Using docker on AWS for bioinformatics workshops

Nicolas Delhomme, May 25th, 2016
CSC, Helsinki, Finland

Umeå Plant Science Centre

- 200+ scientists
- 42 different nationalities
- leading centre in plant physiology with a strong expertise with model (Arabidopsis) and non model "trees" (spruce and poplars)

UPSC bioinformatics

- UPSCb: 4 persons (hiring a fifth)
  - 2 person (Bastian Schiffthaler and myself) involved in training / teaching
- Access to Swedish HPC and a local cluster (200 cores, 100TB storage)
- Processed > 2,000 HTS samples in the last 2 years

Workshop's aim

- Aim: In this workshop, you will learn about Amazon Web Services (AWS) and Docker, and how to apply these resources for your own workshops

Learning Objectives

- By the end of the workshop, you will be able to:
  - Navigate the AWS Console
  - Manage AWS instances (prepare, secure, start, stop and terminate)
  - Deploy docker images on AWS instances

Outline

The goal is to reproduce a real course setup

- Short presentation of the AWS console
- Accessing the AWS console
- Creating, accessing and configuring an AWS instance
- Deploying a docker image on the AWS instance
- Deploying a website on the AWS instance
- Performing an HTS analysis on the AWS instance
- Summing up the AWS + docker workshop and a few info on how to get AWS credit grants

Expertise

- Used for our own research
- Used for teaching under-graduate (training in functional genomics)
- Used in workshops:
  - EMBO Analysis of High-Throughput Sequencing Data, October 2015
  - EBI Advanced RNA-Seq and ChIP-Seq data analysis, April 2016

Website

- Connect to http://52.50.251.25:3000
- Navigate the site a little (but obviously you cannot connect to any instances yet :-))
- Mind your ID, you will be needing it later.

Connecting to AWS

- Use the credentials that were communicated to you
- Point your browser to the "Direct Signin Link"
- Connect using your User Name and password

Setting up the instance

Amazon Web Services

Compute
- **EC2** — Virtual Servers in the Cloud
- **EC2 Container Service** — Run and Manage Docker Containers
- **Elastic Beanstalk** — Run and Manage Web Apps
- **Lambda** — Run Code in Response to Events

Storage & Content Delivery
- **S3** — Scalable Storage in the Cloud
- **CloudFront** — Global Content Delivery Network
- **Elastic File System** PREVIEW — Fully Managed File System for EC2
- **Glacier** — Archive Storage in the Cloud
- **Snowball** — Large Scale Data Transport
- **Storage Gateway** — Hybrid Storage Integration

Database
- **RDS** — Managed Relational Database Service
- **DynamoDB** — Managed NoSQL Database
- **ElastiCache** — In-Memory Cache
- **Redshift** — Fast, Simple, Cost-Effective Data Warehousing
- **DMS** — Managed Database Migration Service

Networking
- **VPC** — Isolated Cloud Resources
- **Direct Connect** — Dedicated Network Connection to AWS
- **Route 53** — Scalable DNS and Domain Name Registration

Developer Tools
- **CodeCommit** — Store Code in Private Git Repositories
- **CodeDeploy** — Automate Code Deployments
- **CodePipeline** — Release Software using Continuous Delivery

Management Tools
- **CloudWatch** — Monitor Resources and Applications
- **CloudFormation** — Create and Manage Resources with Templates
- **CloudTrail** — Track User Activity and API Usage
- **Config** — Track Resource Inventory and Changes
- **OpsWorks** — Automate Operations with Chef
- **Service Catalog** — Create and Use Standardized Products
- **Trusted Advisor** — Optimize Performance and Security

Security & Identity
- **Identity & Access Management** — Manage User Access and Encryption Keys
- **Directory Service** — Host and Manage Active Directory
- **Inspector** — Analyze Application Security
- **WAF** — Filter Malicious Web Traffic
- **Certificate Manager** — Provision, Manage, and Deploy SSL/TLS Certificates

Analytics
- **EMR** — Managed Hadoop Framework
- **Data Pipeline** — Orchestration for Data-Driven Workflows
- **Elasticsearch Service** — Run and Scale Elasticsearch Clusters
- **Kinesis** — Work with Real-Time Streaming Data

Internet of Things
- **AWS IoT** — Connect Devices to the Cloud

Game Development
- **GameLift** — Deploy and Scale Session-based Multiplayer Games

