aws

擴展 Amazon EKS 基礎設施以最佳化運算、工作負載和網路效能

AWS 方案指引

AWS 方案指引: 擴展 Amazon EKS 基礎設施以最佳化運算、工作負載和網路效能

Table of Contents

簡介 1
目標 2
運算擴展 3
叢集 AutoScaler 3
具有過度佈建的 Cluster Autoscaler 3
Karpenter 4
工作負載擴展 5
Horizontal Pod Autoscaler 5
叢集比例自動擴展器 6
Kubernetes 型事件驅動自動擴展器 6
網路擴展 8
Kubernetes 專用 Amazon VPC CNI 外掛程式 8
自訂聯網 9
字首委派 9
Amazon VPC Lattice 10
成本最佳化 11
Kubecost 11
金絲雀 12
AWS Fargate 12
Spot 執行個體 13
預留執行個體 13
AWS Graviton 執行個體 14
後續步驟 15
資源 16
文件歷史紀錄 17
詞彙表 18
..... 18
A 18
B 21
C 23
D 25
E 29
F 30
G 32

H 33
I 34
L 36
M 37
O 40
P 43
Q 45
R 45
S 48
T 51
U 52
V 52
W 53
Z 54
..... Iv

擴展 Amazon EKS 基礎設施以最佳化運算、工作負載和網路效能

Aniket Dekate、Aniket Kurzadkar 和 Ishwar Chauthaiwale，Amazon Web Services (AWS)

2024 年 11 月 (文件歷史記錄)

Amazon Elastic Kubernetes Service (Amazon EKS) 是一種受管 Kubernetes 服務。使用 Amazon EKS，您可以在容器化雲端環境中執行 Kubernetes Pod，而不需要安裝和操作您自己的控制平面。透過 AWS 管理控制平面，Amazon EKS 可減少組織營運管理。使用 Amazon EKS 的其他優點包括雲端環境中的擴展、可靠性和安全性。

本指南旨在協助組織在下列領域最佳化其 Amazon EKS 基礎設施：

- 運算擴展是動態 Kubernetes 環境中應用程式效能的關鍵元件：
  - 高效的資源配置 – 了解動態配置計算資源的技術，以滿足各種需求。
  - 自動化工具 – 取得工具和服務的概觀，以自動化運算擴展，減少手動介入的需求。
- 工作負載擴展有助於確保應用程式可以處理各種工作負載，而不會降低效能：
  - 水平 Pod 自動擴展器 – 深入了解 HPA 如何協助根據即時指標擴展工作負載。
  - 叢集比例自動擴展器 – 了解 CPA 如何自動擴展和維護節點和複本之間的比例關係，隨著叢集大小的變化向上或向下擴展工作負載。
  - 事件驅動擴展 – 檢閱擴展應用程式以回應特定事件或觸發的策略。
- 網路擴展有助於在動態環境中維持服務與高效率資料流程之間的無縫通訊：
  - Amazon VPC CNI 外掛程式 – 了解 VPC CNI 外掛程式如何在 Amazon EKS 叢集中啟用可擴展的網路。
  - 自訂聯網 - 檢閱 Amazon EKS 叢集上的 IP 地址管理和網路流量隔離。
  - 字首委派 - 取得在大型且可擴展的 Amazon EKS 叢集中簡化 IP 管理的概觀。
  - Amazon VPC Lattice – 取得 VPC Lattice 如何管理跨 VPC service-to-service聯網以實現無縫擴展的概觀。
- 成本最佳化可協助企業查看其資源的花費，並將費用適當地指派給部門或專案：
  - 適當調整資源大小 – 考慮適當調整工作負載雲端資源大小的技巧。
  - 成本監控和控制 – 檢閱追蹤和最佳化雲端費用的工具和最佳實務。

每個區段都著重於建立可靠、有效且經濟實惠雲端環境所需的特定目標。

目標

本指南可協助您和組織達成下列業務目標：

- 增強的資源效率 – 根據即時需求動態擴展運算、工作負載和網路資源，以達到最佳的資源使用率。

  此目標強調了向上和向下擴展資源以回應實際使用模式的重要性。水平 Pod 自動擴展器和 Amazon VPC CNI 外掛程式等工具可協助組織僅使用所需的資源、將浪費降至最低，並最大化效能。

- 改善應用程式效能 – 即使在工作負載和流量模式波動的情況下，也能維持應用程式的高效能和回應能力。

  此目標著重於策略，以協助確保應用程式可以處理尖峰流量和繁重工作負載，而不會犧牲效能。事件驅動型工作負載擴展、高效率運算配置和可擴展性網路架構等技術，是實現此目標的關鍵。

- 無縫可擴展性 – 可順暢擴展基礎設施元件，輕鬆成長和適應不斷變化的業務需求。

  無縫可擴展性對於預期成長或遇到不同流量層級的組織至關重要。此目標說明在運算、工作負載和網路資源之間實作可擴展解決方案的重要性，因此擴展可以自動、高效且透明。

- 成本最佳化 – 將雲端成本降至最低，同時維持或改善效能和可擴展性。

  成本最佳化可以包含減少費用，例如適當調整資源、使用經濟實惠的擴展解決方案，以及監控支出。目標是平衡成本節省與高效能和可擴展性的需求。

運算擴展

運算擴展是動態 Kubernetes 環境中應用程式效能的關鍵元件。Kubernetes 透過動態調整運算資源（例如 CPU 和記憶體）來減少浪費，以回應即時需求。此功能有助於避免過度佈建或佈建不足，這也可以節省營運費用。Kubernetes 可讓基礎設施在尖峰時間自動擴展，以及在離峰期間自動縮減，有效消除手動介入的需求。

Kubernetes 的整體運算擴展會自動化擴展程序，進而提升應用程式的彈性和可擴展性，並增強其容錯行為。最後，Kubernetes 的功能可增強卓越營運和生產力。

本節討論下列類型的運算擴展：

- Cluster Autoscaler
- 具有過度佈建的 Cluster Autoscaler
- Karpenter

叢集 AutoScaler

根據 Pod 的需求，Cluster Autoscaler 工具會自動修改大小，方法是在必要時新增節點，或在不需要節點且未充分利用時移除節點。

將 Cluster Autoscaler 工具視為工作負載的擴展解決方案，其中需求逐漸增加，擴展的延遲不是主要問題。

Cluster Autoscaler 工具提供下列主要功能：

- 擴展 – 動態擴展和縮減節點，以回應實際的資源需求。
- Pod 排程 – 有助於確保每個 Pod 正在運作，並擁有運作所需的資源，防止資源不足。
- 成本效益 – 消除未充分利用節點的不必要的操作費用。

具有過度佈建的 Cluster Autoscaler

Cluster Autoscaler 具有與 Cluster Autoscaler 類似的過度佈建函數，可在其中有效率地部署節點，並透過在節點上執行低優先順序 Pod 來節省時間。透過此技術，流量會重新導向至這些 Pod，以因應需求突然激增，讓應用程式繼續運作而不會中斷。

具有過度佈建的 Cluster Autoscaler 提供虛擬 Pod 的功能，可在工作負載非常大、不需要延遲且擴展需要快速時，用來輕鬆部署和執行節點。

具有過度佈建的 Cluster Autoscaler 提供下列主要功能：

- 更好的回應能力 – 透過讓多餘的容量持續可存取，擴展叢集以回應需求激增所需的時間更短。
- 資源保留 – 管理流量中意外的峰值，可在極少的停機時間下有效地協助正確的管理。
- 順暢擴展 – 將資源分配延遲降至最低，有助於更順暢的擴展程序。

Karpenter

Karpenter for Kubernetes 在開放原始碼、效能和可自訂性方面優於傳統 Cluster Autoscaler 工具。使用 Karpenter，您只能自動啟動所需的運算資源，以即時處理叢集的需求。Karpenter 旨在提供更有效率且回應靈敏的擴展。

具有極端可變或複雜工作負載的應用程式，其中快速擴展決策至關重要，受益於使用 Karpenter。它與整合 AWS，提供改善的部署和節點選擇最佳化。

Karpenter 包含下列主要功能：

- 動態佈建 – Karpenter 為用途提供正確的執行個體和大小，並根據 Pod 的特定需求動態佈建新節點。
- 進階排程 – 使用智慧型 Pod 配置，Karpenter 會安排節點，以便盡可能有效地使用 GPU、CPU、記憶體和儲存體等資源。
- 快速擴展 – Karpenter 可以快速擴展，經常在幾秒鐘內做出反應。此回應能力對於流量突然的模式或工作負載需要立即擴展時很有幫助
- 成本效益 – 透過仔細選擇最有效的執行個體，您可以降低營運成本，並利用 提供的其他節省成本替代方案 AWS，例如隨需執行個體、Spot 執行個體和預留執行個體。

工作負載擴展

Kubernetes 中的工作負載擴展對於在動態環境中維護應用程式效能和資源效率至關重要。擴展有助於確保應用程式可以處理各種工作負載，而不會降低效能。Kubernetes 可讓您根據即時指標自動擴展或縮減資源，讓組織快速回應流量的變化。這種彈性不僅改善了使用者體驗，還最佳化了資源使用率，有助於最大限度地降低與使用不足或過度佈建資源相關的成本。

此外，有效的工作負載擴展支援高可用性，確保應用程式即使在尖峰需求期間也能保持回應。Kubernetes 中的工作負載擴展透過動態調整容量以滿足目前需求，讓組織能夠更好地利用雲端資源。

本節討論下列類型的工作負載擴展：

- 水平 Pod Autoscaler
- 叢集比例自動擴展器
- Kubernetes 型事件驅動自動擴展器

Horizontal Pod Autoscaler

Horizontal Pod Autoscaler (HPA) 是一種 Kubernetes 功能，可根據觀察到的 CPU 使用率或其他選取指標，自動調整部署、複寫控制器或狀態集中的 Pod 複本數量。HPA 可確保應用程式可以管理波動的流量和工作負載層級，而不需要手動介入。HPA 提供保留最佳效能的方法，同時有效利用可用資源。

在使用者需求可能隨時間大幅波動的情況下，例如 Web 應用程式、微服務和 APIs，HPA 特別有用。

Horizontal Pod Autoscaler 提供下列主要功能：

- 自動擴展 – HPA 會自動增加或減少 Pod 複本數量，以回應即時指標，確保應用程式可以擴展以滿足使用者需求。
- 指標型決策 – 根據預設，HPA 會根據 CPU 使用率進行擴展。不過，它也可以使用自訂指標，例如記憶體用量或應用程式特定的指標，允許更量身打造的擴展策略。
- 可設定的參數 – 您可以選擇最小和最大複本計數，以及所需的使用率百分比，讓您了解擴展的嚴重性。
- 與 Kubernetes 整合 – 為了監控和修改資源，HPA 可與 Kubernetes 生態系統的其他元素一起運作，包括指標伺服器、Kubernetes API 和自訂指標轉接器。

