**AWS區域構成**

1. **資料中心 => 邊緣站點 => 可用區(彼此隔離) => AWS區域**
2. **AWS 節點**

* 位於全世界
* 持續測量互聯網的連接情況，以找到最佳的路由請求方式。
* 使用節點的AWS 服務：Amazon CloudFront、Amazon Route 53、AWS Shield 和AWS WAF（防火牆）

**服務種類**

* **基礎設施即服務(IaaS)**：例如Amazon EC2、Amazon EBS、Amazon VPC
* **平臺即服務(PaaS)**：例如AWS Lambda、Amazon RDS
* **軟體即服務(SaaS)**：例如Amazon Chime（通信服務）

**AWS服務**

**存儲**

* **Amazon S3**
* 最基本
* 類似電腦上的網路附加存儲(NAS)
* **Amazon Elastic Block Store (Amazon EBS)**
* 類似電腦上的直連式存儲(DAS)、硬碟
* 專門與Amazon EC2 配合使用
* 適用於事務密集型工作負載
* **Amazon Elastic File System (Amazon EFS)**
* 類似電腦上的存儲區域網路(SAN)
* 網路檔案系統(NFS)：創建一個文件系統，供EC2掛載
* 在同一VPC內的多台EC2可共用同一個**掛載目標**（即文件系統的IP位址）
* 可按需擴展至PB級

**計算機**

* **Amazon Elastic Compute Cloud (Amazon EC2)**
* 最基本虛擬機
* **Amazon Elastic Container Service (Amazon ECS)**
* 容器編排服務
* **Amazon Elastic Container Registry (Amazon ECR)**
* Docker容器註冊表（可用來檢索映射檔）
* **Amazon Elastic Kubernetes Service (Amazon EKS)**
* 使用Kubernetes 的容器化應用程式
* **AWS Fargate**
* 適用於Amazon ECS 的引擎
* 無需管理伺服器即可運行容器
* **Amazon Lightsail**
* 羽量級雲平臺
* **AWS Batch**
* 可運行數十萬個批次處理工作負載
* **AWS Elastic Beanstalk**
* **平臺即服務**：只需要上傳code，但保留對底層資源的完全控制
* **AWS Lambda**
* **平臺即服務**：只需要上傳code，在適當時機觸發code
* **AWS Outposts**
* 在本地資料中心運行AWS基礎設施
* **VMware Cloud on AWS**
* 欲置混合雲，幫助遷移本機伺服器到AWS平臺

**資料庫**

* **Amazon Relational Database Service (Amazon RDS)**
* 關聯式資料庫
* **Amazon Aurora**
* 關聯式資料庫
* **Amazon Redshift**
* 能夠對PB 級數據分析查詢
* **Amazon DynamoDB**
* 鍵值資料庫

**聯網和內容分發**

* **Amazon Virtual Private Cloud (Amazon VPC)**
* 類似網路管道和交換機
* 在雲中預置一個邏輯上隔離的部分
* **Elastic Load Balancing**
* 類似路由器
* 可在多個目標之間自動分配傳入的流量
* **Amazon CloudFront**
* 內容分發網路(CDN)服務
* 使用AWS邊緣站點高速地向全球客戶分發資料、API
* **AWS Transit Gateway**
* 使客戶能夠將其VPC及本地網路連接到單個閘道，方便集中管理
* **Amazon Route 53**
* DNS服務：將網功能變數名稱稱轉換為IP
* 以便將使用者路由至互聯網應用程式
* **AWS Direct Connect**
* 建立從您的資料中心到AWS 私人網路的連接方法
* **AWS VPN**

**安全性、身份與合規性**

* **AWS Identity and Access Management (IAM)**
* 創建和管理AWS 用戶和組
* **AWS Organizations**
* 組織帳戶
* 可整合帳單
* 創建**服務控制策略(SCP)** ：集中控制多個帳戶

