

AWS Certified Solutions Architect

課程2 前置作業

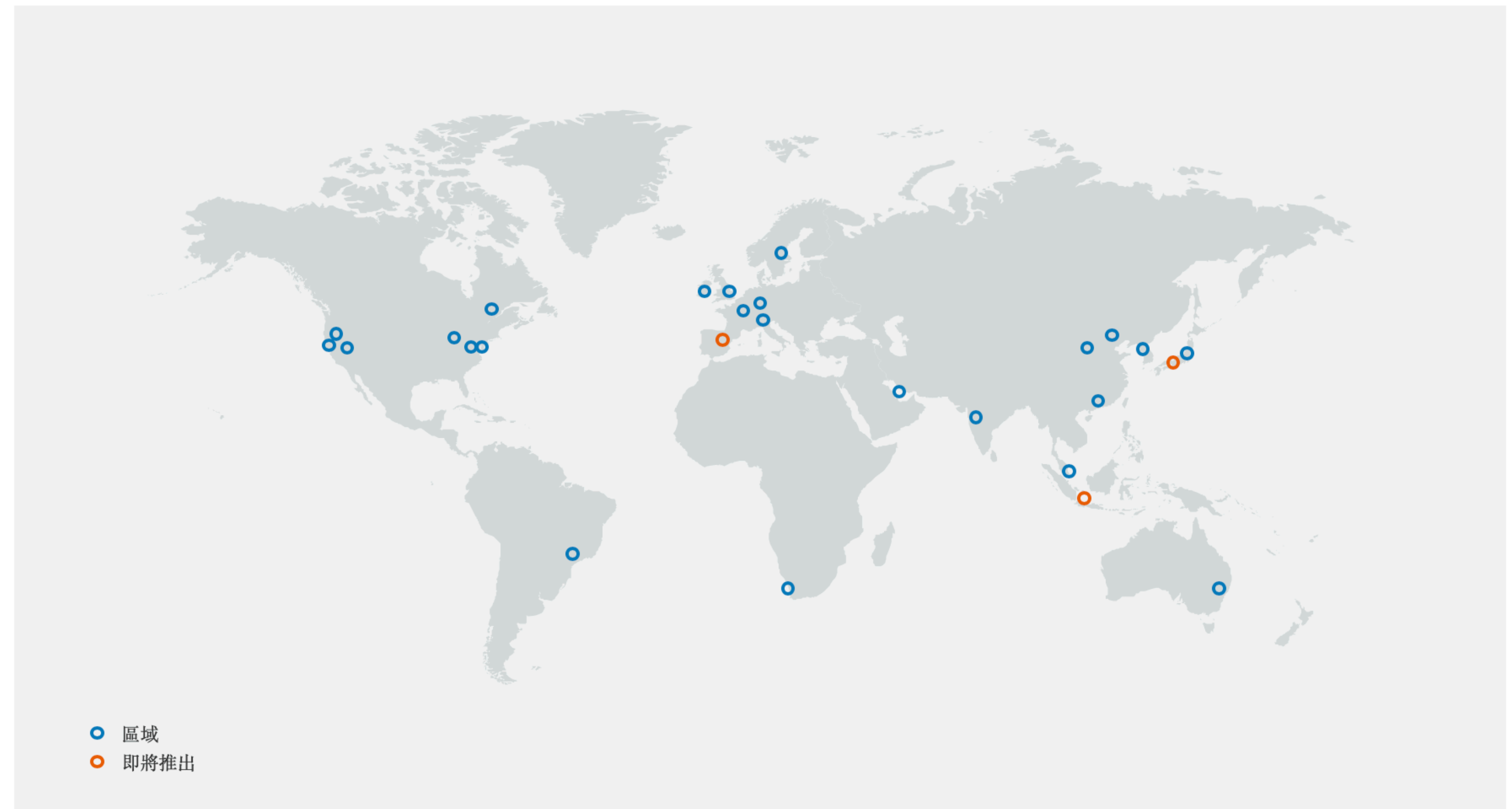
Jacky

AWS基本操作介紹

Global infrastructure

AWS 全球基礎設施地圖

AWS 現在提供遍及全球 24 個地理區域內的 76 個可用區域，且已宣佈計劃在印尼、日本和西班牙增加九個可用區域和三個 AWS 區域。



- 參考來源
- <https://aws.amazon.com/tw/about-aws/global-infrastructure/?p=ngi&loc=1>

