia.com

Jeff Weiss - jweiss@nvidia.com