＝＞創建組織單位（OU）＝＞創建帳戶組＝＞將策略附加到某個組

* 類似IAM但SCP不能授予權限，僅能限制
* SCP僅適用於已啟用所有功能的組織
* 用戶只能訪問AWS Organizations 策略和IAM 策略**都允許**的內容
* **Amazon Cognito**
* 添加對用戶註冊、登錄和訪問的控制
* 使用通用身份管理標準，例如安全斷言標記語言(SAML) 2.0
* **AWS Artifact**
* 允許按需訪問或下載AWS 安全性與合規性報告
* 線上接受AWS協議（如業務合作協定(BAA)）並跟蹤其狀態
* **AWS Key Management Service (AWS KMS)**
* 創建和管理金鑰
* 使用硬體安全模組(HSM) 來保護金鑰安全
* 客戶主金鑰(CMK)用於控制對資料加密金鑰（可用來加／解密資料）的訪問
* 可從IAM 控制台中的加密金鑰部分使用KMS
* **AWS Shield**
* 分散式拒絕服務(DDoS) 防護服務：持續檢測和自動內聯緩解功能
* 最大程度地減少應用程式的停機時間和延遲
* AWS Shield Advanced（付費）可進一步保護Amazon EC2、Elastic Load Balancing、Amazon CloudFront、AWS Global Accelerator 和Amazon Route 53上所運行的應用程式。

**成本管理**

* **AWS 成本和使用情況報告**
* **AWS 預算**
* 可設置預算，並在成本即將超過時發出告警
* **AWS Cost Explorer**

**管理和監管**

* **AWS 管理主控台**
* **AWS 命令列介面**
* **AWS Config**
* 可持續監控您的AWS 資源配置
* 是區域性服務
* **Amazon CloudWatch**
* 對資源進行監控
* **AWS Auto Scaling**
* 對資源進行擴展
* **AWS Trusted Advisor**
* 針對整體環境提供五個類別的建議：
* 成本優化
* 安全性
* 性能
* 容錯能力
* 服務限制：基於快照檢查使用量是否超過服務限制的80%
* **AWS Well-Architected Tool**
* 指導您如何改進以達到AWS架構最佳實踐
* **AWS Service Catalog**
* 組織可以創建和管理產品目錄
* **Amazon Macie**
* 可識別敏感性資料，供查看資料的訪問情況
* 在資料洩漏時發出警報
* **Amazon Inspector**
* 會自動評估應用程式的風險、漏洞
* **Amazon GuardDuty**
* 威脅檢測服務
* **AWS CloudTrail**
* 跟蹤使用者活動和API使用情況
* 要保留超過90天的日誌，或發出警報，需創建一個新的跟蹤。

**免費服務**

* 入站資料傳輸
* 同一AWS 區域之間服務資料傳輸
* Amazon VPC
* IAM
* AWS Elastic Beanstalk
* AWS CloudFormation：用於創建一批AWS資源，通過有序且可預測的方式對其進行預置。
* AWS OpsWorks
* 自動擴縮
* 整合帳單

**安全性、身份與合規性**

1. **IAM**

**基本組件**

* **IAM用戶**
* **IAM組(Group)**：IAM用戶的集合
* 一個用戶可屬於多個組
* 組不能包含其他組
* **IAM策略(Policy)**：定義許可權的文檔（JSON格式）
* **IAM角色(Role)**：被授予臨時存取權限或特定策略的一個身分，由需要的人代入
* 在創建時，需要選擇該角色會用到的AWS服務和策略
* 在使用服務時即可在IAM role選取這個角色
* 當代入角色時，會提供角色的臨時安全憑證

**訪問類型**

* **編程訪問權限**
* 使用戶有訪問AWS CLI 之許可權
* 需提供訪問金鑰ID和秘密訪問密鑰
* **AWS管理主控台訪問**
* 使用戶有訪問AWS管理主控台之許可權
* 需提供帳戶ID或帳戶別名、IAM 用戶名和密碼

**授權**

* IAM用戶默認情況下無權訪問任何資源（隱式拒絕），需透過創建策略予以授權
* IAM 服務配置的範圍為所有AWS 區域

**策略**

* 顯式拒絕語句的優先順序高於顯式允許語句
* 基於身份的策略
* 託管策略：創建一個策略文檔，再附加到多個用戶、組和角色
* 內聯策略：針對單個用戶、組或角色創建文檔。
* 基於資源的策略
* 僅內聯方式
* 創建方式：點擊該資源＝＞Permissions＝＞XXX Policy
* [IAM 策略模擬器](https://policysim.aws.amazon.com/)：可用來模擬策略有無成功