- 更好的資源使用率 – HPA 透過動態修改 Pod 數量，協助確保有效使用資源、降低成本並改善效能。

叢集比例自動擴展器

Cluster proportional Autoscaler (CPA) 是一種 Kubernetes 元件，旨在根據可用的節點數量自動調整叢集中的 Pod 複本數量。與根據資源使用率指標（例如 CPU 和記憶體）擴展的傳統自動擴展器不同，CPA 會根據叢集本身的大小按比例擴展工作負載。

此方法對於需要維持與叢集大小相關的特定備援或可用性層級的應用程式特別有用，例如 CoreDNS 和其他基礎設施服務。CPA 的一些主要使用案例包括下列項目：

- 過度佈建
- 橫向擴展核心平台服務
- 橫向擴展工作負載，因為 CPA 不需要指標伺服器或 Prometheus Adapter

透過自動化擴展程序，CPA 可協助企業維護平衡的工作負載分佈、提高資源效率，並確保應用程式可適當佈建以滿足使用者需求。

Cluster proportional Autoscaler 提供下列主要功能：

- 節點型擴展 – CPA 會根據可排程的叢集節點數量來擴展複本，讓應用程式能夠根據叢集的大小按比例擴展或收縮。
- 比例調整 – 為確保應用程式可以根據叢集大小的變更進行擴展，自動擴展器會在節點數量與複本數量之間建立比例關係。此關係用於計算工作負載所需的複本數量。
- 與 Kubernetes 元件整合 – CPA 可與 Horizontal Pod Autoscaler (HPA) 等標準 Kubernetes 元件搭配使用，但特別著重於節點計數，而非資源使用率指標。此整合允許更全面的擴展策略。
- Golang API 用戶端 – 為了監控節點數量及其可用核心，CPA 會使用在 Pod 內執行的 Golang API 用戶端，並與 Kubernetes API 伺服器通訊。
- 可設定的參數 – 使用者可以使用 `ConfigMap`設定閾值和擴展參數，讓 CPA 用來修改其行為，並確保其遵循預期的擴展計畫。

Kubernetes 型事件驅動自動擴展器

Kubernetes 型事件驅動自動擴展器 (KEDA) 是一種開放原始碼專案，可讓 Kubernetes 工作負載根據需要處理的事件數量進行擴展。KEDA 透過允許應用程式動態回應各種工作負載來增強應用程式的可擴展性，尤其是事件驅動的工作負載。

透過根據事件自動化擴展程序，KEDA 可協助組織最佳化資源使用率、改善應用程式效能，並減少與過度佈建相關的成本。這種方法對於遇到不同流量模式的應用程式特別有用，例如微型服務、無伺服器函數和即時資料處理系統。

KEDA 提供下列主要功能：

- 事件驅動擴展 – KEDA 可讓您根據外部事件來源定義擴展規則，例如訊息佇列、HTTP 請求或自訂指標。此功能有助於確保應用程式因應即時需求進行擴展。
- 輕量型元件 – KEDA 是一種單一用途的輕量型元件，不需要大量設定或額外負荷即可輕鬆整合至現有的 Kubernetes 叢集。
- 與 Kubernetes 整合 – KEDA 擴展了 Kubernetes 原生元件的功能，例如 Horizontal Pod Autoscaler (HPA)。KEDA 會將事件驅動的擴展功能新增至這些元件，增強而不是取代它們。
- 支援多個事件來源 – KEDA 與各種事件來源相容，包括熱門的訊息平台，例如 RabbitMQ、Apache Kafka 等。由於這種適應性，您可以自訂擴展以符合您的唯一事件驅動型架構。
- 自訂擴展器 – 使用自訂擴展器，您可以指定 KEDA 可用來啟動擴展動作的特定指標，以回應特定商業邏輯或需求。
- 宣告式組態 – 根據 Kubernetes 原則，您可以使用 KEDA 透過使用 Kubernetes 自訂資源來定義應如何進行擴展，以宣告方式描述擴展行為。

網路擴展

Kubernetes 中的網路擴展對於維護服務之間的無縫通訊和支援動態環境中的高效資料流程至關重要。擴展網路基礎設施有助於確保叢集可以處理不同層級的流量，而不會遇到瓶頸或延遲問題。Kubernetes 提供工具和機制來擴展網路資源，允許組織在流量模式變更時維持最佳效能。

網路擴展的這種彈性透過確保快速且可靠的連線來增強整體使用者體驗。網路擴展也會最佳化網路資源的使用，協助降低與未充分利用或過度負荷網路元件相關的成本。

此外，有效的網路擴展對於支援高可用性和彈性至關重要。透過動態調整網路容量和路由，組織可以確保即使在尖峰需求或意外流量尖峰期間，服務仍可存取和回應。此方法可提高雲端聯網資源的使用率，確保基礎設施始終符合目前的需求。

本節討論下列網路擴展類型：

- 適用於 Kubernetes 的 Amazon VPC CNI 外掛程式
- 自訂聯網
- 字首委派
- Amazon VPC Lattice

Kubernetes 專用 Amazon VPC CNI 外掛程式

適用於 Kubernetes 的 Amazon VPC 容器網路界面 (CNI) 外掛程式是 Amazon EKS 中的重要元件。VPC CNI 外掛程式透過整合 Kubernetes Pod 與 Amazon VPC 來提供進階聯網功能。透過此外掛程式，每個 Pod 都會從虛擬私有雲端 (VPC) 指派唯一的 IP 地址，藉此增強網路隔離和效能。隨著叢集的成長和網路需求波動，Amazon VPC CNI 外掛程式在確保高效和可擴展的網路操作方面扮演關鍵角色。

外掛程式會自動管理 VPC 內 IP 地址的配置和路由，簡化網路管理並降低 IP 衝突的風險。它支援字首委派等功能，允許更靈活的 IP 管理。

VPC CNI 外掛程式可協助組織最佳化網路效能、增強安全性，並降低 IP 耗盡的風險。這些功能對於網路需求波動的大型動態環境特別重要，例如微服務架構、高密度工作負載和多租戶應用程式。

Amazon VPC CNI 外掛程式提供下列主要功能：

- 增強型聯網 – VPC CNI 外掛程式允許每個 Pod 直接從 VPC 接收自己的 IP 地址，以提供強大的隔離和網路效能。這種方法對於需要高網路輸送量和低延遲的工作負載至關重要。

- 字首委派 – 為了克服大型叢集中的 IP 地址耗盡問題，字首委派會將較大的 IPs 區塊動態配置給節點，然後細分以供 Pod 使用。此方法可確保有效的 IP 使用率，並簡化網路擴展。
- 自訂聯網 – 使用者可以為 Pod 設定自訂網路介面 (ENIs)，這有助於跨多個介面分配 Pod 流量，減少網路擁塞並改善可擴展性。
- 支援 IPv6 – 透過在 Amazon EKS 叢集中啟用 IPv6，使用者可以大幅擴展可用的 IP 地址空間，促進大型分散式應用程式的擴展，而不受 IPv4 限制。
- 與 Kubernetes 整合 – VPC CNI 外掛程式可與 Kubernetes 網路元件無縫搭配使用，確保跨 Pod、服務和外部端點有效率地管理 IPs，並支援進階功能，例如 Pod 的安全群組。

自訂聯網

Amazon EKS 中的自訂聯網可將特定網路介面指派給 Pod，從而增強對 IP 地址管理和網路流量的控制。這種方法特別適用於 IP 地址耗盡是問題的情況，或出於安全、合規或效能原因需要隔離網路流量的情況。自訂聯網可協助組織有效率地管理 IP 地址空間、隔離流量，並確保可擴展的網路效能。

透過自訂聯網，管理員可以更有效率地管理網路資源。管理員可以使用自訂聯網來協助確保 Pod 具有必要的網路隔離，並且叢集可以擴展而不會遇到 IP 地址限制。

自訂聯網提供下列主要功能：

- 增強型 IP 管理 – 自訂聯網允許將特定網路介面 (ENIs) 指派給 Pod，透過將 Pod 流量分散到多個 ENIs 來協助管理 IP 地址耗盡。此功能在具有高密度工作負載的叢集中尤其重要。
- 流量隔離 – 透過自訂網路介面，您可以根據特定條件來分隔 Pod 流量，例如應用程式類型或安全需求。此方法可更好地控制流量在叢集內外的流動方式。
- 支援 IPv6 – Amazon EKS 中的自訂聯網也支援 IPv6，提供 IPv4 地址限制的解決方案。網路可以有效率地擴展，而不會發生 IP 地址衝突，即使在大規模部署中也是如此。
- 可擴展性和靈活性 – 隨著叢集擴展，自訂聯網可動態管理網路介面。新的 Pod 會獲指派適當的網路資源，無需手動介入。此方法有助於維持靈活且可擴展的網路環境，以適應不斷變化的工作負載。

字首委派

Kubernetes 中的字首委派，特別是在 Amazon EKS 中，旨在隨著叢集擴展來簡化和最佳化 IP 地址管理。透過將較大的 IP 地址區塊（字首）動態配置到節點，字首委派可降低 IP 耗盡的風險，並簡化 IP 空間的管理。

此方法可增強網路效率、將分段降至最低，並協助叢集順利擴展，無需手動調整 IP 範圍。字首委派對於大規模部署、高密度工作負載，以及彈性、動態 IP 管理對於維護網路效能和可擴展性至關重要的環境特別重要。

字首委派提供下列主要功能：

- 高效 IP 地址管理 – 字首委派允許動態配置 IP 範圍，降低 IP 耗盡的風險，並確保高效使用可用的 IP 空間。
- 簡化的網路管理 – 透過允許節點處理自己的 IP 配置，字首委派可將網路分段降至最低，並簡化路由程序，讓您更輕鬆地視需要擴展叢集。
- 支援大規模部署 – 在具有高密度工作負載的大型叢集中，字首委派允許新節點加入叢集，無需手動 IP 範圍調整，即可實現無縫擴展。

