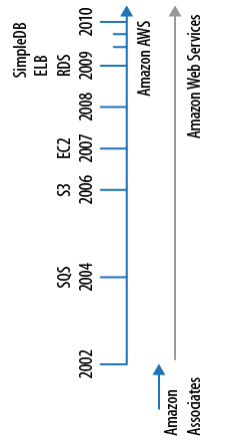
# Amazon Web Services



## Services

- Simple Storage Service (S3)
- CloudFront
- Elastic Compute Cloud (EC2)
- Relational Database Service (RDS)
- SimpleQueueService (SQS)
- SimpleDB
- Elastic MapReduce
- Elastic Beanstack
- EliastiCache
- Email Sending Service (SES)
- Simple Notification Service (SNS)
- DynamoDB
- Virtual Private Cloud (VPC)

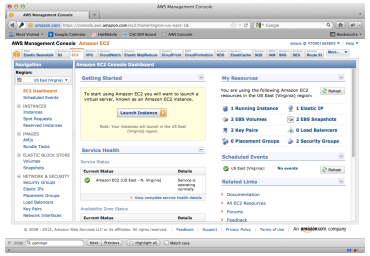



## Regions and Availability Zones

- http://aws.amazon.com/about-aws/globalinfrastructure/#reglink-na
- Region
  - Set of Availability Zones located in one geographic area
  - Currently 8 regions:
    - us-east-1 (Northern Virginia), us-west-1 (Northern California), us-west-2 (Oregon)
    - eu-west-1 (Ireland)
    - ap-southeast-1 (Singapore), ap-northeast-1 (Tokyo), ap-southeast-2 (Sydney)
    - sa-east-1 (Sao Paulo)
- Availability Zone
  - Independent failure domains
    - Distinct location
    - Independent power grid and network connection
  - Zones within a region networked with inexpensive low-latency connections
  - Example: us-east-1a, us-east1b, us-east-1c, us-east-1d

## Interacting with AWS

- Command line tools
  - EC2 http://aws.amazon.com/developertools/351
  - X.509 Certificates
- Web-base Console
  - https://console.aws.amazon.com/
  - Login-password credentials
- AWS SDK
  - Access Keys

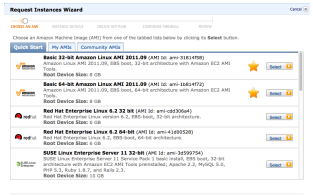


## Elastic Compute Cloud (EC2)

- **Introduced in 2006 (beta), 2007 (full production version)**
- **Based on Xen virtualization**
- **Introduced Infrastructure as a Service (IaaS)**
  - Rent a server on an hourly basis
  - Renting one instance for 10,000 hours is just as cheap as renting 10,000 instances for an hour.

## EC2 Instance

- **Definition: an active virtual server**
- **Created from an Amazon Machine Image (AMI)**
  - Like a boot CD. Root image with everything necessary to start an instance:
- **Large number of images available**
  - Elastic Block Store (EBS)-backed
    - Limited to 1 region
    - Can be suspended and restarted
    - Easy to create new images
  - S3-backed
    - Can be run anywhere
    - Cannot be suspended
    - New image creation is more complicated




## Usage Model

- **Single AMI used to create multiple instances**
  - Customized for application
  - Typically, all applications are pre-loaded
- **Instance are ephemeral**
  - Created when needed
  - Terminated when demand goes away
- **Instance storage (FS) is ephemeral**
  - Goes away when the instance is terminated
- **Persistent storage:**
  - Additional EBS volumes
  - S3, RDS, SimpleDB, DynamoDB

