

# AWS 소개

한기용

# 목차

- 클라우드 컴퓨팅 소개
- AWS란?
- EC2 론치 및 Apache Web 서버 실행

클라우드 컴퓨팅이란?

# 클라우드 컴퓨팅의 정의

- 컴퓨팅 자원(하드웨어, 소프트웨어 등등)을 네트워크를 통해 서비스 형태로 사용하는 것.
- 키워드: “No Provisioning”, “Pay As You Go”, “Virtualization”
  - 클라우드상의 자원들을 필요한만큼 (거의) 실시간으로 할당하여 사용한만큼 지불한다.

# 클라우드 컴퓨팅의 장점

- **No Up-Front Investment**

- 데이터센터의 공간을 사거나 하드웨어/소프트웨어등의 직접 살 필요가 없다.

- **No Provisioning Delay**

- 용량증대등으로 인해 서버를 늘려야할 경우 주문,설치등으로 인한 지연이 없다.

- **No Idling Computing Resource**

- 쓰만큼 지불하기 때문에 리소스 관리를 적절히 하면 비용을 최소화할 수 있다.

# 클라우드 컴퓨팅의 영향

- Easy to start an online service start-up
  - 클라우드 컴퓨팅으로 인해 초기 시작비용이 대폭 감소
  - 오픈소스로 인해 개발에 드는 비용도 대폭 감소
- Advance of Virtualization Technology
  - 데이터센터 자체를 Virtualization하려는 노력이 진행중.
    - Rackspace와 NASA 주도의 Open Stack.
    - Facebook의 Open Compute (데이터센터 설계를 오픈소스화)

# 클라우드 컴퓨팅의 단점

- **Strong Tie-up with Provider**

- 사용하는 클라우드 컴퓨팅 업체의 서비스에 최적화되어 다른 환경으로의 이전이 어려울 수 있다.

- **Downtime**

- 해당 업체 서비스에 문제가 있을 경우 같이 영향을 받는다 (2012년 6월 AWS 서비스 문제로 Netflix, Instagram, Pinterest 서비스 다운)

- **Security Concern**

- 데이터들이 외부에 더 노출됨 (데이터전송시 등등)

- **More expensive**

- 많은 업체들이 규모가 커지면 결국 자체 데이터센터를 설립 (Microsoft, Facebook, Google, Yahoo 등등)

**AWS란?**



# AWS

- 가장 큰 클라우드 컴퓨팅 서비스 업체.
- 2002년 아마존의 상품데이터를 **API**로 제공하면서 시작하여 현재 30여개의 서비스를 전세계 10개의 지역에서 제공.
  - 대부분의 서비스들이 오픈소스 프로젝트들을 기반으로 함. ElasticMapReduce와 ElastiCache.
- 사용고객
  - Netflix, Zynga등의 상장업체들도 사용.
  - 실리콘밸리 스타트업의 75% 이상이 AWS를 사용.
  - 일부 국내 업체들도 사용시작.
- 다양한 종류의 소프트웨어/플랫폼 서비스를 제공.
  - AWS의 서비스만으로 쉽게 온라인서비스 생성.
  - 뒤에서 일부 서비스를 따로 설명.

# EC2 – Elastic Compute Cloud (1)

- AWS의 서버 호스팅 서비스.
  - 리눅스 혹은 윈도우 서버를 론치하고 어카운트를 생성하여 로그인 가능 (구글앱엔진과의 가장 큰 차이점).
  - 가상 서버들이라 전용서버에 비해 성능이 떨어짐.
- 다양한 종류의 서버 타입 제공
  - <http://aws.amazon.com/ec2/>
  - 예를 들어 미국 동부에서 스몰타입의 무료 리눅스 서버를 하나 할당시 시간당 8.5센트의 비용지불.
    - 1.7GB 메모리, 1 가상코어, 160GB 하드디스크
  - Incoming network bandwidth는 공짜이지만 outgoing은 유료.