AWS 定價

AWS 定價

概觀

免費方案

成本優化

資源



按用量付費

按用量付費可讓您輕鬆適應千變萬化的商業需求，而不用過度投入預算，還能加強您對變化的回應能力。使用按用量付費模式，您能根據需求而不是預測來調整業務，這樣可以降低過度佈建或容量不足的風險。



預訂可以節省費用

對於某些服務，如 Amazon EC2 和 Amazon RDS，您可以投資預留容量。使用預留執行個體時，相較於相等的隨需容量，可節省最多 75% 的費用。當您購買預留執行個體時，預付款越多，折扣越大。



使用越多，付費越少

使用 AWS，您可以獲得數量折扣，並隨著用量增加而實現重要的費用節省。對於如 S3 等服務，則採用分級定價，這表示使用越多，每 GB 的費用越低。AWS 也提供多種取得服務的選項，協助您解決各種業務需求。

參考來源：

<https://aws.amazon.com/tw/pricing/>

除了價格，如何選擇Region？

CloudPing.info

Amazon Web Services™ are available in several regions. Click the button below to estimate the latency from your browser to each AWS™ region.

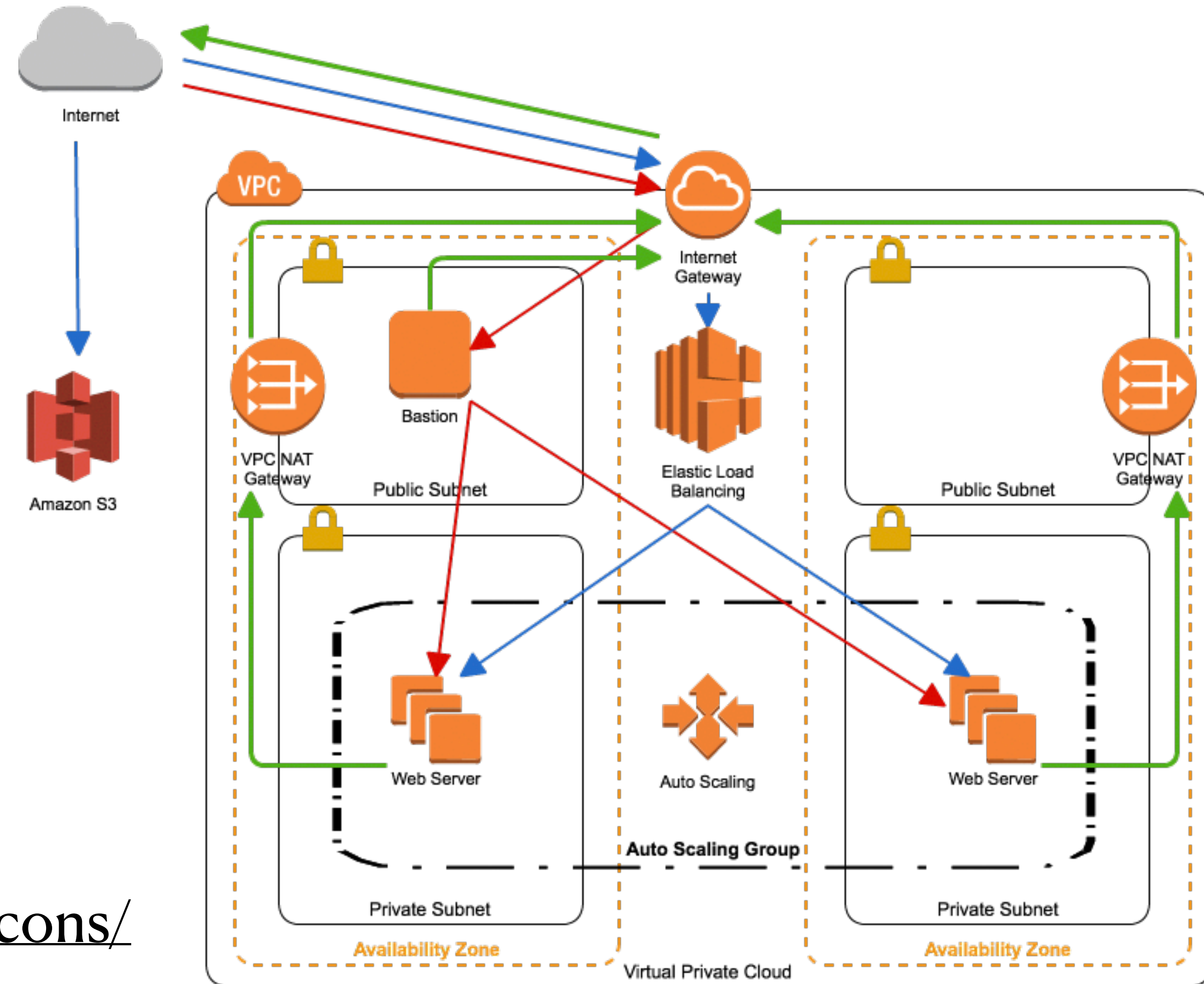
Region	Latency
US-East (Virginia)	596 ms
US East (Ohio)	680 ms
US-West (California)	190 ms
US-West (Oregon)	497 ms
Canada (Central)	640 ms
Europe (Ireland)	358 ms
Europe (London)	611 ms
Europe (Frankfurt)	512 ms
Europe (Paris)	344 ms
Europe (Stockholm)	347 ms
Middle East (Bahrain)	375 ms
Asia Pacific (Hong Kong)	50 ms
Asia Pacific (Mumbai)	548 ms
Asia Pacific (Osaka-Local)	85 ms
Asia Pacific (Seoul)	114 ms