Amazon VPC Lattice

Amazon VPC Lattice 可讓您在 VPC 內和跨 VPCs 進行高效且安全service-to-service通訊，尤其是在微服務架構中。除了 (IAM) 整合之外，VPC Lattice 還使用安全群組和網路存取控制清單 AWS Identity and Access Management （網路 ACLs) 等安全措施進行精細的應用程式身分驗證。VPC Lattice 核心的 layer-7 代理服務提供連線、負載平衡、身分驗證、授權、可觀測性、流量管理和服務探索。

透過簡化聯網和安全組態，VPC Lattice 可協助組織最佳化流量管理、增強應用程式效能，以及無縫擴展多個 VPCs和 AWS 區域。這對於需要一致且可靠聯網的分散式應用程式特別有用，例如微服務、跨區域部署和複雜的雲端原生環境。

Amazon VPC Lattice 提供下列主要功能：

- Service-to-service聯網 – VPC Lattice 可簡化微服務架構內服務之間的聯網和安全性組態。它提供統一的平台來管理通訊，讓 服務可以獨立擴展，同時維持高效能和安全性。
- 跨 VPC 聯網 – VPC Lattice 對管理多個 VPCs或區域的流量至關重要。它提供一致的聯網架構，允許服務無縫通訊，無論其實體位置為何。此功能對於跨多個 VPCs 或地理區域的大型應用程式尤其重要。
- 增強型安全管理 – 透過將安全政策直接整合到網路層，VPC Lattice 支援service-to-service通訊。此功能可降低跨分散式環境管理安全性的複雜性，讓您更輕鬆地擴展並降低營運開銷。
- 簡化流量管理 – VPC Lattice 提供進階流量管理功能，包括路由、負載平衡和容錯移轉機制。透過這些功能，流量可有效率地分散到 服務，最佳化網路效能並增強應用程式的可擴展性。

成本最佳化

為了支援有效的資源控制，Kubernetes 成本最小化對於使用此容器協同運作技術的企業至關重要。由於 Kubernetes 設定的複雜性，因此很難正確追蹤其支出，其中包括 Pod 和節點等多個元件。透過運用成本最佳化技術，企業可以查看資源的花費，並適當地將費用指派給部門或專案。

雖然動態擴展具有優勢，但如果未正確管理，可能會導致不可預見的費用。有效率的成本管理有助於僅在真正需要時才配置資源，避免意外的支出激增。

本節討論下列成本最佳化方法：

- Kubecost
- 金絲雀
- AWS Fargate
- Spot 執行個體
- 預留執行個體
- AWS Graviton 執行個體

Kubecost

Kubecost 是一種成本管理解決方案，可協助企業追蹤、控制和最大化雲端基礎設施的支出。它專為 Kubernetes 叢集而打造。Kubecost 可讓您深入了解資源使用率和即時成本意識，讓您更了解使用雲端資源的位置和數量。透過這些洞見，您可以最佳化基礎設施支出、提高資源效率，並對雲端投資做出更明智的決策。

Kubecost 提供下列主要功能：

- 成本分配 – Kubecost 為 Kubernetes 資源提供全面的成本分配，包括工作負載、服務、命名空間和標籤。此功能可協助團隊依環境、專案或團隊監控成本。
- 即時成本監控 – 它提供雲端成本的即時監控，讓組織立即洞察支出模式，並協助防止意外的成本超支。
- 最佳化建議 – Kubecost 提供將資源使用率降至最低的實際建議，包括減少閒置資源、適當調整工作負載大小，以及最大化儲存體費用。
- 預算和提醒 – Kubecost 使用者可以建立預算，並在支出接近或超過預定條件時收到提醒。此功能可協助團隊遵守財務限制。

金絲雀

Goldilocks 是一種 Kubernetes 公用程式，旨在協助使用者最佳化 Kubernetes 工作負載的資源請求和限制。它提供有關如何為在 Kubernetes 叢集中執行的容器設定 CPU 和記憶體資源的建議。這些建議可協助您確保應用程式擁有正確數量的資源，可有效率地執行，而不會浪費資源。此最佳化可以節省成本、改善效能，以及更有效率地使用 Kubernetes 叢集。

Goldilocks 提供下列主要功能：

- 資源建議 – Goldilocks 透過分析 Kubernetes 工作負載的過去 CPU 和記憶體耗用統計資料，決定資源請求和限制的理想設定。透過這樣做，可以更輕鬆地避免佈建不足或過度，這可能會導致效能問題和資源浪費。
- VPA 整合 – Goldilocks 利用 Kubernetes Vertical Pod Autoscaler (VPA) 收集資料並提供建議。它在「建議模式」中執行，表示它實際上不會變更資源設定，但提供這些設定應該是什麼的指導。
- 命名空間型分析 – 透過允許您將特定命名空間設為目標進行分析，Goldilocks 可讓您微調要最佳化和監控的工作負載。
- 視覺化儀表板 – Web 型儀表板會以視覺化方式顯示建議的資源請求和限制，可讓您直接了解資料並對其採取動作。
- 非侵入性操作 – Goldilocks 不會變更叢集的設定，因為它在建議模式下操作。如果需要，您可以在檢閱建議後手動套用建議的資源設定。

AWS Fargate

在 Amazon EKS 的環境中， https://docs.aws.amazon.com/eks/latest/userguide/fargate.htmlAWS Fargate 可讓您在不管理基礎 Amazon EC2 執行個體的情況下執行 Kubernetes Pod。這是一個無伺服器運算引擎，可讓您專注於部署和擴展容器化應用程式，而無需擔心基礎設施。

AWS Fargate 提供下列主要功能：

- 無基礎設施管理 – Fargate 無需佈建、管理或擴展 Amazon EC2 執行個體或 Kubernetes 節點。 會 AWS 處理所有基礎設施管理，包括修補和擴展。
- Pod 層級隔離 – 與以 Amazon EC2 為基礎的工作者節點不同，Fargate 提供任務或 Pod 層級隔離。每個 Pod 都在自己的隔離運算環境中執行，可增強安全性和效能。
- 自動擴展 – Fargate 會根據需求自動擴展 Kubernetes Pod。您不需要管理擴展政策或節點集區。
- 每秒計費 – 您只需為每個 Pod 執行的確切持續時間內消耗的 vCPU 和記憶體資源付費，這是特定工作負載經濟實惠的選項。

- 減少額外負荷 – 透過消除管理 EC2 執行個體的需求，Fargate 可讓您專注於建置和管理應用程式，而不是基礎設施操作。

Spot 執行個體

Spot 執行個體相較於隨需執行個體定價可大幅節省成本，也是在 Amazon EKS 叢集中執行 Amazon EC2 工作者節點的實惠選項。不過，在需要隨需執行個體容量的情況下， AWS 可以中斷 Spot 執行個體。當需要容量時， AWS 可以在 2 分鐘內通知 回收 Spot 執行個體，使其對關鍵且具狀態的工作負載較不可靠。

對於對成本敏感且可承受中斷的工作負載，Amazon EKS 中的 Spot 執行個體是不錯的選擇。在 Kubernetes 叢集中使用 Spot 執行個體和隨需執行個體的組合，可協助您節省成本，而不會犧牲重要工作負載的可用性。

Spot 執行個體提供下列主要功能：

- 節省成本 – Spot 執行個體的成本可能低於隨需執行個體定價，因此非常適合成本敏感的工作負載。
- 適用於容錯工作負載 – 非常適合無狀態、容錯工作負載，例如批次處理、CI/CD 任務、機器學習或大規模資料處理，其中執行個體可以替換而不會發生重大中斷。
- 自動擴展群組整合 – Amazon EKS 會將 Spot 執行個體與 Kubernetes Cluster Autoscaler 整合，以其他可用的 Spot 執行個體或隨需執行個體自動取代中斷的 Spot 執行個體節點。

預留執行個體

在 Amazon EKS 中，預留執行個體是執行 Kubernetes 工作負載的 Amazon EC2 工作者節點定價模式。透過使用預留執行個體，您承諾在 1 年或 3 年期間內使用特定執行個體類型，以換取相較於隨需執行個體定價的成本節省。在 Amazon EKS 中保留執行個體是一種經濟實惠的方式，可在 Amazon EC2 工作者節點上執行一致的長期工作負載。

預留執行個體通常用於 Amazon EC2。不過，如果工作負載需要長期且可預測的用量，則 Amazon EKS 叢集 （即 EC2 執行個體） 中的工作者節點也可以受益於此節省成本的模型。

需要高可用性和一致效能的生產服務、資料庫和其他具狀態應用程式，是非常適合預留執行個體的穩定工作負載範例。

預留執行個體提供下列主要功能：

- 節省成本 – 預留執行個體相較於隨需執行個體可節省成本，取決於期限長度 (1 或 3 年） 和付款計劃（全部預付、部分預付或無預付）。
- 長期承諾 – 您承諾特定執行個體類型、大小和 的 1 年或 3 年期間 AWS 區域。這非常適合穩定且隨時間持續執行的工作負載。
- 可預測定價 – 由於您致力於特定期限，預留執行個體提供可預測的每月或預付成本，讓您更輕鬆地為長期工作負載編列預算。
- 執行個體彈性 – 使用可轉換預留執行個體，您可以在保留期間變更執行個體類型、系列或大小。可轉換預留執行個體比標準預留執行個體提供更多彈性，這不允許變更。
- 保證容量 – 預留執行個體可確保在進行保留的可用區域中提供容量，這對於需要一致運算能力的關鍵工作負載至關重要。
- 無中斷風險 – 與 Spot 執行個體不同，預留執行個體不會受到 中斷 AWS。這使得它們非常適合執行需要保證執行時間的任務關鍵工作負載。

AWS Graviton 執行個體

AWS Graviton 是一系列以 ARM 為基礎的處理器 AWS ，旨在為雲端工作負載提供更高的效能和成本效益。在 Amazon EKS 的環境中，您可以使用 Graviton 執行個體作為工作者節點來執行 Kubernetes 工作負載，從而大幅提高效能並節省成本。

Graviton 執行個體是雲端原生和運算密集型應用程式的絕佳選項，因為它們提供比 x86 執行個體更高的價格效能比。不過，當您考慮採用 Graviton 執行個體時，請將 ARM 相容性納入考量。