Mobile Services
- **Mobile Hub** — Build, Test, and Monitor Mobile Apps
- **Cognito** — User Identity and App Data Synchronization
- **Device Farm** — Test Android, iOS, and Web Apps on Real Devices in the Cloud
- **Mobile Analytics** — Collect, View and Export App Analytics
- **SNS** — Push Notification Service

Application Services
- **API Gateway** — Build, Deploy and Manage APIs
- **AppStream** — Low Latency Application Streaming
- **CloudSearch** — Managed Search Service
- **Elastic Transcoder** — Easy-to-Use Scalable Media Transcoding
- **SES** — Email Sending and Receiving Service
- **SQS** — Message Queue Service
- **SWF** — Workflow Service for Coordinating Application Components

Enterprise Applications
- **WorkSpaces** — Desktops in the Cloud
- **WorkDocs** — Secure Enterprise Storage and Sharing Service
- **WorkMail** — Secure Email and Calendaring Service

Resource Groups

Learn more

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

Create a Group | **Tag Editor**

Additional Resources

Getting Started
Read our documentation or view our training to learn more about AWS.

AWS Console Mobile App
View your resources on the go with our AWS Console mobile app, available from Amazon Appstore, Google Play, or iTunes.

AWS Marketplace
Find and buy software, launch with 1-Click and pay by the hour.

AWS re:Invent Announcements
Explore the next generation of AWS cloud capabilities. See what's new

Service Health

All services operating normally.

Updated: May 23 2016 02:43:00 GMT+0300

Service Health Dashboard

An important page

- **Dashboard**
- Bills
- Cost Explorer
- Budgets
- Reports
- Cost Allocation Tags
- Payment Methods
- Payment History
- Consolidated Billing
- Preferences
- Credits
- Tax Settings
- DevPay

Billing & Cost Management Dashboard

Spend Summary — Cost Explorer

Welcome to the AWS Account Billing console. Your last month, month-to-date, and month-end forecasted costs appear below.

Current month-to-date balance for May 2016

$0.00

▸ Important Information about these Costs

☑ Include Subscription Charges

Top Free Tier Services by Usage — View all

Service	Month-to-date usage/Free Tier limit	Forecasted month-end usage/Free Tier limit
EC2 - Linux	15.47% (116.00/750 Hrs)	21.79% (163.45/750 Hrs)
EBS - Volumes	4.05% (1.22/30 GB-Mo)	5.71% (1.71/30 GB-Mo)
KMS - Requests	0.01% (2.00/20,000 Requests)	0.01% (2.82/20,000 Requests)

Month-to-Date Spend by Service — Bill Details

The chart below shows the proportion of costs spent for each service you use.

Month-to-Date Top Services by Spend	Amount
EC2	$0.00
DataTransfer	$0.00
kms	$0.00
Tax	$0.00
Total	**$0.00**

EC2 instance

EC2 Dashboard
Events
Tags
Reports
Limits

INSTANCES
Instances
Spot Requests
Reserved Instances
Scheduled Instances
Dedicated Hosts

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

AUTO SCALING
Launch Configurations
Auto Scaling Groups

COMMANDS
Command History
Documents

Resources

You are using the following Amazon EC2 resources in the EU West (Ireland) region:

- 1 Running Instances
- 0 Dedicated Hosts
- 1 Volumes
- 2 Key Pairs
- 0 Placement Groups
- 0 Elastic IPs
- 0 Snapshots
- 0 Load Balancers
- 3 Security Groups

Build and run distributed, fault-tolerant applications in the cloud with Amazon Simple Workflow Service.

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 EU West (Ireland) region

Service Health

Service Status:

EU West (Ireland):
This service is operating normally

Availability Zone Status:

eu-west-1a:
Availability zone is operating normally

eu-west-1b:
Availability zone is operating normally

eu-west-1c:
Availability zone is operating normally

Service Health Dashboard

Scheduled Events

EU West (Ireland):

No events

Account Attributes

Supported Platforms
VPC

Default VPC
vpc-3629ce52

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:

Tableau Server (10 users)

Provided by Tableau
Rating ★☆☆☆☆
Pay by the hour for Tableau software and AWS usage
View all Business Intelligence

SAP HANA One 244GB

Provided by SAP America, Inc
Rating ★★★★★
Pay by the hour for SAP HANA One 244GiB software and AWS usage
View all Business Intelligence

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

Step 1: Choose an Amazon Machine Image (AMI)