1. **AWS 帳戶根用戶**

* **首次創建AWS 帳戶時的身分，**具有完全存取權限　**＝＞危險！**
* 建議使用IAM 來創建其他用戶，並分配最低所需許可權（最小特權原則）
* IAM儀表板有**安全狀態審查機制**，供使用者一步一步審查，包括：

1. **停止使用帳戶根使用者：**
2. 禁用並刪除您的帳戶根用戶訪問金鑰（如果存在）
3. 將您的帳戶根使用者憑證存儲在安全的位置
4. **在帳戶根用戶上啟動MFA**

* MFA：通過由虛擬設備（EX：Google身份驗證器、Authy 雙重身份驗證）、U2F安全金鑰設備生成的代碼，或硬體設備提供的金鑰卡或顯卡進行身分驗證
* 步驟

1. Manage MFA ＝＞ Assign MFA device：選擇VirtualMFA device
2. 需下載一個應用程式（如Google 身份驗證器）
3. 掃描QR code並完成驗證程式
4. **創建IAM帳戶並使用IAM組分配權限**
5. **為所有使用者規範並啟用密碼策略**

* Account Settings => Password Policy

1. **以IAM 用戶身分登錄：**從IAM 儀表板找到IAM使用者登錄連結，瀏覽並使用帳密登錄
2. **資料安全機制**

**加密**

* 靜態數據：創建加密檔案系統 ＝＞ 使用AWS KMS解密
* 網路傳輸數據：透過AWS Certificate Manager ＝＞ 請求SSL或TLS證書（https） ＝＞ 部署在AWS 資源（如負載等化器或CloudFront）上

**限制資源存取權的方法**

* 阻止公有訪問：會覆蓋任何其他策略
* 編寫IAM策略
* 存取控制清單(ACL)：不常用

**聯網和內容分發**

1. **網路概念**

**IP**

* IPv4：32 位
* IPv6：128 位元元元元元（4個字母或數字x8組，用(:)隔開）

**無類別域間路由(CIDR)**

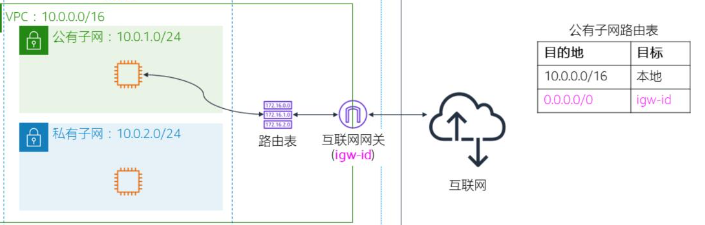
* 訂立一個**IP區間**的表達方式
* 標記法：**IP / IP前幾位必須固定**
* 192.0.0.2/24：表明範圍從192.0.0.0到192.0.0.256（紅色表示要固定的24位）
* /32：每一位都固定，在設置防火牆規則時要這麼訂立
* 0.0.0.0/0：每一位都靈活的互聯網

1. **Amazon VPC**

* 可選擇自己的**IP範圍、創建子網、配置路由表和網路閘道**
* **VPC**：屬於單個AWS區域，並可劃分成一個或多個可用區（子網）
* **子網**：屬於單個可用區，即VPC內的CIDR塊（IP範圍）
* IPv4 CIDR 塊最大為/16，最小為/28（即16個IP）
* 創建後無法更改位址範圍
* **預留IP地址**：對於每個CIDR 塊，AWS 都會預留五個IP不提供使用，這些IP用於