AWS Graviton 執行個體提供下列主要功能：

- ARM 型架構 – AWS Graviton 處理器是以 ARM 架構為基礎，與傳統 x86 架構不同，但對於許多工作負載具有高效率。
- 具成本效益 – 相較於以 x86 為基礎的 Amazon EC2 EC2 執行個體通常可提供更好的價格效能。這使它們成為執行 Amazon EKS 之 Kubernetes 叢集的吸引人選項。
- 效能 – Graviton2 處理器是第二代 AWS Graviton，在運算效能、記憶體輸送量和能源效率方面提供顯著改善。它們非常適合 CPU 密集型和記憶體密集型工作負載。
- 多樣化執行個體類型 – Graviton 執行個體有多種系列，例如 t4g、m7g、c7g 和 r7g，涵蓋從一般用途到運算最佳化、記憶體最佳化和爆量工作負載的各種使用案例。
- Amazon EKS 節點群組 – 您可以設定由 Amazon EKS 或自我管理節點群組管理的節點群組，以包含 Graviton 型執行個體。透過此方法，您可以在相同的 Kubernetes 叢集以及 x86 型執行個體上執行針對 ARM 架構最佳化的工作負載。

後續步驟

本指南提供的資訊可協助您在運算擴展、工作負載擴展、網路擴展和成本最佳化方面最佳化 Amazon EKS。透過了解並套用這些概念，組織可以實現高效率、可擴展且符合成本效益的雲端環境，以滿足其動態需求。

有效實作運算和工作負載擴展有助於確保有效率地使用資源，而且應用程式即使在尖峰時間也能維持高效能。接受網路擴展技術，例如自訂聯網和字首委派，支援管理網路資源和無縫可擴展性。強調成本最佳化有助於組織平衡效能與財務效率。

將此指引整合到您的雲端策略，可協助您增強基礎設施的效能和可擴展性，並節省成本。這種全方位的方法可讓您建置強大的雲端環境，以支援組織的成長，並適應不斷變化的業務需求。

資源

AWS 部落格

- 使用 Spot 執行個體為 EKS 的成本最佳化和彈性建置
- 使用 Amazon EKS 將 AWS Graviton 與 x86 CPUs 混合，以最佳化成本和彈性

AWS 文件

- Amazon VPC CNI
- Amazon Elastic Kubernetes Service (AWS 白皮書： 上的部署選項概觀 AWS)
- Amazon EKS 最佳實務指南
- Karpenter
- 進一步了解 Kubecost
- 使用 簡化運算管理 AWS Fargate

其他資源

- Cluster Autoscaling (Kubernetes 文件)
- Goldilocks：建議資源請求的開放原始碼工具 (Fairwinds 部落格)
- 水平 Pod Autoscaling (Kubernetes 文件)
- Kubecost (Kubecost 文件)
- Kubernetes 事件驅動的 Autoscaling (KEDA 文件)

文件歷史紀錄

下表說明本指南的重大變更：擴展 Amazon EKS 基礎設施以最佳化運算、工作負載和網路效能。如果您想收到有關未來更新的通知，可以訂閱 RSS 摘要。

變更	描述	日期
初次出版	—	2024 年 11 月 11 日

AWS 規範性指引詞彙表

以下是 AWS Prescriptive Guidance 提供的策略、指南和模式中常用的術語。若要建議項目，請使用詞彙表末尾的提供意見回饋連結。

數字

7 R

將應用程式移至雲端的七種常見遷移策略。這些策略以 Gartner 在 2011 年確定的 5 R 為基礎，包括以下內容：

- 重構/重新架構 – 充分利用雲端原生功能來移動應用程式並修改其架構，以提高敏捷性、效能和可擴展性。這通常涉及移植作業系統和資料庫。範例：將您的現場部署 Oracle 資料庫 遷移至 Amazon Aurora PostgreSQL 相容版本。
- 平台轉換 (隨即重塑) – 將應用程式移至雲端，並引入一定程度的優化以利用雲端功能。範例：將內部部署 Oracle 資料庫 遷移至 中的 Amazon Relational Database Service (Amazon RDS) for Oracle AWS 雲端。
- 重新購買 (捨棄再購買) – 切換至不同的產品，通常從傳統授權移至 SaaS 模型。範例：將您的客戶關係管理 (CRM) 系統 遷移至 Salesforce.com。
- 主機轉換 (隨即轉移) – 將應用程式移至雲端，而不進行任何變更以利用雲端功能。範例：將您的現場部署 Oracle 資料庫 遷移至 中 EC2 執行個體上的 Oracle AWS 雲端。
- 重新放置 (虛擬機器監視器等級隨即轉移) – 將基礎設施移至雲端，無需購買新硬體、重寫應用程式或修改現有操作。您可以將伺服器從內部部署平台遷移到相同平台的雲端服務。範例：將 Microsoft Hyper-V應用程式 遷移至 AWS。
- 保留 (重新檢視) – 將應用程式保留在來源環境中。其中可能包括需要重要重構的應用程式，且您希望將該工作延遲到以後，以及您想要保留的舊版應用程式，因為沒有業務理由來進行遷移。
- 淘汰 – 解除委任或移除來源環境中不再需要的應用程式。

A

A2A Agent-to-Agent)

支援任務委派和狀態轉移的agent-to-agent協同合作的狀態通訊協定。

ABAC

請參閱屬性型存取控制。

抽象服務

請參閱 受管服務。

ACID

請參閱原子性、一致性、隔離性、耐久性。

主動-主動式遷移

一種資料庫遷移方法，其中來源和目標資料庫保持同步 (透過使用雙向複寫工具或雙重寫入操作)，且兩個資料庫都在遷移期間處理來自連接應用程式的交易。此方法支援小型、受控制批次的遷移，而不需要一次性切換。它更靈活，但需要比主動-被動遷移更多的工作。

主動-被動式遷移

一種資料庫遷移方法，其中來源和目標資料庫保持同步，但只有來源資料庫會在資料複寫至目標資料庫時處理來自連線應用程式的交易。目標資料庫在遷移期間不接受任何交易。

客服人員

一種 AI 系統，可以使用工具自動推理、規劃和採取行動來實現目標。

客服人員操作

在生產環境中大規模建置、測試、部署和執行 AI 代理器的操作實務。

彙總函數

在一組資料列上運作的 SQL 函數，會計算群組的單一傳回值。彙總函數的範例包括 SUM和 MAX。

AI

請參閱人工智慧。

AIOps

請參閱人工智慧操作。

匿名化

永久刪除資料集中個人資訊的程序。匿名化有助於保護個人隱私權。匿名資料不再被視為個人資料。

反模式

經常用於經常性問題的解決方案，其中解決方案具有反生產力、無效或比替代解決方案更有效。

應用程式控制

一種安全方法，僅允許使用核准的應用程式，以協助保護系統免受惡意軟體攻擊。

應用程式組合

有關組織使用的每個應用程式的詳細資訊的集合，包括建置和維護應用程式的成本及其商業價值。此資訊是產品組合探索和分析程序的關鍵，有助於識別要遷移、現代化和優化的應用程式並排定其優先順序。

人工智慧 (AI)

電腦科學領域，致力於使用運算技術來執行通常與人類相關的認知功能，例如學習、解決問題和識別模式。如需詳細資訊，請參閱什麼是人工智慧？

人工智慧操作 (AIOps)

使用機器學習技術解決操作問題、減少操作事件和人工干預以及提高服務品質的程序。如需有關如何在 AWS 遷移策略中使用 AIOps 的詳細資訊，請參閱操作整合指南。

非對稱加密

一種加密演算法，它使用一對金鑰：一個用於加密的公有金鑰和一個用於解密的私有金鑰。您可以共用公有金鑰，因為它不用於解密，但對私有金鑰存取應受到高度限制。

原子性、一致性、隔離性、耐久性 (ACID)

一組軟體屬性，即使在出現錯誤、電源故障或其他問題的情況下，也能確保資料庫的資料有效性和操作可靠性。

屬性型存取控制 (ABAC)

根據使用者屬性 (例如部門、工作職責和團隊名稱) 建立精細許可的實務。如需詳細資訊，請參閱《 AWS Identity and Access Management (IAM) 文件》中的 ABAC for AWS。

授權資料來源

存放主要版本資料的位置，被視為最可靠的資訊來源。您可以將授權資料來源中的資料複製到其他位置，以處理或修改資料，例如匿名、修訂或假名化資料。

可用區域

中的不同位置 AWS 區域 ，可隔離其他可用區域中的故障，並提供相同區域中其他可用區域的低成本、低延遲網路連線能力。

AWS 雲端採用架構 (AWS CAF)

的指導方針和最佳實務架構 AWS ，可協助組織制定高效且有效的計劃，以成功地移至雲端。 AWS CAF 將指導方針組織到六個重點領域：業務、人員、治理、平台、安全和營運。業務、人員和控管層面著重於業務技能和程序；平台、安全和操作層面著重於技術技能和程序。例如，人員層面針對處理人力資源 (HR)、人員配備功能和人員管理的利害關係人。因此， AWS CAF 為人員開發、訓練和通訊提供指引，協助組織做好成功採用雲端的準備。如需詳細資訊，請參閱 AWS CAF 網站和 AWS CAF 白皮書。

AWS 工作負載資格架構 (AWS WQF)

評估資料庫遷移工作負載、建議遷移策略並提供工作預估值的工具。 AWS WQF 隨附於 AWS Schema Conversion Tool (AWS SCT)。它會分析資料庫結構描述和程式碼物件、應用程式程式碼、相依性和效能特性，並提供評估報告。

B

錯誤的機器人

旨在中斷或傷害個人或組織的機器人。

BCP

請參閱業務持續性規劃。

行為圖

資源行為的統一互動式檢視，以及一段時間後的互動。您可以將行為圖與 Amazon Detective 搭配使用來檢查失敗的登入嘗試、可疑的 API 呼叫和類似動作。如需詳細資訊，請參閱偵測文件中的行為圖中的資料。

大端序系統

首先儲存最高有效位元組的系統。另請參閱 Endianness。

二進制分類

預測二進制結果的過程 (兩個可能的類別之一)。例如，ML 模型可能需要預測諸如「此電子郵件是否是垃圾郵件？」等問題 或「產品是書還是汽車？」

Bloom 篩選條件

一種機率性、記憶體高效的資料結構，用於測試元素是否為集的成員。

藍/綠部署

一種部署策略，您可以在其中建立兩個不同但相同的環境。您可以在一個環境（藍色）中執行目前的應用程式版本，並在另一個環境（綠色）中執行新的應用程式版本。此策略可協助您快速復原，並將影響降至最低。

機器人

透過網際網路執行自動化任務並模擬人類活動或互動的軟體應用程式。有些機器人有用或有益，例如在網際網路上編製資訊索引的 Web 爬蟲程式。某些其他機器人稱為惡意機器人，旨在中斷或傷害個人或組織。