Asia Pacific (Singapore)	104 ms
Asia Pacific (Sydney)	189 ms
Asia Pacific (Tokyo)	101 ms
South America (São Paulo)	632 ms
China (Beijing)	3675 ms
China (Ningxia)	569 ms
AWS GovCloud (US-East)	500 ms
AWS GovCloud (US)	206 ms

HTTP Ping

參考來源：

<https://www.cloudping.info>

AWS 架構圖示



工具參考來源：

<https://aws.amazon.com/tw/architecture/icons/>

操作AWS方法(1)

- Management Console
 - 用帳/密登入
<http://aws.amazon.com>
- CLI
 - 要安裝 AWS CLI Tools
 - 使用Access Key ID/Secret access key 存取

參考來源：

<https://docs.aws.amazon.com/cli/latest/userguide/installing.html>

操作AWS方法(2)

- API
 - Support Restful API / SDK(各語言套件) 方式存取
 - 使用 Access Key ID/Secret access key存取或Role權限存取
 - SDK Support JavaScript(Node.js)、Java、PHP、Ruby、Python、Golang、C#、C++
 - Java建議用SpringCloud、Golang建議用GoCloud套件操作
- CDK/ CloudFormation(cfn)
 - IaC : InfrastructureasCode
 - CDK 實際上也是產生 CFN 去建環境
 - CDK : 支援 TypeScript / python / Java / C# , 未來會支援 golang及 其他更多語言

參考來源：

<https://aws.amazon.com/tw/tools/>

AWS Certified Solutions Architect

課程2 Storge S3

Jacky

Amazon Simple Storage Service (Amazon S3)