1. 網路位址
2. VPC本地路由器（內部通信）
3. DNS解析
4. 未來使用
5. 網路廣播位址

* 公有子網可聯網，但私有子網不能
* **VPC 中的實例**
* 會自動獲得一個私有IP
* 在預設VPC中啟動時，會自動被分配一個公有IP
* 可修改子網的**自動分配公有IP位址**屬性，以在自創的VPC創建實例時自動分配公有IP地址
* **彈性IP 位址**
* 專用於動態雲計算的靜態公有IPv4 地址
* 將彈性IP位址與VPC中的實例相關聯 ＝＞ 當某台實例故障時，IP可以迅速映射到其他實例繼續作業
* 已停止的實例在重新開機時會接收新的公有IP，若需要一個永久的公有IP：可在實例所在區域分配一個新的彈性IP ＝＞ 將彈性IP 位址與EC2關聯
* 一個帳戶在每個區域最多可擁有五個彈性IP
* 可能會產生額外費用
* **彈性網路介面**
* **主網路介面：**每個實例的預設網路介面，有一個私有IPv4 地址，無法與實例分離
* 可額外將彈性網路介面與VPC中的實例相關聯 ＝＞ 當網路介面重新連接到另一個實例時，它的屬性也會隨之附加
* **路由表**
* 路由表包含一組規則（路由），用於定向子網網路流量的
* **目的地**：CIDR 塊
* **目標**：例如本地
* 預設每個路由表都包含本地路由
* 每個子網都必須與一個路由表相關聯，且只能一個
* 多個子網可與同一個路由表關聯
* 主路由表控制未與任何其他路由表顯式關聯的所有子網
* **互聯網閘道(網關)**
* 可使公有子網中的實例（需有公有IP）和互聯網通信
* 作法
* 將一個互聯網閘道連接到您的VPC
* 添加路由
* 公有子網路由表：目的地－0.0.0.0/0 **目標**－互聯網閘道



* **網路位址轉譯(NAT)閘道**
* 可使私有子網中的實例連接到互聯網
* 作法
* 指定NAT閘道要放在哪個公有子網
* 指定彈性IP 位址（私有子網）
* 更新路由
* 公有子網路由表：目的地－0.0.0.0/0 目標－互聯網閘道
* 私有子網路由表：目的地－0.0.0.0/0 **目標**－NAT閘道
* 比起在公有子網使用NAT實例，NAT閘道有更高的可用性
* **VPC 共用**
* 與AWS Organizations 內的其他帳戶共用VPC
* **VPC 對等連接**
* 兩個VPC之間的網路連接，使其中的實例可以彼此通信
* 配置路由表
* VPC-1路由表：目的地－VPC-2 **目標**－對等連接ID
* VPC-2路由表：目的地－VPC-1 **目標**－對等連接ID
* **AWS 網站到網站VPN**
* 使私有子網內實體連線到遠端私網
* 作法
* 創建一個**虛擬私人網絡(VPN)閘道**並將其連接到您的VPC
* 定義另一端客戶閘道的配置
* 創建一個客戶端的路由表
* 更新安全性群組規則
* 建立一個**AWS 網站到網站VPN**
* 配置路由
* 公有子網路由表：目的地－0.0.0.0/0 目標－互聯網閘道
* 私有子網路由表：目的地－遠端私網 **目標**－虛擬私人網絡(VPN)閘道
* **AWS Direct Connect（DX）**
* 類似AWS 網站到網站VPN，不通過互聯網，以獲得優於互聯網穩定的連網體驗
* 使用虛擬區域網路(VLAN)
* **VPC 終端節點**
* 連接到AWS服務
* 路由配置
* 公有子網路由表：目的地－服務 **目標**－VPC 終端節點
* **介面終端節點**：可連接到由AWS 合作夥伴網路(APN)提供的服務，需付費
* **閘道終端節點**：不收費
* **AWS Transit Gateway**
* 將所有對等連接的VPC及本地網路連接到單個閘道，方便集中管理
* 每個新網路只需連接到Transit Gateway

1. **安全性**

* **安全性群組**
* 實例的防火牆
* 有狀態
* 預設沒有入站規則，但允許所有出站規則
* **自訂安全性群組：**可以“允許”規則，但不可以“拒絕”規則
* **網路存取控制清單（網路ACL）**
* 子網的防火牆
* 無狀態
* 每個子網都必須與一個（且只能一個）網路ACL關聯
* 預設與**預設網路ACL** 關聯
* **預設網路ACL**允許所有入站和出站流量
* **自定義網路ACL**：默認拒絕所有入站和出站流量。
* 可以允許或拒絕規則
* 依編號從數字最小的規則開始評估