殭屍網路

受到惡意軟體感染且受單一方控制之機器人的網路，稱為機器人繼承器或機器人運算子。殭屍網路是擴展機器人及其影響的最佳已知機制。

分支

程式碼儲存庫包含的區域。儲存庫中建立的第一個分支是主要分支。您可以從現有分支建立新分支，然後在新分支中開發功能或修正錯誤。您建立用來建立功能的分支通常稱為功能分支。當準備好發佈功能時，可以將功能分支合併回主要分支。如需詳細資訊，請參閱關於分支 (GitHub 文件)。

碎片存取

在特殊情況下，以及透過核准的程序，讓使用者快速取得他們通常無權存取 AWS 帳戶 之 的存取權。如需詳細資訊，請參閱 Well-Architected 指南中的 AWS 實作打破玻璃程序指標。

棕地策略

環境中的現有基礎設施。對系統架構採用棕地策略時，可以根據目前系統和基礎設施的限制來設計架構。如果正在擴展現有基礎設施，則可能會混合棕地和綠地策略。

緩衝快取

儲存最常存取資料的記憶體區域。

業務能力

業務如何創造價值 (例如，銷售、客戶服務或營銷)。業務能力可驅動微服務架構和開發決策。如需詳細資訊，請參閱在 AWS上執行容器化微服務白皮書的圍繞業務能力進行組織部分。

業務連續性規劃 (BCP)

一種解決破壞性事件 (如大規模遷移) 對營運的潛在影響並使業務能夠快速恢復營運的計畫。

C

CAF

請參閱AWS 雲端採用架構。

Canary 部署

版本對最終使用者的緩慢和增量版本。當您有信心時，您可以部署新版本，並完全取代目前的版本。

CCoE

請參閱 Cloud Center of Excellence。

CDC

請參閱變更資料擷取。

變更資料擷取 (CDC)

追蹤對資料來源 (例如資料庫表格) 的變更並記錄有關變更的中繼資料的程序。您可以將 CDC 用於各種用途，例如稽核或複寫目標系統中的變更以保持同步。

混沌工程

故意引入故障或破壞性事件，以測試系統的彈性。您可以使用 AWS Fault Injection Service (AWS FIS) 執行實驗，為您的 AWS 工作負載帶來壓力，並評估其回應。

CI/CD

請參閱持續整合和持續交付。

分類

有助於產生預測的分類程序。用於分類問題的 ML 模型可預測離散值。離散值永遠彼此不同。例如，模型可能需要評估影像中是否有汽車。

公民開發人員

在沒有專業技術技能的情況下，使用無程式碼/低程式碼平台建立 AI 應用程式的商業使用者。

用戶端加密

在目標 AWS 服務 接收資料之前，在本機加密資料。

雲端卓越中心 (CCoE)

一個多學科團隊，可推動整個組織的雲端採用工作，包括開發雲端最佳實務、調動資源、制定遷移時間表以及領導組織進行大規模轉型。如需詳細資訊，請參閱 AWS 雲端 企業策略部落格上的 CCoE 文章。

雲端運算

通常用於遠端資料儲存和 IoT 裝置管理的雲端技術。雲端運算通常連接到邊緣運算技術。

雲端操作模型

在 IT 組織中，用於建置、成熟和最佳化一或多個雲端環境的操作模型。如需詳細資訊，請參閱建置您的雲端操作模型。

採用雲端階段

組織在遷移至 時通常會經歷的四個階段 AWS 雲端：

- 專案 – 執行一些與雲端相關的專案以進行概念驗證和學習用途
- 基礎 – 進行基礎投資以擴展雲端採用 (例如，建立登陸區域、定義 CCoE、建立營運模型)
- 遷移 – 遷移個別應用程式
- 重塑 – 優化產品和服務，並在雲端中創新

部落格文章中的 Stephen Orban 定義了這些階段： AWS 雲端 企業策略部落格上的邁向雲端優先之旅和採用階段。如需有關它們如何與 AWS 遷移策略關聯的資訊，請參閱遷移整備指南。

CMDB

請參閱組態管理資料庫。

程式碼儲存庫

透過版本控制程序來儲存及更新原始程式碼和其他資產 (例如文件、範例和指令碼) 的位置。常見的雲端儲存庫包括 GitHub或 Bitbucket Cloud。程式碼的每個版本都稱為分支。在微服務結構中，每個儲存庫都專用於單個功能。單一 CI/CD 管道可以使用多個儲存庫。

冷快取

一種緩衝快取，它是空的、未填充的，或者包含過時或不相關的資料。這會影響效能，因為資料庫執行個體必須從主記憶體或磁碟讀取，這比從緩衝快取讀取更慢。

冷資料

很少存取且通常是歷史資料的資料。查詢這類資料時，通常可接受慢查詢。將此資料移至效能較低且成本較低的儲存層或類別，可以降低成本。

電腦視覺 (CV)

使用機器學習從數位影像和影片等視覺化格式分析和擷取資訊的 AI 欄位。例如，Amazon SageMaker AI 提供 CV 的影像處理演算法。

組態偏離

對於工作負載，組態會從預期狀態變更。這可能會導致工作負載不合規，而且通常是漸進和無意的。

組態管理資料庫 (CMDB)

儲存和管理有關資料庫及其 IT 環境的資訊的儲存庫，同時包括硬體和軟體元件及其組態。您通常在遷移的產品組合探索和分析階段使用 CMDB 中的資料。

一致性套件

您可以組合的 AWS Config 規則和修補動作集合，以自訂您的合規和安全檢查。您可以使用 YAML 範本，將一致性套件部署為 AWS 帳戶 和 區域中或整個組織的單一實體。如需詳細資訊，請參閱 AWS Config 文件中的一致性套件。

持續整合和持續交付 (CI/CD)

自動化軟體發行程序的來源、建置、測試、暫存和生產階段的程序。CI/CD 通常被描述為管道。CI/CD 可協助您將程序自動化、提升生產力、改善程式碼品質以及加快交付速度。如需詳細資訊，請參閱持續交付的優點。CD 也可表示持續部署。如需詳細資訊，請參閱持續交付與持續部署。

CV

請參閱電腦視覺。

D

靜態資料

網路中靜止的資料，例如儲存中的資料。

資料分類

根據重要性和敏感性來識別和分類網路資料的程序。它是所有網路安全風險管理策略的關鍵組成部分，因為它可以協助您確定適當的資料保護和保留控制。資料分類是 AWS Well-Architected Framework 中安全支柱的元件。如需詳細資訊，請參閱資料分類。

資料偏離

生產資料與用於訓練 ML 模型的資料之間有意義的變化，或輸入資料隨時間有意義的變更。資料偏離可以降低 ML 模型預測的整體品質、準確性和公平性。

傳輸中的資料

在您的網路中主動移動的資料，例如在網路資源之間移動。

資料網格

架構架構，提供分散式、分散式資料擁有權與集中式管理。

資料最小化

僅收集和處理嚴格必要資料的原則。在 中實作資料最小化 AWS 雲端 可以降低隱私權風險、成本和分析碳足跡。

資料周邊

AWS 環境中的一組預防性防護機制，可協助確保只有信任的身分才能從預期的網路存取信任的資源。如需詳細資訊，請參閱在 上建置資料周邊 AWS。

資料預先處理

將原始資料轉換成 ML 模型可輕鬆剖析的格式。預處理資料可能意味著移除某些欄或列，並解決遺失、不一致或重複的值。

資料來源

在整個資料生命週期中追蹤資料的來源和歷史記錄的程序，例如資料的產生、傳輸和儲存方式。

資料主體

正在收集和處理資料的個人。

資料倉儲

支援商業智慧的資料管理系統，例如 分析。資料倉儲通常包含大量歷史資料，通常用於查詢和分析。

資料庫定義語言 (DDL)

用於建立或修改資料庫中資料表和物件之結構的陳述式或命令。

資料庫處理語言 (DML)

用於修改 (插入、更新和刪除) 資料庫中資訊的陳述式或命令。

DDL

請參閱資料庫定義語言。

深度整體

結合多個深度學習模型進行預測。可以使用深度整體來獲得更準確的預測或估計預測中的不確定性。

深度學習

一個機器學習子領域，它使用多層人工神經網路來識別感興趣的輸入資料與目標變數之間的對應關係。

深度防禦

這是一種資訊安全方法，其中一系列的安全機制和控制項會在整個電腦網路中精心分層，以保護網路和其中資料的機密性、完整性和可用性。當您在 上採用此策略時 AWS，您可以在 AWS Organizations 結構的不同層新增多個控制項，以協助保護資源。例如，defense-in-depth方法可能會結合多重要素驗證、網路分割和加密。

委派的管理員

在 中 AWS Organizations，相容的服務可以註冊 AWS 成員帳戶，以管理組織的帳戶和管理該服務的許可。此帳戶稱為該服務的委派管理員。如需詳細資訊和相容服務清單，請參閱 AWS Organizations 文件中的可搭配 AWS Organizations運作的服務。

deployment

在目標環境中提供應用程式、新功能或程式碼修正的程序。部署涉及在程式碼庫中實作變更，然後在應用程式環境中建置和執行該程式碼庫。

開發環境

請參閱 環境。

偵測性控制

一種安全控制，用於在事件發生後偵測、記錄和提醒。這些控制是第二道防線，提醒您注意繞過現有預防性控制的安全事件。如需詳細資訊，請參閱在 AWS上實作安全控制中的偵測性控制。

開發值串流映射 (DVSM)

一種程序，用於識別對軟體開發生命週期中的速度和品質造成負面影響的限制並排定優先順序。DVSM 擴展了最初專為精簡製造實務設計的價值串流映射程序。它著重於透過軟體開發程序建立和移動價值所需的步驟和團隊。

數位分身

真實世界系統的虛擬呈現，例如建築物、工廠、工業設備或生產線。數位分身支援預測性維護、遠端監控和生產最佳化。

維度資料表

在星星結構描述中，較小的資料表包含有關事實資料表中量化資料的資料屬性。維度資料表屬性通常是文字欄位或離散數字，其行為類似於文字。這些屬性通常用於查詢限制、篩選和結果集標記。

災難

防止工作負載或系統在其主要部署位置中實現其業務目標的事件。這些事件可能是自然災難、技術故障或人為動作的結果，例如意外設定錯誤或惡意軟體攻擊。

災難復原 (DR)

您用來將災難造成的停機時間和資料遺失降至最低的策略和程序。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的 上工作負載的災難復原 AWS：雲端中的復原。