## Instance Types

| Name | Memory | Compute Units | Storage | Platform | I/O Perf | API Name |
|---|---|---|---|---|---|---|
| Micro | 0.6 GB | 2 (only for short bursts) | EBS only | 32/64-bit | Low | t1.micro |
| Standard Small | 1.7 GB | 1 (1 core x 1 unit) | 160 GB | 32-bit | Moderate | m1.small |
| High-CPU Medium | 1.7 GB | 5 (2 cores x 2.5 units) | 350 GB | 32-bit | Moderate | c1.medium |
| High-CPU Extra Large | 7 GB | 20 (8 cores x 2.5 units) | 1690 GB | 64-bit | High | c1.xlarge |
| Standard Large | 7.5 GB | 4 (2 cores x 2 units) | 850 GB | 64-bit | High | m1.large |
| Standard Extra Large | 15 GB | 8 (4 cores x 2 units) | 1690 GB | 64-bit | High | m1.xlarge |
| High-Memory Extra Large | 17.1 GB | 6.5 (2 cores x 3.25 units) | 420 GB | 64-bit | Moderate | m2.xlarge |
| Cluster GPU Quadruple Extra Large | 22 GB | 33.5 (2 x Intel Xeon X5570) | 1690 GB | 64-bit | Very High | cg1.4xlarge |
| Cluster Compute Quadruple Extra Large | 23 GB | 33.5 (2 x Intel Xeon X5570) | 1690 GB | 64-bit | Very High | cc1.4xlarge |
| High-Memory Double Extra Large | 34.2 GB | 13 (4 cores x 3.25 units) | 850 GB | 64-bit | High | m2.2xlarge |
| High-Memory Quadruple Extra Large | 68.4 GB | 26 (8 cores x 3.25 units) | 1690 GB | 64-bit | High | m2.4xlarge |

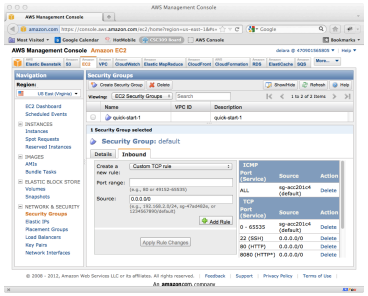
- **One EC2 Compute Unit provides the equivalent CPU capacity of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor.**

## EC2 Pricing

- **Types**
  - On demand
  - Spot instances
  - Pre-paid reserved instances
- **Charges:**
  - Instance type
  - Data transfer

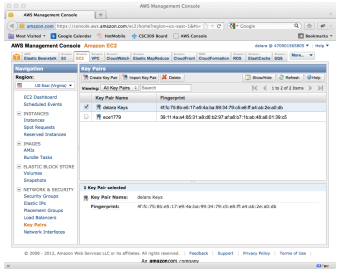
## Security Group

- **Network firewall rules for instance**
- **By default instance is not accessible**
- **Can limit by protocol, port, source IP**



## Key Pairs

- **SSH credentials to log into instance**
- **Install private key (pem file) in .ssh**
- **ssh –i <keypair>.pem  <user>@<instance>**
- **e.g., ssh –i .ssh/ece1779.pem ubuntu@23.21.224.179**



## Elastic IP

- **EC2 instance gets assigned a dynamic  IP**
- **Possible to assign a static IP (Elastic IP)**
- **Elastic IP belongs to AWS account**
- **Can be associated with an instance**
- **Charges:**
  - 1 per instance is free
  - Additional address cost
  - Charge for addresses not associate with an instance (0.005 per hour)
  - IP remaps above 100 per month

## Elastic Block Store (EBS)

- Provides block level storage volumes for EC2 instances.
- Persists independently from the life of an instance
- Database, file system, or access to raw block level storage.
- Can only be attached to 1 instance at time
- Charges:
  - Storage
  - Request

## Create a Custom EC2 Instance

- Select AMI
- Choose instance type
- Choose availability zone
- Set key-pair
- Set security group
- Attach elastic IP
- Attach EBD volumes
- ssh into instance
- Customize
- Create Image

## AWS SDK

- API for AWS services
- Interact with S3, EC2, SimpleDB, etc.
- Available for Java, PHP, Python, Ruby, .NET, Android, iOS

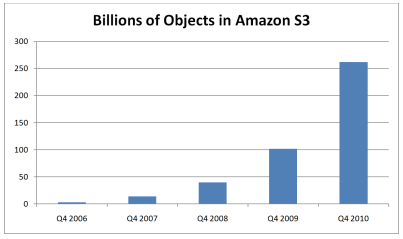
- Java SDK http://aws.amazon.com/sdkforjava/
- Requires AWS Access Keys

```
Access Key Id:     AKIAIPR2ERJTBA6GNBTQ
Secret Access Key: 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
```

- Available though Identity and Access Management (IAM)

## Simple Storage Service (S3)

- Released in 2006
- Scalable fault tolerant data store
- 99.999999999% durability
- 99.99% availability
- Jeff Barr: “If you store 10,000 objects with us, on average we may lose one of them every 10 million years or so.”