- 網際網路儲存服務
- object storage，儲存在上面就是一般的檔案
- 在儲存貯體中存放無限量的資料。您可以視需要將許多物件上傳到 Amazon S3 儲存貯體。每個物件最多可包含 5 TB 的資料。每個物件會透過開發人員指派的唯一金鑰來存放與擷取。
- 預設 可以建立 100 buckets in each of your AWS accounts
- 根目錄:全球唯一的 **Bucket**
- 每一個檔案都是一個 **Object**
- 每一個 Object 會有全球唯一的 URL
- 搭配 CDN 可作為靜態網站存取 (*.html、*.js、*.css、*.jpg...)
- 只能透過 API 存取，不能mount到EC2裡
- select/put/copy/list...
- URL存取: virtual-hosted style
https://**MYBUCKET**.s3.amazonaws.com/**Object**/.....
<https://jackytestbucket.s3-ap-northeast-1.amazonaws.com/index.html>

Amazon S3 資料一致性模式

- PUTS 提供寫入後讀取一致性，但需注意一點。那就是如果您在物件建立之前提出金鑰名稱的 HEAD 或 GET 請求，然後在那之後不久建立物件，則由於最終一致性，後續 GET 可能不會傳回物件。

變更的相關資訊必須跨 Amazon S3 複寫，這可能需要一些時間，因此您可以觀察下列行為：

- 一個處理序會將新物件寫入至 Amazon S3，並立即列出其儲存貯體內的金鑰。在完全傳播變更之前，物件可能不會出現在清單中。
- 一個處理序會取代現有的物件，並立即嘗試讀取該物件。在完全傳播變更之前，Amazon S3 可能會傳回先前的資料。
- 一個處理序會刪除現有的物件，並立即嘗試讀取該物件。在完全傳播刪除之前，Amazon S3 可能會傳回已刪除的資料。
- 一個處理序會刪除現有的物件，並立即列出其儲存貯體內的金鑰。在完全傳播刪除之前，Amazon S3 可能會列出已刪除的物件。

Amazon S3 儲存體方案

- 對於需要高效能的使用案例 (需要毫秒存取時間) 以及經常存取的資料，Amazon S3 提供以下儲存體方案：
 - **S3 標準** — 預設儲存體方案。若在上傳物件時未指定儲存體方案，Amazon S3 會指派 S3 標準型儲存體方案。
 - **低冗餘型** — 低冗餘儲存 (RRS) 儲存體方案是專為非關鍵且可重現的資料所設計，可以低於 S3 標準 儲存體方案的備援層級存放。

RRS 物件的平均年物件遺失率預計為百分之 0.01。如果遺失了 RRS 物件，Amazon S3 會在對該物件提出請求時傳回 405 錯誤。
- 自動最佳化經常存取物件與不常存取物件的儲存體方案(將連續 30 天內沒有存取的物件移動到不常存取層。)
- 不常存取物件的儲存體方案
 - **S3 標準-IA** 和 **S3 單區域 – IA** 儲存體方案是針對存活時間較長且不常存取的資料所設計。(IA 代表不常存取。)

案例介紹

使用 S3 SDK (node js)