DML

請參閱資料庫處理語言。

領域驅動的設計

一種開發複雜軟體系統的方法，它會將其元件與每個元件所服務的不斷發展的領域或核心業務目標相關聯。Eric Evans 在其著作 Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003) 中介紹了這一概念。如需有關如何將領域驅動的設計與 strangler fig 模式搭配使用的資訊，請參閱使用容器和 Amazon API Gateway 逐步現代化舊版 Microsoft ASP.NET (ASMX) Web 服務。

DR

請參閱災難復原。

偏離偵測

追蹤與基準組態的偏差。例如，您可以使用 AWS CloudFormation 來偵測系統資源中的偏離，也可以使用 AWS Control Tower 來偵測登陸區域中可能影響控管要求合規性的變更。 https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-stack-drift.html

DVSM

請參閱開發值串流映射。

E

EDA

請參閱探索性資料分析。

EDI

請參閱電子資料交換。

邊緣運算

提升 IoT 網路邊緣智慧型裝置運算能力的技術。與雲端運算相比，邊緣運算可以減少通訊延遲並改善回應時間。

電子資料交換 (EDI)

在組織之間自動交換商業文件。如需詳細資訊，請參閱什麼是電子資料交換。

加密

一種運算程序，可將人類可讀取的純文字資料轉換為加密文字。

加密金鑰

由加密演算法產生的隨機位元的加密字串。金鑰長度可能有所不同，每個金鑰的設計都是不可預測且唯一的。

端序

位元組在電腦記憶體中的儲存順序。大端序系統首先儲存最高有效位元組。小端序系統首先儲存最低有效位元組。

端點

請參閱 服務端點。

端點服務

您可以在虛擬私有雲端 (VPC) 中託管以與其他使用者共用的服務。您可以使用 建立端點服務，AWS PrivateLink 並將許可授予其他 AWS 帳戶 或 AWS Identity and Access Management (IAM) 委託人。這些帳戶或主體可以透過建立介面 VPC 端點私下連接至您的端點服務。如需詳細資訊，請參閱 Amazon Virtual Private Cloud (Amazon VPC) 文件中的建立端點服務。

企業資源規劃 (ERP)

一種系統，可自動化和管理企業的關鍵業務流程 （例如會計、MES 和專案管理）。

信封加密

使用另一個加密金鑰對某個加密金鑰進行加密的程序。如需詳細資訊，請參閱 AWS Key Management Service (AWS KMS) 文件中的信封加密。

環境

執行中應用程式的執行個體。以下是雲端運算中常見的環境類型：

- 開發環境 – 執行中應用程式的執行個體，只有負責維護應用程式的核心團隊才能使用。開發環境用來測試變更，然後再將開發環境提升到較高的環境。此類型的環境有時稱為測試環境。
- 較低的環境 – 應用程式的所有開發環境，例如用於初始建置和測試的開發環境。
- 生產環境 – 最終使用者可以存取的執行中應用程式的執行個體。在 CI/CD 管道中，生產環境是最後一個部署環境。
- 較高的環境 – 核心開發團隊以外的使用者可存取的所有環境。這可能包括生產環境、生產前環境以及用於使用者接受度測試的環境。

epic

在敏捷方法中，有助於組織工作並排定工作優先順序的功能類別。epic 提供要求和實作任務的高層級描述。例如， AWS CAF 安全概念包括身分和存取管理、偵測控制、基礎設施安全、資料保護和事件回應。如需有關 AWS 遷移策略中的 Epic 的詳細資訊，請參閱計畫實作指南。

ERP

請參閱企業資源規劃。

探索性資料分析 (EDA)

分析資料集以了解其主要特性的過程。您收集或彙總資料，然後執行初步調查以尋找模式、偵測異常並檢查假設。透過計算摘要統計並建立資料可視化來執行 EDA。

F

事實資料表

星狀結構描述中的中央資料表。它存放有關業務操作的量化資料。一般而言，事實資料表包含兩種類型的資料欄：包含度量的資料，以及包含維度資料表外部索引鍵的資料欄。

快速失敗

一種使用頻繁和增量測試來縮短開發生命週期的理念。這是敏捷方法的關鍵部分。

故障隔離界限

在 中 AWS 雲端，像是可用區域 AWS 區域、控制平面或資料平面等界限會限制故障的影響，並有助於改善工作負載的彈性。如需詳細資訊，請參閱AWS 故障隔離界限。

功能分支

請參閱分支。

特徵

用來進行預測的輸入資料。例如，在製造環境中，特徵可能是定期從製造生產線擷取的影像。

功能重要性

特徵對於模型的預測有多重要。這通常表示為可以透過各種技術來計算的數值得分，例如 Shapley Additive Explanations (SHAP) 和積分梯度。如需詳細資訊，請參閱 的機器學習模型可解譯性 AWS。

特徵轉換

優化 ML 程序的資料，包括使用其他來源豐富資料、調整值、或從單一資料欄位擷取多組資訊。這可讓 ML 模型從資料中受益。例如，如果將「2021-05-27 00:15:37」日期劃分為「2021」、「五月」、「週四」和「15」，則可以協助學習演算法學習與不同資料元件相關聯的細微模式。

少量擷取提示

在要求 LLM 執行類似的任務之前，提供少量示範任務和所需輸出的範例。此技術是內容內學習的應用程式，其中模型會從內嵌在提示中的範例 (快照) 中學習。少量的提示對於需要特定格式、推理或網域知識的任務來說非常有效。另請參閱零鏡頭提示。

FGAC

請參閱精細存取控制。

精細存取控制 (FGAC)

使用多個條件來允許或拒絕存取請求。

閃切遷移

一種資料庫遷移方法，透過變更資料擷取使用連續資料複寫，以盡可能在最短的時間內遷移資料，而不是使用分階段方法。目標是將停機時間降至最低。

FM

請參閱基礎模型。

基礎模型 (FM)

大型深度學習神經網路，已針對廣義和未標記資料的大量資料集進行訓練。FMs 能夠執行各種一般任務，例如了解語言、產生文字和影像，以及以自然語言交談。如需詳細資訊，請參閱什麼是基礎模型。

FM 閘道

集中式中介，可控制和標準化對基礎模型的存取。也稱為 LLM 閘道。

G

生成式 AI

已針對大量資料進行訓練的 AI 模型子集，可使用簡單的文字提示建立新的內容和成品，例如影像、影片、文字和音訊。如需詳細資訊，請參閱什麼是生成式 AI。

地理封鎖

請參閱地理限制。

地理限制 (地理封鎖)

Amazon CloudFront 中的選項，可防止特定國家/地區的使用者存取內容分發。您可以使用允許清單或封鎖清單來指定核准和禁止的國家/地區。如需詳細資訊，請參閱 CloudFront 文件中的限制內容的地理分佈。

Gitflow 工作流程

這是一種方法，其中較低和較高環境在原始碼儲存庫中使用不同分支。Gitflow 工作流程被視為舊版，而以幹線為基礎的工作流程是現代、偏好的方法。

黃金影像

系統或軟體的快照，做為部署該系統或軟體新執行個體的範本。例如，在製造中，黃金映像可用於在多個裝置上佈建軟體，並有助於提高裝置製造操作的速度、可擴展性和生產力。

綠地策略

新環境中缺乏現有基礎設施。對系統架構採用綠地策略時，可以選擇所有新技術，而不會限制與現有基礎設施的相容性，也稱為棕地。如果正在擴展現有基礎設施，則可能會混合棕地和綠地策略。

防護機制

有助於跨組織單位 (OU) 來管控資源、政策和合規的高層級規則。預防性防護機制會強制執行政策，以確保符合合規標準。透過使用服務控制政策和 IAM 許可界限來將其實作。偵測性防護機制可

偵測政策違規和合規問題，並產生提醒以便修正。它們是透過使用 AWS Config AWS Security Hub CSPM、Amazon GuardDuty、、Amazon Inspector AWS Trusted Advisor和自訂 AWS Lambda 檢查來實作。

護欄 (AI)

可篩選、驗證和限制代理程式輸入和輸出的安全機制，以協助確保負責任且安全的 AI 行為。

H

HA

請參閱高可用性。

異質資料庫遷移

將來源資料庫遷移至使用不同資料庫引擎的目標資料庫 (例如，Oracle 至 Amazon Aurora)。異質遷移通常是重新架構工作的一部分，而轉換結構描述可能是一項複雜任務。AWS 提供有助於結構描述轉換的 AWS SCT。

高可用性 (HA)

工作負載在遇到挑戰或災難時持續運作的能力，無需介入。HA 系統的設計目的是自動容錯移轉、持續提供高品質的效能，並處理不同的負載和故障，並將效能影響降至最低。

歷史現代化

一種方法，用於現代化和升級操作技術 (OT) 系統，以更好地滿足製造業的需求。歷史資料是一種資料庫，用於從工廠中的各種來源收集和存放資料。

保留資料

從用於訓練機器學習模型的資料集中保留的部分歷史標記資料。您可以使用保留資料，透過比較模型預測與保留資料來評估模型效能。

human-in-the-loop (HitL)

一種工作流程模式，其中代理程式執行會在關鍵決策點暫停進行人工審核和核准。

異質資料庫遷移

將您的來源資料庫遷移至共用相同資料庫引擎的目標資料庫 (例如，Microsoft SQL Server 至 Amazon RDS for SQL Server)。同質遷移通常是主機轉換或平台轉換工作的一部分。您可以使用原生資料庫公用程式來遷移結構描述。

熱資料

經常存取的資料，例如即時資料或最近的轉譯資料。此資料通常需要高效能儲存層或類別，才能提供快速的查詢回應。

修補程序

緊急修正生產環境中的關鍵問題。由於其緊迫性，通常會在典型 DevOps 發行工作流程之外執行修補程式。

超級護理期間

在切換後，遷移團隊在雲端管理和監控遷移的應用程式以解決任何問題的時段。通常，此期間的長度為 1-4 天。在超級護理期間結束時，遷移團隊通常會將應用程式的責任轉移給雲端營運團隊。

I

IaC

請參閱基礎設施即程式碼。

身分型政策

連接至一或多個 IAM 主體的政策，可定義其在 AWS 雲端 環境中的許可。

閒置應用程式

90 天期間 CPU 和記憶體平均使用率在 5% 至 20% 之間的應用程式。在遷移專案中，通常會淘汰這些應用程式或將其保留在內部部署。

IIoT

請參閱工業物聯網。

不可變的基礎設施

為生產工作負載部署新基礎設施的模型，而不是更新、修補或修改現有的基礎設施。不可變基礎設施本質上比可變基礎設施更一致、可靠且可預測。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的使用不可變基礎設施部署最佳實務。