1. **Amazon Route 53**

* DNS服務：把客戶的網功能變數名稱稱轉換為IP，從而將客戶路由到我們的互聯網應用程式
* 提供檢查資源運行狀況的功能，以便路由至可用的資源
* 也提供功能變數（網域）名稱註冊功能
* 多種類型的路由策略
* **簡單路由（輪詢）**
* **加權輪詢路由**
* 依權重將流量路由到多個資源，以便提供不同種的回應
* 可用來進行A/B 測試
* **延遲路由(LBR)**
* 將流量路由到提供最佳延遲（最快）的區域
* 原理：將客戶路由到AWS 終端節點，以便測量各應用程式性能
* 可實現多區域部屬
* **地理位置路由**
* 根據使用者的位置來路由
* 或是在終端節點間均衡負載，使每個使用者位置一致地路由到同一終端節點
* **地理位置臨近度路由**
* 根據資源的位置來路由
* 以及將流量從一個位置中的資源轉移到另一個位置
* **容錯移轉路由**
* 監控網站中斷情形，並將使用者重定向到備用位置
* **多值應答路由**
* 可配置返回多個值來回應查詢
* 允許您檢查每個資源的運行狀況，以便只返回正常資源的值

1. **Amazon CloudFront**

* 一種**內容分發網路(CDN)服務**：緩存在源服務器上的請求檔（HTML,CSS…）副本，從緩存邊緣直接分發副本至本地，從而以最快的速度向請求者分發內容
* 也提供動態內容，但唯一且不可緩存
* **全球資料中心（邊緣網站）網路**
* 提供熱門內容
* **區域性邊緣緩存**
* 更靠近請求者
* 提供熱度下降的內容
* 比單個邊緣網站的緩存空間更大，保留時間也更長
* 可程式設計
* 可與Lambda@Edge 集成，將複雜的應用程式移近使用者
* 提供持續集成和持續交付(CI/CD) 環境
* 收費
* 數據傳出
* HTTP(S) 請求
* 失效請求：每月可免費請求將1000 個URL從緩存中移除（失效）
* 專用IP 自訂安全通訊端層(SSL)

**計算服務**

1. **Amazon EC2**

* **創建步驟**

1. **選擇AMI(Amazon系統映射檔)**

AMI包括以下元件

* 根卷：完整的作業系統(OS)
* 塊儲存設備映射：附加的卷

1. **選擇實例類型**
2. **T3（T 系列第三代）**

* 最通用
* CPU性能可突增

1. **C5**

* 針對計算密集型工作負載
* 使用案例：科學建模、批次處理、廣告投放、多人遊戲和視頻編碼

1. **R5**

* 針對記憶體密集型應用程式
* 使用案例：高性能資料庫、Apache Hadoop/Spark 集群

1. **指定網路設置**

* 確認區域是否正確、是否有公有IP
* 啟動多個新的實例時，EC2會嘗試被分散放置在不同區域，減少故障 ＝＞ 可使用**置放群組**將一組實例放置在一起

1. **附加IAM角色(可選)**

* 創建角色也就是創建一個實例設定檔，與角色同名

1. **用戶數據腳本(可選)**

* 在首次啟動時運行的腳本（安裝wget, httpd…）
* 若想在每次啟動都運行，可以創建**MIME分段檔使用者資料腳本**

1. **指定存儲（硬碟）**

* 可以指定目標根卷磁片大小
* 也可附加其他存儲卷
* 一些存儲選項
* **Amazon EBS**
* **Amazon EC2 實例存儲**
* 實例本身的臨時資料塊級存儲
* 只要實例終止，存儲的資料會全部遺失（重新啟動不會）
* 無法透過API輕易使實例終止

1. **添加標籤**

* 為資源自訂分類或標記
* 包含鍵和值
* 取小寫會消失在實例主清單中

1. **安全組設置**
2. **指定或創建密鑰對**

* 公鑰私鑰為密鑰對：公鑰加密，私鑰解密
* **休眠**
* 只有由Amazon EBS支援存儲的某些Linux AMI 支援
* 需要加密EBS根卷才能休眠
* 必須在首次啟動實例時啟用休眠
* **實例中繼資料**
* 有關實例的資料：包括公／私有IP、公有主機名稱、實例ID、安全性群組、區域和可用區等
* 訪問URL：http://169.254.169.254/latest/meta-data/
* 使用者資料：http://169.254.169.254/latest/user-data
* **計價方式**
* **專用實例**：有自己的專用硬體和實例
* **專用主機**：專用的物理伺服器，適用於您具有現有的按通訊端、按核心或按虛擬機器的許可，並且必須滿足特定的公司合規性和法規要求的情況
* **預留實例**：預留1年或3年
* **計畫預留實例**：適用於每日／週／月會用到一樣的容量重複執行計畫時
* **Spot 實例**：依每小時市場波動價格競價買實例