## S3 Pricing

- **Charges:**
  - Storage
  - Request
  - Data transfer
- **No charge**
  - Uploads
  - Transfers between S3 and EC2

## S3 (cont.)

- **Data stored in buckets**
- **Bucket name has to be globally unique**
- **Can contain any number of objects**
- **1 bite to 5 TB is size**
- **Every object has a URL**

  [http|https]://<bucketname>.<S3endpoint>/<objectkey>

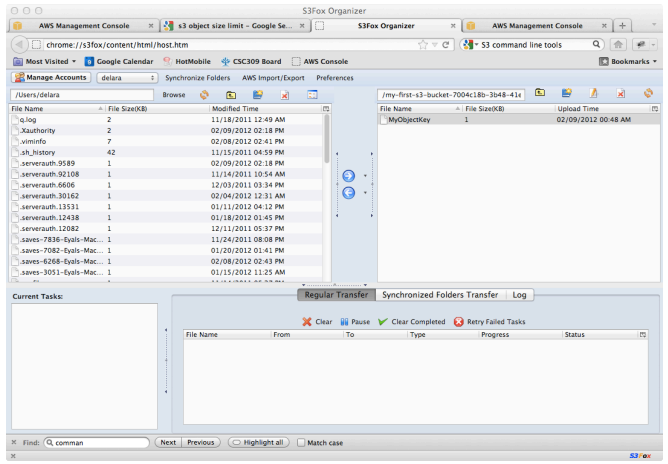
  http://my-first-s3-bucket-7004.s3.amazonaws.com/MyObjectKey

## S3 (cont.)

- **Support access control list (ACL)**
  - Groups of AWS users
  - Any AWS user
  - Anonymous user (the whole world)

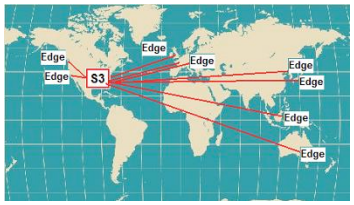
| Permission | When applied to a bucket | When applied to an object |
|---|---|---|
| READ | Permission to list the contents of the bucket | Permission to read the object or its metadata |
| WRITE | Permission to create, replace, or delete any object in the bucket | Unsupported for objects |
| READ_ACP | Permission to read the bucket's ACL | Permission to read the object's ACL |
| WRITE_ACP | Permission to overwrite the bucket's ACL | Permission to overwrite the object's ACL |
| FULL_CONTROL | All of the above | All of the above |

## S3Fox



## Amazon CloudFront

- **Delivers your static and streaming content using a global network of edge locations.**
- **Requests for your objects are automatically routed to the nearest edge location.**
- **Distribution**
  - Specify origin S3 bucket
  - Set a DNS name (optional)
- **Example**
  - S3 https://s3.amazonaws.com/my-first-s3-bucket-4860b609-bc49-430a-b80d-0a239d6f5874/Strawberry.gif
  - CloudFront http://ddhozpxh07c2.cloudfront.net/Strawberry.gif
- **Charges:**
  - Storage
  - Request
  - Data transfer




## CloudFront Edge Network

| United States | Europe | Asia | South America |
|---|---|---|---|
| Ashburn, VA | Amsterdam | Hong Kong | Sao Paulo |
| Dallas/Fort Worth, TX | Dublin | Osaka | |
| Jacksonville, FL | Frankfurt | Singapore | |
| Los Angeles, CA (2) | London | Tokyo | |
| Miami, FL | Milan | | |
| New York, NY (2) | Paris | | |
| Newark, NJ | Stockholm | | |
| Palo Alto, CA | | | |
| San Jose, CA | | | |
| Seattle, WA | | | |
| South Bend, IN | | | |
| St. Louis, MO | | | |

## Relational Database Service (RDS)

- **Easy access to MySQL or Oracle database**
- **Automatic backups**
  - Can roll back database to previous state on 1 second granularity
- **Automatic replication**
- **Automatic patching**
- **Charge**
  - Instance type
  - Storage

## Web Application Architecture