傳入 (輸入) VPC

在 AWS 多帳戶架構中，接受、檢查和路由來自應用程式外部之網路連線的 VPC。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

增量遷移

一種切換策略，您可以在其中將應用程式分成小部分遷移，而不是執行單一、完整的切換。例如，您最初可能只將一些微服務或使用者移至新系統。確認所有項目都正常運作之後，您可以逐步移動其他微服務或使用者，直到可以解除委任舊式系統。此策略可降低與大型遷移關聯的風險。

工業 4.0

2016 年 Klaus Schwab 推出的術語，透過連線能力、即時資料、自動化、分析和 AI/ML 的進展，指製造程序的現代化。

基礎設施

應用程式環境中包含的所有資源和資產。

基礎設施即程式碼 (IaC)

透過一組組態檔案來佈建和管理應用程式基礎設施的程序。IaC 旨在協助您集中管理基礎設施，標準化資源並快速擴展，以便新環境可重複、可靠且一致。

工業物聯網 (IIoT)

在製造業、能源、汽車、醫療保健、生命科學和農業等產業領域使用網際網路連線的感測器和裝置。如需詳細資訊，請參閱建立工業物聯網 (IIoT) 數位轉型策略。

檢查 VPC

在 AWS 多帳戶架構中，集中式 VPC，可管理 VPCs 之間 （在相同或不同的 中 AWS 區域)、網際網路和內部部署網路之間的網路流量檢查。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

物聯網 (IoT)

具有內嵌式感測器或處理器的相連實體物體網路，其透過網際網路或本地通訊網路與其他裝置和系統進行通訊。如需詳細資訊，請參閱什麼是 IoT？

可解釋性

機器學習模型的一個特徵，描述了人類能夠理解模型的預測如何依賴於其輸入的程度。如需詳細資訊，請參閱 的機器學習模型可解譯性 AWS。

IoT

請參閱物聯網。

IT 資訊庫 (ITIL)

一組用於交付 IT 服務並使這些服務與業務需求保持一致的最佳實務。ITIL 為 ITSM 提供了基礎。

IT 服務管理 (ITSM)

與組織的設計、實作、管理和支援 IT 服務關聯的活動。如需有關將雲端操作與 ITSM 工具整合的資訊，請參閱操作整合指南。

ITIL

請參閱 IT 資訊庫。

ITSM

請參閱 IT 服務管理。

L

標籤型存取控制 (LBAC)

強制存取控制 (MAC) 的實作，其中使用者和資料本身都會獲得明確指派的安全標籤值。使用者安全標籤和資料安全標籤之間的交集會決定使用者可以看到哪些資料列和資料欄。

登陸區域

登陸區域是架構良好的多帳戶 AWS 環境，可擴展且安全。這是一個起點，您的組織可以從此起點快速啟動和部署工作負載與應用程式，並對其安全和基礎設施環境充滿信心。如需有關登陸區域的詳細資訊，請參閱設定安全且可擴展的多帳戶 AWS 環境。

大型語言模型 (LLM)

預先訓練大量資料的深度學習 AI 模型。LLM 可以執行多個任務，例如回答問題、摘要文件、將文字翻譯成其他語言，以及完成句子。如需詳細資訊，請參閱什麼是 LLMs。

大型遷移

遷移 300 部或更多伺服器。

LBAC

請參閱標籤型存取控制。

最低權限

授予執行任務所需之最低許可的安全最佳實務。如需詳細資訊，請參閱 IAM 文件中的套用最低權限許可。

隨即轉移

請參閱 7 Rs。

小端序系統

首先儲存最低有效位元組的系統。另請參閱 Endianness。

LLM

請參閱大型語言模型。

較低的環境

請參閱 環境。

M

機器學習 (ML)

一種使用演算法和技術進行模式識別和學習的人工智慧。機器學習會進行分析並從記錄的資料 (例如物聯網 (IoT) 資料) 中學習，以根據模式產生統計模型。如需詳細資訊，請參閱機器學習。

主要分支

請參閱分支。

惡意軟體

旨在危及電腦安全或隱私權的軟體。惡意軟體可能會中斷電腦系統、洩露敏感資訊，或取得未經授權的存取。惡意軟體的範例包括病毒、蠕蟲、勒索軟體、特洛伊木馬、間諜軟體和鍵盤記錄器。

受管服務

AWS 服務 會 AWS 操作基礎設施層、作業系統和平台，而您會存取端點來存放和擷取資料。Amazon Simple Storage Service (Amazon S3) 和 Amazon DynamoDB 是受管服務的範例。這些也稱為抽象服務。

製造執行系統 (MES)

一種軟體系統，用於追蹤、監控、記錄和控制生產程序，將原物料轉換為現場成品。

MAP

請參閱遷移加速計劃。

MCP

請參閱模型內容通訊協定。

模型內容通訊協定 (MCP)

用於代理程式對工具通訊的無狀態通訊協定。

MCP 伺服器

透過模型內容通訊協定公開一或多個工具的服務。

機制

建立工具、推動工具採用，然後檢查結果以進行調整的完整程序。機制是在操作時強化和改善自身的循環。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的建置機制。

成員帳戶

屬於組織一部分的管理帳戶 AWS 帳戶 以外的所有 AWS Organizations。帳戶一次只能是一個組織的成員。

製造執行系統

請參閱製造執行系統。

訊息佇列遙測傳輸 (MQTT)

根據發佈/訂閱模式的輕量型machine-to-machine(M2M) 通訊協定，適用於資源受限的 IoT 裝置。

微服務

一種小型的獨立服務，它可透過定義明確的 API 進行通訊，通常由小型獨立團隊擁有。例如，保險系統可能包含對應至業務能力 (例如銷售或行銷) 或子領域 (例如購買、索賠或分析) 的微服務。微服務的優點包括靈活性、彈性擴展、輕鬆部署、可重複使用的程式碼和適應力。如需詳細資訊，請參閱使用無 AWS 伺服器服務整合微服務。

微服務架構

一種使用獨立元件來建置應用程式的方法，這些元件會以微服務形式執行每個應用程式程序。這些微服務會使用輕量型 API，透過明確定義的介面進行通訊。此架構中的每個微服務都可以進行更新、部署和擴展，以滿足應用程式特定功能的需求。如需詳細資訊，請參閱在 上實作微服務 AWS。

Migration Acceleration Program (MAP)

此 AWS 計畫提供諮詢支援、訓練和服務，以協助組織建立強大的營運基礎，以移至雲端，並協助抵銷遷移的初始成本。MAP 包括用於有條不紊地執行舊式遷移的遷移方法以及一組用於自動化和加速常見遷移案例的工具。

大規模遷移

將大部分應用程式組合依波次移至雲端的程序，在每個波次中，都會以更快的速度移動更多應用程式。此階段使用從早期階段學到的最佳實務和經驗教訓來實作團隊、工具和流程的遷移工廠，以透過自動化和敏捷交付簡化工作負載的遷移。這是 AWS 遷移策略的第三階段。

遷移工廠

可透過自動化、敏捷的方法簡化工作負載遷移的跨職能團隊。遷移工廠團隊通常包括營運、業務分析師和擁有者、遷移工程師、開發人員以及從事 Sprint 工作的 DevOps 專業人員。20% 至 50% 之間的企業應用程式組合包含可透過工廠方法優化的重複模式。如需詳細資訊，請參閱此內容集中的遷移工廠的討論和雲端遷移工廠指南。

遷移中繼資料

有關完成遷移所需的應用程式和伺服器的資訊。每種遷移模式都需要一組不同的遷移中繼資料。遷移中繼資料的範例包括目標子網路、安全群組和 AWS 帳戶。

遷移模式

可重複的遷移任務，詳細描述遷移策略、遷移目的地以及所使用的遷移應用程式或服務。範例：使用 AWS Application Migration Service 重新託管遷移至 Amazon EC2。

遷移組合評定 (MPA)

線上工具，提供驗證商業案例以遷移至 的資訊 AWS 雲端。MPA 提供詳細的組合評定 (伺服器適當規模、定價、總體擁有成本比較、遷移成本分析) 以及遷移規劃 (應用程式資料分析和資料收集、應用程式分組、遷移優先順序，以及波次規劃)。MPA 工具 （需要登入） 可供所有 AWS 顧問和 APN 合作夥伴顧問免費使用。

遷移準備程度評定 (MRA)

使用 AWS CAF 取得組織雲端整備狀態的洞見、識別優缺點，以及建立行動計劃以消除已識別差距的程序。如需詳細資訊，請參閱遷移準備程度指南。MRA 是 AWS 遷移策略的第一階段。

遷移策略

用來將工作負載遷移至 的方法 AWS 雲端。如需詳細資訊，請參閱本詞彙表中的 7 個 Rs 項目，並請參閱動員您的組織以加速大規模遷移。

機器學習 (ML)

請參閱機器學習。

現代化

將過時的 (舊版或單一) 應用程式及其基礎架構轉換為雲端中靈活、富有彈性且高度可用的系統，以降低成本、提高效率並充分利用創新。如需詳細資訊，請參閱《》中的現代化應用程式的策略 AWS 雲端。

現代化準備程度評定

這項評估可協助判斷組織應用程式的現代化準備程度；識別優點、風險和相依性；並確定組織能夠在多大程度上支援這些應用程式的未來狀態。評定的結果就是目標架構的藍圖、詳細說明現代化程序的開發階段和里程碑的路線圖、以及解決已發現的差距之行動計畫。如需詳細資訊，請參閱《》中的評估應用程式的現代化準備 AWS 雲端程度。

單一應用程式 (單一)

透過緊密結合的程序作為單一服務執行的應用程式。單一應用程式有幾個缺點。如果一個應用程式功能遇到需求激增，則必須擴展整個架構。當程式碼庫增長時，新增或改進單一應用程式的功能也會變得更加複雜。若要解決這些問題，可以使用微服務架構。如需詳細資訊，請參閱將單一體系分解為微服務。

MPA

請參閱遷移產品組合評估。

MQTT

請參閱訊息佇列遙測傳輸。

多類別分類

一個有助於產生多類別預測的過程 (預測兩個以上的結果之一)。例如，機器學習模型可能會詢問「此產品是書籍、汽車還是電話？」 或者「這個客戶對哪種產品類別最感興趣？」

可變基礎設施

更新和修改生產工作負載現有基礎設施的模型。為了提高一致性、可靠性和可預測性， AWS Well-Architected Framework 建議使用不可變基礎設施做為最佳實務。