Cancel and Exit

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

1 to 22 of 22 AMIs

My AMIs

AWS Marketplace

Community AMIs

Free tier only

Amazon Linux AMI 2016.03.1 (HVM), SSD Volume Type - ami-b0ac25c3

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

Select

64-bit

Red Hat Enterprise Linux 7.2 (HVM), SSD Volume Type - ami-8b8c57f8

Red Hat
Free tier eligible

Red Hat Enterprise Linux version 7.2 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm

Select

64-bit

SUSE Linux Enterprise Server 12 SP1 (HVM), SSD Volume Type - ami-f4278487

SUSE Linux
Free tier eligible

SUSE Linux Enterprise Server 12 Service Pack 1 (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

Select

64-bit

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-f95ef58a

Ubuntu
Free tier eligible

Ubuntu Server 14.04 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

Select

64-bit

Microsoft Windows Server 2012 R2 Base - ami-29eb7e5a

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
☐	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate
☑	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
☐	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
☐	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
☐	General purpose	t2.large	2	8	EBS only	-	Low to Moderate
☐	General purpose	m4.large	2	8	EBS only	Yes	Moderate
☐	General purpose	m4.xlarge	4	16	EBS only	Yes	High
☐	General purpose	m4.2xlarge	8	32	EBS only	Yes	High
☐	General purpose	m4.4xlarge	16	64	EBS only	Yes	High

Cancel | Previous | Review and Launch | Next: Configure Instance Details

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

Step 3: Configure Instance Details

Purchasing option ⓘ ☐ Request Spot instances

Network ⓘ vpc-3629ce52 (172.31.0.0/16) (default) — Create new VPC

Subnet ⓘ No preference (default subnet in any Availability Zone) — Create new subnet

Auto-assign Public IP ⓘ Use subnet setting (Enable)

IAM role ⓘ None — Create new IAM role

Shutdown behavior ⓘ Stop

Enable termination protection ⓘ ☑ Protect against accidental termination

Monitoring ⓘ ☐ Enable CloudWatch detailed monitoring
Additional charges apply.

Tenancy ⓘ Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy.

▼ Advanced Details

User data ⓘ ◉ As text ○ As file ☐ Input is already base64 encoded

```
#!/bin/bash
curl https://get.docker.com | sh
sudo usermod -a -G docker ubuntu
sudo service docker start
```

Cancel | Previous | Review and Launch | Next: Add Storage

Configure Instance Advanced details

While creating the instance, we provide the following script to be executed at startup.

```
#!/bin/bash
curl https://get.docker.com | sh
sudo usermod -a -G docker ubuntu
sudo service docker start
```

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

Step 4: Add Storage

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.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-02012b28	8	General Purpose SSD (GP2)	100 / 3000	N/A	☑	Not Encrypted

Add New Volume

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.

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

Step 5: Tag Instance

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	Helsinki2016-Nico	⊗

Create Tag (Up to 10 tags maximum)

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

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 group name: launch-wizard-3

Description: launch-wizard-3 created 2016-05-24T10:04:37.909+03:00

Type	Protocol	Port Range	Source
SSH	TCP	22	Anywhere 0.0.0.0/0

Add Rule

Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Create Security Group | Actions

Filter by tags and attributes or search by keyword | 1 to 5 of 5

Name	Group ID	Group Name	VPC ID	Description
	sg-b2ed44d5	Cloud-vm-helsinki-2016	vpc-3629ce52	Cloud-vm-helsinki-2016

Security Group: sg-b2ed44d5

Description | **Inbound** | Outbound | Tags

193.167.0.0/16

Edit

Type	Protocol	Port Range	Source
Custom TCP Rule	TCP	9050	[redacted]
SSH	TCP	22	[redacted]
Custom TCP Rule	TCP	8050	[redacted]
Custom TCP Rule	TCP	3000	[redacted]

Create Security Group | Actions

Filter by tags and attributes or search by keyword | 1 to 5 of 5

Name	Group ID	Group Name	VPC ID	Description
	sg-b2ed44d5	Cloud-vm-helsinki-2016	vpc-3629ce52	Cloud-vm-helsinki-2016

Security Group: sg-b2ed44d5

Description | Inbound | **Outbound** | Tags

Edit

Type	Protocol	Port Range	Destination
All traffic	All	All	0.0.0.0/0

Launch Status

Your instances are now launching

The following instance launches have been initiated: i-08ca7eee5b0cab870 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.