# EC2 – Elastic Compute Cloud (2)

- 세 가지 종류의 구매 옵션

구매 옵션	설명
On-Demand	시간당 비용을 지불되며 가장 흔히 사용하는 옵션
Reserved	1년이나 3년간 사용을 보장하고 1/3 정도 디스카운트를 받는 옵션
Spot Instance	일종의 경매방식으로 놓고 있는 리소스들을 보다 싼 비용으로 사용할 수 있는 옵션

# S3 구성

- <http://aws.amazon.com/s3/>
- S3는 아마존이 제공하는 대용량 클라우드 스토리지 서비스
- S3는 데이터 저장관리를 위해 계층적 구조를 제공
  - 글로벌 네임스페이스를 제공하기 때문에 톱레벨 디렉토리 이름 선정에 주의.
  - S3에서는 디렉토리를 버킷(Bucket)이라고 부름
- 버킷이나 파일별로 액세스 컨트롤 가능

**EC2 론치 및 APACHE WEB 서버 실행**

# 하려는 작업 소개

- EC2 콘솔로 로그인
- 콘솔에서 각자 `t1.micro` 인스턴스의 서버를 하나씩 론치
- SSH 프로그램을 사용하여 그 서버로 로그인
- Apache web server를 실행
- 자신의 랩탑의 웹 브라우저에서 그 웹서버를 접근

# 각자 EC2 console로 로그인

- 조교로부터 받은 Account ID와 비밀번호를 사용하여 주어진 링크를 통해 로그인
- 각자 사용해야하는 지역(region)이 다르므로 꼭 유념할 것
- 먼저 각자 콘솔로 로그인
- 반드시 지역 확인
- Services 탭에서 EC2 선택

- EC2 Dashboard
- Events
- Tags
- Reports
- Limits
- INSTANCES
- Instances
- Spot Requests
- Reserved Instances
- IMAGES
- AMIs
- Bundle Tasks
- ELASTIC BLOCK STORE
- Volumes
- Snapshots
- NETWORK & SECURITY
- Security Groups
- Elastic IPs
- Placement Groups
- Load Balancers

## Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) region:

0 Running Instances	0 Elastic IPs
0 Volumes	0 Snapshots
5 Key Pairs	0 Load Balancers
1 Placement Groups	9 Security Groups

... Easily deploy Ruby, PHP, Java, .NET, Python, Node.js & Docker applications with [Elastic Beanstalk](#).

## 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 East (N. Virginia) region

## Service Health

Service Status:

US East (N. Virginia)

## Scheduled Events

US East (N. Virginia):

No events

### US East (N. Virginia)

- US West (Oregon)
- US West (N. California)
- EU (Ireland)
- EU (Frankfurt)
- Asia Pacific (Singapore)
- Asia Pacific (Tokyo)
- Asia Pacific (Sydney)
- South America (São Paulo)

## AWS Marketplace

Find **free software trial** products in the AWS Marketplace from the [EC2 Launch Wizard](#).

Or try these popular AMIs:

[Vyatta Virtual Router/Firewall/VPN](#)

Provided by Vyatta, Inc.




# EC2 서버 론치하기

- 왼쪽 창에서 **Instances** 링크 선택
- 그리고나서 오른쪽에서 **Launch Instance** 버튼 선택

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'AWS' dropdown, 'Services' dropdown, 'Edit' dropdown, and user information 'Keeyong Han', 'N. Virginia' region, and 'Support' link. On the left sidebar, under the 'INSTANCES' section, the 'Instances' link is highlighted with a red box. In the main content area, the 'Launch Instance' button is highlighted with a red box. Below it, there's a search bar and a message stating 'You do not have any running instances in this region.' with a 'Launch Instance' button. The bottom of the console shows a footer with 'Select an instance above' and some icons.

# 론치 사양 선택 (1)

- 첫번째 OS 선택 – Ubuntu Server 14.04 LTS

 **AWS** ▾ **Services** ▾ **Edit** ▾ Keeyong Han ▾ N. Virginia ▾ Support ▾

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

**SUSE Linux**  
Free tier eligible


SUSE Linux Enterprise Server 12 (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

64-bit

**Ubuntu**  
Free tier eligible


**Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-9aaa1cf6**  
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

**Windows**  
Free tier eligible

**Microsoft Windows Server 2012 R2 Base - ami-c4fe9aac**  
Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English]  
Root device type: ebs    Virtualization type: hvm

**Select**  
64-bit



**Are you launching a database instance? Try Amazon RDS.**  
Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database of your choice (MySQL, PostgreSQL, Oracle, SQL Server) in the cloud. It provides cost-efficient

[Hide](#)

# 론치 사양 선택 (2)

- 두번째 서버 타입 선택 – t2.micro

## Step 2: Choose an Instance Type



T2 instances are VPC-only. Your T2 instance will launch into a VPC and subnet that we create for you. [Learn more](#) about T2 and VPC.