參考來源：

<https://aws.amazon.com/tw/sdk-for-node-js/>

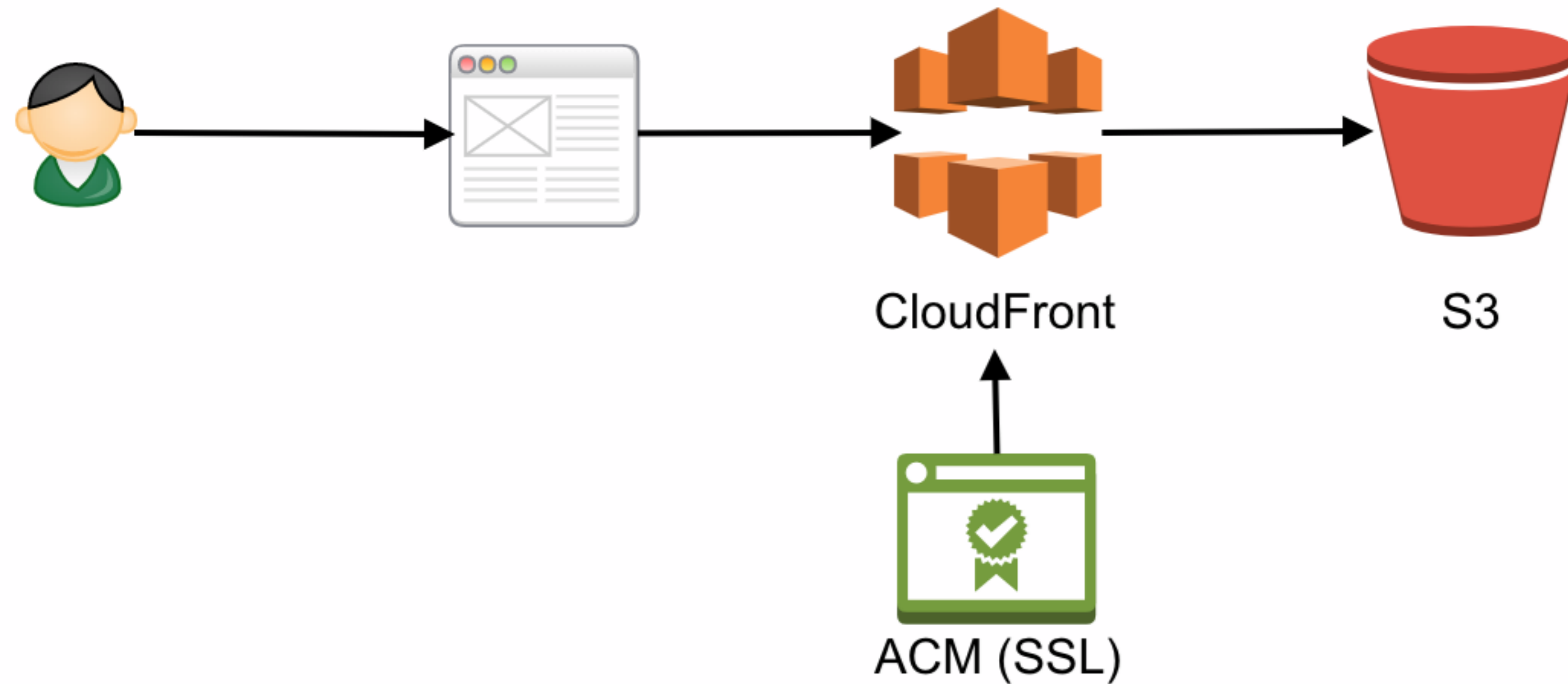
使用 api 存取(Browser-Base Upload to S3)