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)

Connecting to the instance

- Save the ssh key provided in the website
- Connect using ssh:

```
ssh -i [path-to-key] ubuntu@IP
```

Initial setup

Initial instance configuration

This is just and initial setup:

```
sudo locale-gen UTF-8
```

Here we check that the startup script worked:

```
id
```

```
service docker status
```

Next, we get the course docker image (it is in an S3 bucket, publicly available)

```
wget https://s3-eu-west-1.amazonaws.com/helsinki2016/helsinki2016-light.tgz
```

And load it in our instance

```
zcat helsinki2016-light.tgz | docker load
```

Finally, we start the docker (use your ID to replace the [ID] token)

```
docker run -p 90[ID]:443 -p 80[ID]:80 -d cloud-wm:helsinki2016-light
```

and check that the docker image is running.

```
docker ps -a
```

Next, deploy the course website

Course website setup

This is to replicate our usual training setup, where we use a website for attendees to connect to the different services (terminal, apache, *etc.*)

We first clone the repository

```
git clone https://github.com/ekorpela/cloud-vm-workshop.git
```

Then we install the tool to run the website (nodejs)

```
sudo apt-get install nodejs
```

To avoid unintended interruption, we use screen (a window manager that runs on the server and not on your computer).

```
screen -S website
```

In that screen session, we start the website

```
cd cloud-vm-workshop/materials/NicolasDelhomme/website/bin
```

```
nodejs www
```

Once this is done, we exit the screen using `Ctrl+A D`

You can now connect to your course website

- simply point your browser to http://[IP]:3000 where [IP] is your instance public IP

Let's now be the student!

- Click on"Connect to the server"
- Next to you name, select "Connect to Terminal"
- Accept (press Enter) the host (localhost) and port (22) default. The user and password are "training"

Once you are logged in

- 1. Use the tool FastQC (fastqc on the command line) to analyse the data in the "share" directory

- 2. On the website, click on the "Connect to Apache 2" link next to your name and navigate to your FastQC output directory. Have a look at the html reports.

Back to reality ;-)

AWS | Services | Edit | h2016 @ ndelhomme | Ireland | Support

EC2 Dashboard
Events
Tags
Reports
Limits

INSTANCES
Instances
Spot Requests
Reserved Instances
Scheduled Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

NETWORK & SECURITY
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces

Launch Instance | Connect | Actions

Actions menu:
- Connect
- Get Windows Password
- Launch More Like This
- Instance State ▸ (Start, Stop, Reboot, Terminate)
- Instance Settings ▸
- Image ▸
- Networking ▸
- CloudWatch Monitoring ▸