1. **Amazon ECS**

* 部屬並運行容器的管理工具
* **任務定義**：類似Docker document──對構成一個應用程式的所有容器（最多十個）的描述
* **任務**：任務定義完後產生的實體容器
* **任務計畫程式**：將任務放置到集群（實例或伺服器）的策略
* **集群**：運行容器的實例或伺服器
* **EC2 Linux／Windows + 聯網集群：**需自行創建、控管EC2
* **僅聯網集群（由AWS Fargate支援）：**一切託管

1. **Amazon EKS**

* Kubernetes 也是用於管理容器的開源軟體
* **節點：**實例集群
* **Pod**
* 多個容器組成的單位，方便控管
* 每個Pod 都有一個IP 和DNS

1. **Amazon ECR**
   * Docker容器註冊表：存放映射檔
   * 可在Amazon ECS任務定義中指定Amazon ECR 存儲庫
2. **AWS Lambda** 
   * 使您無需管理伺服器即可運行code
   * 創建**Lambda 函數（code）** ＝＞ 設置**觸發方式（事件源）**
   * 函數的設定最終會存為一個**部署套裝程式（ZIP檔）**

* **層**：有關資料庫設置的相關內容
* 利用層，則可以在函數中直接指定使用哪個資料庫，而不必在部署套裝程式中創建資料庫

**存儲服務**

1. **存儲解決方案**
   * **物件存儲**：修改時必須更新整個檔
   * **資料塊存儲**：修改時僅更改小部分內容資料塊 ＝＞ 速度快，但較貴
2. **Amazon EBS** 
   * 最常附加於EC2實例的資料塊級存儲（**＝硬碟**）
   * Amazon EBS卷的備份稱為**快照**

* 備份到Amazon S3
* 第一個快照：**基準快照**
* 第二個之後的快照：僅捕獲與之前不同的資料
* 可創建卷的**時間點快照**，並從快照復原卷
  + 只有SSD 可用於EC2 實例的開機磁碟區
* 預置IOPS型SSD：提供最高性能
* 通用型SSD：最常用
  + 加密傳輸Amazon EBS 卷無需額外付費
  + 成本
* 卷存儲量
* 快照存儲
* IOPS

1. **Amazon S3** 
   * 物件級存儲
   * 存儲桶具有URL（以**存儲桶名稱**作為路徑，因此必須在全球具有唯一性）
   * 冗餘存儲：存儲桶內的資料會自動在至少三個不同可用區被備份
   * **存儲類分析功能(Analytics)**：可自動分析存儲被訪問的頻率，根據**生命週期策略**轉移到適當的存儲類別，以降低成本
   * 存儲類別包括：

* **Amazon S3 標準**
* 低延遲、高輸送量
* 單個檔案容量上限為5TB
* 對不同請求類型（GET／PUT／COPY）收費
* **Amazon S3 標準–IA（不頻繁訪問）**
* 在Amazon S3 標準中30天未被訪問的資料，會轉往這
* 低延遲、高輸送量但可用性較低（檢索較慢）
* 成本較低
* 適合：災難恢復檔的資料存儲
* **Amazon S3 單區–IA（不頻繁訪問）**
* 將資料存儲在單個可用區中（沒有冗餘存儲）
* 適合：可輕鬆重新創建的資料
* **Amazon S3 Glacier**
* 在Amazon S3 –IA 30天未被訪問的資料，會轉往這保留1年之後刪除
* 單個檔案容量上限為40TB ＝＞ 容量更大，更適合積存
* 成本極低（但檢索成本較高）
* 提供三種檢索時間選項：加速（最快）、標準、批量（最慢）

＝＞ 收費不同

* 請求類型：UPLOAD
* 資料預設對伺服器端（https）都會加密
* **Amazon S3 Glacier Deep Archive**
* 對於不頻繁訪問的資料能保留更久，成本最低
* **Amazon S3 智能分層**
* 自動監控資料，並移至最經濟高效的訪問層
  + **伺服器端加密(HTTPS)方法**（可選）

1. 使用**託管的加密金鑰(SSE-S3)**

* 用唯一的金鑰來加密每個物件，用定期輪換的主金鑰來加密金鑰
* 使用256位高級加密標準(AES-256)