參考來源：

<https://docs.aws.amazon.com/AmazonS3/latest/API/sgv4-post-example.html>

案例介紹

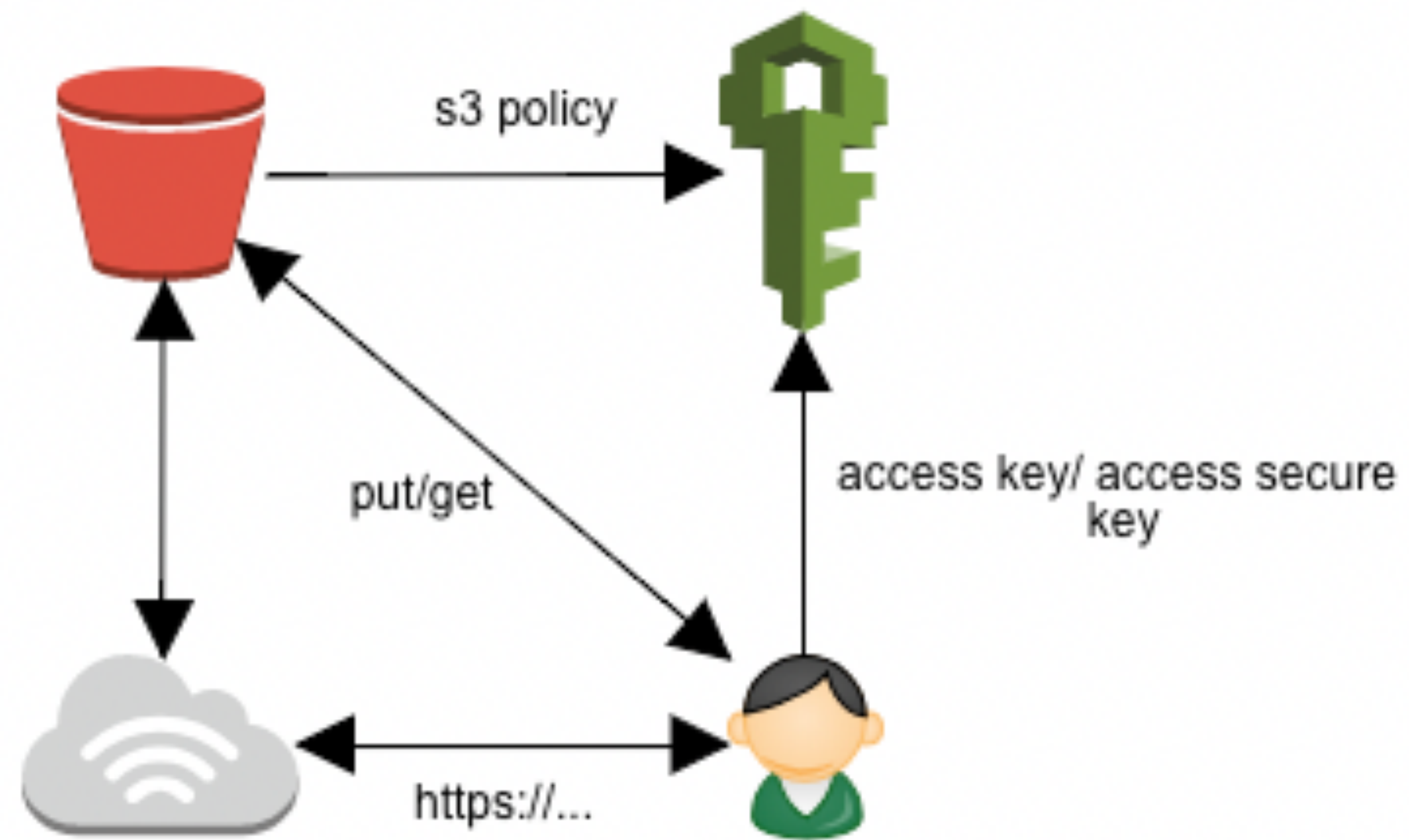
CloudFront + S3 建立 Static Website



參考來源：

https://docs.aws.amazon.com/zh_tw/AmazonS3/latest/user-guide/static-website-hosting.html