Filter by tags and attributes or s

1 to 2 of 2

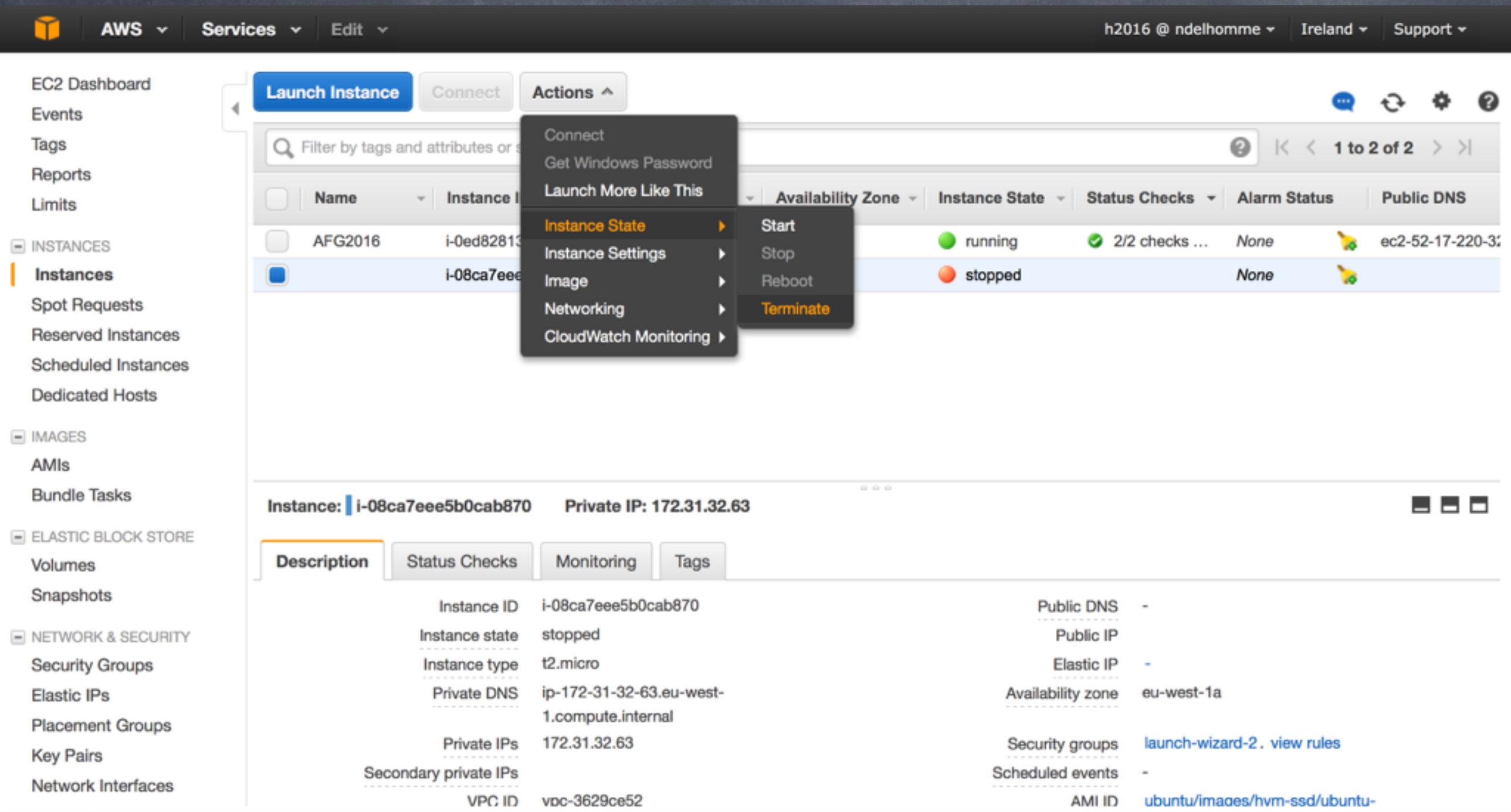
Name	Instance I	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
AFG2016	i-0ed82813		running	2/2 checks ...	None	ec2-52-17-220-3
	i-08ca7eee		running	2/2 checks ...	None	ec2-52-51-38-76.

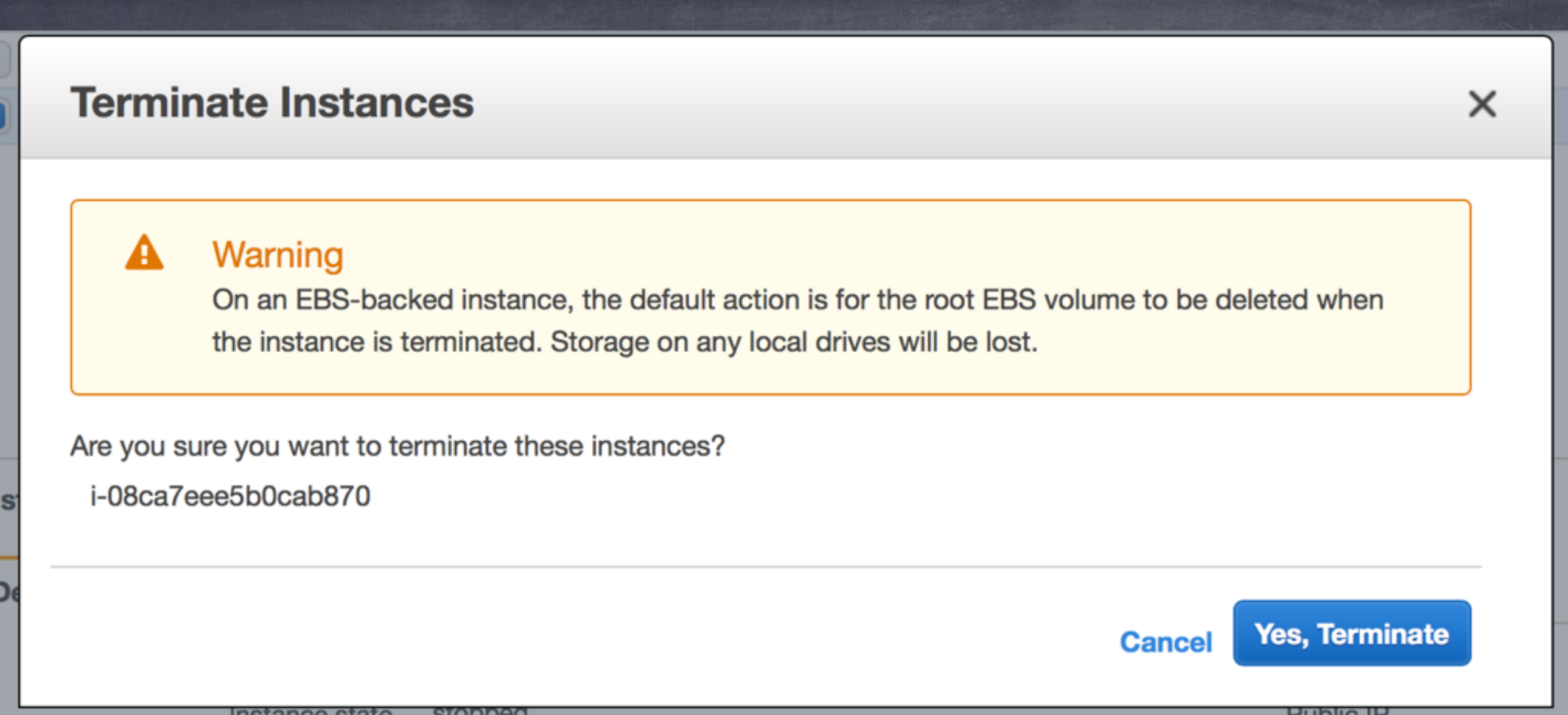
Instance: i-08ca7eee5b0cab870 **Public DNS:** ec2-52-51-38-76.eu-west-1.compute.amazonaws.com