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
<input type="checkbox"/>	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
<input type="checkbox"/>	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High
<input type="checkbox"/>	General purpose	m3.2xlarge	8	30	4 x 80 (SSD)	Yes	High

Cancel

Previous

Review and Launch

Next: Configure Instance Details

# 론치 사양 선택 (3)

- 세번째 서버 타입 선택 – Review and Launch 버튼 클릭

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

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.

Number of instances	<input type="text" value="1"/>
Purchasing option	<input type="checkbox"/> Request Spot Instances
Network	<input type="text" value="vpc-10f30d75 (172.31.0.0/16) (default)"/> <a href="#">Create new VPC</a>
Subnet	<input type="text" value="No preference (default subnet in any Availability Zone)"/> <a href="#">Create new subnet</a>
Auto-assign Public IP	<input type="text" value="Use subnet setting (Enable)"/>
IAM role	<input type="text" value="None"/>
Shutdown behavior	<input type="text" value="Stop"/>
Enable termination protection	<input type="checkbox"/> Protect against accidental termination
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring <a href="#">Additional charges apply.</a>
Tenancy	<input type="text" value="Shared tenancy (multi-tenant hardware)"/> <a href="#">Additional charges will apply for dedicated tenancy.</a>

▶ Advanced Details

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

# 서버 접근을 위한 Key Pair 생성

- “Create a new key pair” 선택하고 이름 지정 후 Download Key Pair 선택 (이름과 위치 기억)

**Select an existing key pair or create a new key pair** [X]

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](#).

Create a new key pair [v]

**Key pair name**  
keeyong

Download Key Pair

You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel Launch Instances

서버가 론치되기를 대기 ...

# 서버가 론치되었으면 ...

- 서버의 public 주소를 알아낸다 (이를 반드시 기억)

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm St
		t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: Public IP:

Description

Status Checks

Monitoring

Tags

Instance ID		Public DNS	-
Instance state	running	Public IP	
Instance type	t2.micro	Elastic IP	-

# SSH로 서버에 로그인

- 리눅스나 맥에서는 아주 간단
  - 터미널 열고 **ssh**를 아래와 같이 실행
  - **ssh -i key파일 ubuntu@자기서버PublicIP**
    - 앞서 론치한 서버의 **IP** 주소와 다운로드받은 키 파일의 위치를 알아야함
- 윈도우에서는 **Putty**라는 프로그램을 다운로드받아야함
  - 거기다가 **AWS**에서 받은 **private key**를 변환해주어야함 ☹



# 로그인 후 아파치 웹서버 설치

- 아파치 웹 서버 설치
  - `sudo apt-get install apache2`
  - Apt-get은 Ubuntu 리눅스의 패키지 매니저
- 웹서버가 보여주는 내용 수정
  - `sudo vi /var/www/html/index.html`
- 설치가 잘 되었으면 자신의 웹 브라우저에서 이를 액세스
  - `http://자기서버PublicIP/`

# Security Group에서 포트 80 열기

The screenshot shows the AWS Management Console interface for editing a Security Group. The main panel displays a table of Security Groups with columns: Name, Group ID, Group Name, VPC ID, and Description. A search bar at the top shows 'search : sg-6732ec02'. Below the table, the 'Edit inbound rules' dialog is open, showing a list of inbound rules. The first rule is 'SSH' (Type) using 'TCP' (Protocol) on port '22' (Port Range) from 'Anywhere' (Source). The second rule is 'Custom TCP Rule' (Type) using 'TCP' (Protocol) on port '80' (Port Range) from 'Anywhere' (Source). The '80' in the Port Range field is highlighted with a blue border. At the bottom of the dialog are buttons for 'Add Rule', 'Cancel', and 'Save'. Below the dialog, the 'Inbound' tab is selected, and an 'Edit' button is visible.

Name	Group ID	Group Name	VPC ID	Description
	sg-6732ec02	launch-wizard-1	vpc-10f30d75	launch-wizar

### Edit inbound rules

Type	Protocol	Port Range	Source
SSH	TCP	22	Anywhere 0.0.0.0/
Custom TCP Rule	TCP	80	Anywhere 0.0.0.0/

Add RuleCancelSave

Security Group: sg-6732ec02

DescriptionInboundOutboundTags

Edit

# 이제 EC2 인스턴스 셧다운

- 마지막으로 생성했던 EC2 인스턴스를 셧다운
  - 안하면 계속 과금 발생 ☺
- 다시 EC2 대시보드로 이동
  - 자신의 인스턴스를 찾아서 선택하고 **Terminate** 메뉴 선택
    - Terminate vs. Stop