Lab—CLI 流程



Lab — CLI 流程

\$ aws configure

Region endpoint 查詢：

https://docs.aws.amazon.com/en_us/general/latest/gr/rande.html

查詢指令

\$ aws s3 help

在 S3Bucket 刪除 s3 檔案

\$ aws s3 rm s3://jackytestbucket/my.log

在 S3Bucket 上傳 s3 檔案

\$ aws s3 cp my.log s3://jackytestbucket

在 S3Bucket 列出 s3 檔案

\$ aws s3 ls s3://jackytestbucket

```
jackey@Kimde-MacBook-Pro > /tmp > aws configure
AWS Access Key ID [None]: AKIA4BC72Q3S5U5V3PFW
AWS Secret Access Key [None]: 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
Default region name [None]: ap-northeast-1
Default output format [None]: json
```

```
✗ jackey@Kimde-MacBook-Pro > /tmp > aws s3 help
jackey@Kimde-MacBook-Pro > /tmp > aws s3 rm s3://jackytestbucket/my.log
delete: s3://jackytestbucket/my.log
jackey@Kimde-MacBook-Pro > /tmp > aws s3 cp my.log s3://jackytestbucket
upload: ./my.log to s3://jackytestbucket/my.log
jackey@Kimde-MacBook-Pro > /tmp > aws s3 ls s3://jackytestbucket
2020-05-20 13:08:34          19 my.log
```

參考來源：

https://docs.aws.amazon.com/zh_tw/cli/latest/userguide/cli-services-s3-commands.html

Lab — 在 cloud9 操作

可以建立一個 EC2 的環境，執行 CLI 的指令，相關指令請參考上一頁

AWS Cloud9

Your environments

Shared with you

Account environments

How-to guide

We do not recommend using your AWS root account to create or work with environments. Use an IAM user instead. This is an AWS security best practice. For more information, see [Setting Up to Use AWS Cloud9](#).

AWS Cloud9 > Your environments

Your environments (1)

Open IDE

View details

Edit

Delete

Create environment

< 1 > ⚙

test

Type	Permissions
EC2	Owner
Description	
No description available	
Owner Arn	
arn:aws:iam::826982172389:root	

Open IDE

Lab — 利用 CloudFormation 操作

- CloudFormation 用 [s3.yaml] 檔案，建立一個 stack，並記住 bucket name
- CloudFormation Stack 的 Output 取得 Access Key 及 Access Secret Key
- 將 Access Key 與 Access Secret Key 設定到 AWS CLI 裡
- 透過 AWS CLI 指令，進行 S3 檔案操作

```
$ aws configure --profile test
```

--profile 參數指定該認證信息的名字，隨便填個多餘的名字就行，一般輸入用戶名。

```
$ echo 'this is a log file' >> my.log
```

```
$ aws s3 cp my.log s3://BUCKET
```

```
$ aws s3 ls s3://BUCKET
```

AWSTemplateFormatVersion: '2010-09-09'

Description: 'User with S3 Bucket'

Parameters:

S3User:
Type: String
Description: 'S3 IAM user name'
Default: ''

S3BucketName:
Type: String
Description: 'S3 bucket name'
Default: ''

Resources:

S3IAMUser:
Type: AWS::IAM::User
Properties:
 UserName: !Sub '\${S3User}'
 Policies:
 - PolicyName: !Sub '\${AWS::StackName}-s3-policy'
 PolicyDocument:
 Version: '2012-10-17'
 Statement:
 - Effect: 'Allow'
 Action:
 - 's3:*'
 Resource:
 - !Sub 'arn:aws:s3:::\${S3Bucket}'
 - !Sub 'arn:aws:s3:::\${S3Bucket}/*'

s3.yaml

S3UserKey:

Type: AWS::IAM::AccessKey

Properties:

Status: Active

UserName: !Ref S3IAMUser

S3Bucket:

Type: AWS::S3::Bucket

Properties:

BucketName: !Sub '\${S3BucketName}'

LifecycleConfiguration:

Rules:

- ExpirationInDays: 365

Status: Enabled

- AbortIncompleteMultipartUpload:

DaysAfterInitiation: 30

Status: Enabled

Outputs:

S3UserAccessKey:

Value: !Ref S3UserKey

Export:

Name: !Sub '\${AWS::StackName}-Access-Key'

S3UserSecretAccessKey:

Value: !GetAtt S3UserKey.SecretAccessKey

Export:

Name: !Sub '\${AWS::StackName}-Secret-Access-Key'

s3.yaml

清理服務資源

將今天建立的相關服務，
若沒有繼續使用，
請記得移除服務。

Homework

1. S3 建立 Static Website
2. 寫一程式將自己的 command history 傳到 S3 儲存