Description | Status Checks | Monitoring | Tags

Instance ID	i-08ca7eee5b0cab870	Public DNS	ec2-52-51-38-76.eu-west-1.compute.amazonaws.com
Instance state	running	Public IP	52.51.38.76
Instance type	t2.micro	Elastic IP	-
Private DNS	ip-172-31-32-63.eu-west-1.compute.internal	Availability zone	eu-west-1a
Private IPs	172.31.32.63	Security groups	launch-wizard-2. view rules
Secondary private IPs		Scheduled events	No scheduled events

Terminate the instance





AWS is very verbose, which minimises the risks (also everything that occur costs is clearly indicated)

Summing up

- What's your impression?
- All the material is available. The Dockerfile is available from the website. I can let it run a while (free tier + some AWS educational grants)

Our training setup

- We use the following setup for RNA-Seq courses with up to 40 participants
- t2.micro (1CPU, 1GB, low to moderate bandwidth) for the website, we leave it running after the course for user to download the materials
- m4.10xlarge (40CPU, 160GB, high bandwidth) for the computing tasks (this server we stop in the evenings)
- d2.xlarge (4CPU, 30GB, high bandwidth, 6TB storage) that we use as an NFS server (easier to set up than S3)
- COSTS: 200$ for a 4 days course + 50$ for the original test / setup

AWS grants

- 2 types:
  - AWS Educate: https://aws.amazon.com/education/awseducate/: students and teacher can apply anytime - limited renewable yearly credits (max 200)
  - AWS Research: https://aws.amazon.com/research-credits/: "3. Train a broader community on the usage of cloud for research workloads via workshops or tutorials." Application are evaluated 4 times a year and credits are granted for a year (we got 2,000 for running our initial EMBO course in October 2015)

Trainer's material sharing, Dissemination and re-usability

- Outcome of the "Best practices in next-generation sequencing data analysis" workshop held in Cambridge in January 2015

- A number of participants are in the room :-)

Manuscript in print in PLOS Computational Biology, Education section

Training in High-Throughput Sequencing: Common Guidelines to Enable Material Sharing, Dissemination and Re-Usability

Bastian Schiffthaler[1], Myrto Kostadima[2], the NGS Trainer Consortium[3], Nicolas Delhomme[1,4*] and Gabriella Rustici[5*]

[1] Department of Plant Physiology, Umeå Plant Science Centre, Umeå University, SE-901 83 Umeå, Sweden

[2] Department of Haematology, University of Cambridge, Long Road, Cambridge CB2 0PT, United Kingdom

[3] Consortium members' name and affiliation are found in Supporting Information Table S1

[4] Department of Forest Genetics and Plant Physiology, Umeå Plant Science Centre, Swedish University of Agricultural Sciences, SE-901 83 Umeå, Sweden

[5] School of Biological Sciences, Department of Genetics, University of Cambridge, Downing Street, Cambridge CB2 3EH, United Kingdom

* These authors contributed equally

http://bioinformatics.upsc.se/htmr

Acknowledgments

- Eija
- Gabriella Rustici

- Bastian Schiffthaler