1. 使用**客戶自己設置的加密金鑰(SSE-C)**
2. 使用**AWS KMS**

**資料庫服務**

1. **Amazon RDS**
   * 支援的資料庫

* MySQL
* Amazon Aurora
* Microsoft SQL Server
* PostgreSQL
* MariaDB
* Oracle
  + 若VPC中有多個可用區，會自動在其他可用區生成資料庫實例的副本
  + 可在不同區域創建唯讀副本實例
  + 適用於
* 中高查詢或寫入速率（約30000 IOPS）
* 不超過一個分區

1. **Amazon DynamoDB**
   * NoSQL服務
   * 存儲在固態硬碟(SSD)上
   * 備份

* 冗餘存儲：在不同可用區
* **全域表**功能：跨區域自動複製表
  + 可為表預置傳輸量
  + 核心組件
* 主鍵：**分區鍵、排序鍵**
* **屬性**：資料其他部分
* **表**：表格
* **專案**：形容篩選查詢過後的資料群體
  + 檢索資料方法
* 使用主鍵
* 掃描：關鍵字檢索，效率較低

1. **Amazon Redshift**
   * 針對PB 級的**結構化資料**快速進行分析
   * 使用列式存儲
   * 高相容性：使用Java資料庫連接(JDBC)和開放式資料庫連接(ODBC)連接器 ＝＞ 自由連接各種SQL和BI分析工具
   * 並行處理架構 ＝＞ 超高性能

* **領導節點**：管理所有通信，編譯執行計劃
* **計算節點**：執行領導節點分配的計畫代碼
  + Amazon Redshift Spectrum：直接在Amazon S3 中運行查詢

1. **Amazon Aurora**
   * 專為雲構建的關聯式資料庫
   * 由Amazon RDS 管理
   * 冗餘存儲
   * 可與AWS Database Migration Service (AWS DMS) 和AWS Schema Conversion Tool集成 ＝＞ 幫助將資料集遷移到Amazon Aurora
   * 提供快速的分散式存儲子系統
   * 即時崩潰恢復

* 在每個讀取操作上都重放資料庫檢查點日誌 ＝＞ 縮短重啟時間
* 將緩衝區緩存移出 ＝＞ 重啟時立即可用

**雲架構**

1. **AWS按照五大支柱架構完善框架**
   * **成本優化支柱**
   * **安全性支柱**
   * **卓越運營支柱**
   * **可靠性支助**

* 處理變更或故障修復
* 橫向聚合系統以降低單點故障的影響
* **可靠性指標：**平均無故障時間(MTBF) = 平均故障時間(MTTF)+平均修復時間(MTTR)
  + **性能效率支柱**
* 提供技術服務、多種類型的資源供選擇
* 無伺服器架構、數分鐘內實現全球化部署

**自動擴展和監控**

1. **Elastic Load Balancing**
   * 可以將傳入的網路流量分配給多個目標
   * 通過偵聽器檢查連接請求，配置用戶端－負載等化器－目標之間的協定和埠號
   * 在VPC中可以指定負載等化器是公開（預設）還是面向內部（私網）
   * 混合負載平衡

* 在AWS和本地資源之間進行負載均衡
* 可使用同一負載等化器：AWS和本地設為兩個目標
* 或使用兩個負載等化器
  + 類型
* **Application Load Balancer（應用程式負載等化器）**
* 在應用程式級別（OSI模型第7層）運行
* 適合HTTP流量
* 使用SSL/TLS密碼和協定
* 支援在Lambda函數或容器化應用程式的多個埠之間進行負載均衡
* 可創建目標群組並加入目標
* **Network Load Balancer（網路負載等化器）**
* 在網路傳輸級別（OSI 模型第4層）運行
* 適合TCP和UDP流量
* 每秒可處理數百萬次請求
* 可創建目標群組並加入目標
* **Classic Load Balancer（較舊的方案）**
* HTTP、TCP、SSL都通用
* 僅可加入目標實例

1. **Amazon EC2 Auto Scaling**
   * **Auto Scaling 組**：自訂所需容量的實例數量
   * 擴展選項

* **手動擴展**：依據自訂的所需最大、最小、所需容量
* **計畫擴展**：依據自訂時間
* **動態按需擴展**：依據CloudWatch警報擴展
* **預測性擴展**：與AWS Auto Scaling配合，使用機器學習來預測流量