O

OAC

請參閱原始存取控制。

OAI

請參閱原始存取身分。

OCM

請參閱組織變更管理。

離線遷移

一種遷移方法，可在遷移過程中刪除來源工作負載。此方法涉及延長停機時間，通常用於小型非關鍵工作負載。

OI

請參閱 操作整合。

OLA

請參閱操作層級協議。

線上遷移

一種遷移方法，無需離線即可將來源工作負載複製到目標系統。連接至工作負載的應用程式可在遷移期間繼續運作。此方法涉及零至最短停機時間，通常用於關鍵的生產工作負載。

OPC-UA

請參閱開放程序通訊 - 統一架構。

開放程序通訊 - 統一架構 (OPC-UA)

用於工業自動化的machine-to-machine(M2M) 通訊協定。OPC-UA 提供資料加密、身分驗證和授權機制的互通性標準。

操作水準協議 (OLA)

一份協議，闡明 IT 職能群組承諾向彼此提供的內容，以支援服務水準協議 (SLA)。

操作整備審查 (ORR)

問題及相關最佳實務的檢查清單，可協助您了解、評估、預防或減少事件和可能失敗的範圍。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的操作準備度審查 (ORR)。

操作技術 (OT)

使用實體環境控制工業操作、設備和基礎設施的硬體和軟體系統。在製造中，OT 和資訊技術 (IT) 系統的整合是工業 4.0 轉型的關鍵重點。

操作整合 (OI)

在雲端中將操作現代化的程序，其中包括準備程度規劃、自動化和整合。如需詳細資訊，請參閱操作整合指南。

組織追蹤

由 建立的線索 AWS CloudTrail ，會記錄 AWS 帳戶 組織中所有 的所有事件 AWS Organizations。在屬於組織的每個 AWS 帳戶 中建立此追蹤，它會跟蹤每個帳戶中的活動。如需詳細資訊，請參閱 CloudTrail 文件中的建立組織追蹤。

組織變更管理 (OCM)

用於從人員、文化和領導力層面管理重大、顛覆性業務轉型的架構。OCM 透過加速變更採用、解決過渡問題，以及推動文化和組織變更，協助組織為新系統和策略做好準備，並轉移至新系統和策略。在 AWS 遷移策略中，此架構稱為人員加速，因為雲端採用專案所需的變更速度。如需詳細資訊，請參閱 OCM 指南。

原始存取控制 (OAC)

CloudFront 中的增強型選項，用於限制存取以保護 Amazon Simple Storage Service (Amazon S3) 內容。OAC 支援所有 S3 儲存貯體、使用 AWS KMS (SSE-KMS) 的伺服器端加密 AWS 區域，以及對 S3 儲存貯體的動態PUT和DELETE請求。

原始存取身分 (OAI)

CloudFront 中的一個選項，用於限制存取以保護 Amazon S3 內容。當您使用 OAI 時，CloudFront 會建立一個可供 Amazon S3 進行驗證的主體。經驗證的主體只能透過特定 CloudFront 分發來存取 S3 儲存貯體中的內容。另請參閱 OAC，它可提供更精細且增強的存取控制。

ORR

請參閱操作整備審核。

OT

請參閱操作技術。

傳出 (輸出) VPC

在 AWS 多帳戶架構中，處理從應用程式內啟動之網路連線的 VPC。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

P

許可界限

附接至 IAM 主體的 IAM 管理政策，可設定使用者或角色擁有的最大許可。如需詳細資訊，請參閱 IAM 文件中的許可界限。

個人身分識別資訊 (PII)

當直接檢視或與其他相關資料配對時，可用來合理推斷個人身分的資訊。PII 的範例包括名稱、地址和聯絡資訊。

PII

請參閱個人身分識別資訊。

手冊

一組預先定義的步驟，可擷取與遷移關聯的工作，例如在雲端中提供核心操作功能。手冊可以採用指令碼、自動化執行手冊或操作現代化環境所需的程序或步驟摘要的形式。

PLC

請參閱可程式設計邏輯控制器。

PLM

請參閱產品生命週期管理。

政策

可定義許可的物件（請參閱身分型政策）、指定存取條件（請參閱資源型政策），或定義組織中所有帳戶的最大許可 AWS Organizations（請參閱服務控制政策）。

混合持久性

根據資料存取模式和其他需求，獨立選擇微服務的資料儲存技術。如果您的微服務具有相同的資料儲存技術，則其可能會遇到實作挑戰或效能不佳。如果微服務使用最適合其需求的資料儲存，則可以更輕鬆地實作並達到更好的效能和可擴展性。

組合評定

探索、分析應用程式組合並排定其優先順序以規劃遷移的程序。如需詳細資訊，請參閱評估遷移準備程度。

述詞

傳回 `true`或 的查詢條件`false`，通常位於 `WHERE`子句中。

述詞下推

一種資料庫查詢最佳化技術，可在傳輸前篩選查詢中的資料。這可減少必須從關聯式資料庫擷取和處理的資料量，並改善查詢效能。

預防性控制

旨在防止事件發生的安全控制。這些控制是第一道防線，可協助防止對網路的未經授權存取或不必要變更。如需詳細資訊，請參閱在 AWS上實作安全控制中的預防性控制。

委託人

中可執行動作和存取資源 AWS 的實體。此實體通常是 AWS 帳戶、IAM 角色或使用者的根使用者。如需詳細資訊，請參閱 IAM 文件中角色術語和概念中的主體。

設計隱私權

透過整個開發程序將隱私權納入考量的系統工程方法。

私有託管區域

一種容器，它包含有關您希望 Amazon Route 53 如何回應一個或多個 VPC 內的域及其子域之 DNS 查詢的資訊。如需詳細資訊，請參閱 Route 53 文件中的使用私有託管區域。

主動控制

旨在防止部署不合規資源的安全控制。這些控制項會在佈建資源之前對其進行掃描。如果資源不符合控制項，則不會佈建。如需詳細資訊，請參閱 AWS Control Tower 文件中的控制項參考指南，並參閱實作安全控制項中的主動控制項。 AWS

產品生命週期管理 (PLM)

管理產品整個生命週期的資料和程序，從設計、開發和啟動，到成長和成熟，再到拒絕和移除。

生產環境

請參閱 環境。

可程式設計邏輯控制器 (PLC)

在製造中，高度可靠、可調整的電腦，可監控機器並自動化製造程序。

提示鏈結

使用一個 LLM 提示的輸出做為下一個提示的輸入，以產生更好的回應。此技術用於將複雜任務分解為子任務，或反覆精簡或展開初步回應。它有助於提高模型回應的準確性和相關性，並允許更精細、個人化的結果。

擬匿名化

將資料集中的個人識別符取代為預留位置值的程序。假名化有助於保護個人隱私權。假名化資料仍被視為個人資料。

發佈/訂閱 (pub/sub)

一種模式，可啟用微服務之間的非同步通訊，以提高可擴展性和回應能力。例如，在微服務型 MES 中，微服務可以將事件訊息發佈到其他微服務可訂閱的頻道。系統可以新增新的微服務，而無需變更發佈服務。

Q

查詢計劃

一系列步驟，如指示，用於存取 SQL 關聯式資料庫系統中的資料。

查詢計劃迴歸

在資料庫服務優化工具選擇的計畫比對資料庫環境進行指定的變更之前的計畫不太理想時。這可能因為對統計資料、限制條件、環境設定、查詢參數繫結的變更以及資料庫引擎的更新所導致。

R

RACI 矩陣

請參閱負責、負責、諮詢、告知 (RACI)。

RAG

請參閱擷取增強生成。

勒索軟體

一種惡意軟體，旨在阻止對計算機系統或資料的存取，直到付款為止。

RASCI 矩陣

請參閱負責、負責、諮詢、告知 (RACI)。

RCAC

請參閱資料列和資料欄存取控制。

僅供讀取複本

用於唯讀用途的資料庫複本。您可以將查詢路由至僅供讀取複本以減少主資料庫的負載。

重新架構師

請參閱 7 個 R。

復原點目標 (RPO)

自上次資料復原點以來可接受的時間上限。這會決定最後一個復原點與服務中斷之間可接受的資料遺失。

復原時間目標 (RTO)

服務中斷與服務還原之間的可接受延遲上限。

重構

請參閱 7 個 R。

區域

地理區域中的 AWS 資源集合。每個 AWS 區域 都獨立於其他 ，以提供容錯能力、穩定性和彈性。如需詳細資訊，請參閱指定 AWS 區域 您的帳戶可以使用哪些 。

迴歸

預測數值的 ML 技術。例如，為了解決「這房子會賣什麼價格？」的問題 ML 模型可以使用線性迴歸模型，根據已知的房屋事實 (例如，平方英尺) 來預測房屋的銷售價格。

重新託管

請參閱 7 Rs。

版本

在部署程序中，它是將變更提升至生產環境的動作。

重新定位

請參閱 7 Rs。

Replatform

請參閱 7 Rs。

回購

請參閱 7 Rs。

彈性

應用程式抵禦中斷或從中斷中復原的能力。在 中規劃彈性時，高可用性和災難復原是常見的考量 AWS 雲端。如需詳細資訊，請參閱AWS 雲端 彈性。

資源型政策

附接至資源的政策，例如 Amazon S3 儲存貯體、端點或加密金鑰。這種類型的政策會指定允許存取哪些主體、支援的動作以及必須滿足的任何其他條件。

負責者、當責者、事先諮詢者和事後告知者 (RACI) 矩陣

矩陣，定義所有涉及遷移活動和雲端操作之各方的角色和責任。矩陣名稱衍生自矩陣中定義的責任類型：負責人 (R)、責任 (A)、已諮詢 (C) 和知情 (I)。支援 (S) 類型為選用。如果您包含支援，則矩陣稱為 RASCI 矩陣，如果您排除它，則稱為 RACI 矩陣。

回應性控制

一種安全控制，旨在驅動不良事件或偏離安全基準的補救措施。如需詳細資訊，請參閱在 AWS上實作安全控制中的回應性控制。

保留

請參閱 7 個 R。

淘汰

請參閱 7 個 R。

檢索增強生成 (RAG)

一種生成式 AI 技術，其中 LLM 會在產生回應之前參考訓練資料來源以外的授權資料來源。例如，RAG 模型可能會對組織的知識庫或自訂資料執行語意搜尋。如需詳細資訊，請參閱什麼是 RAG。

輪換

定期更新秘密的程序，讓攻擊者更難存取登入資料。

資料列和資料欄存取控制 (RCAC)

使用已定義存取規則的基本、彈性 SQL 表達式。RCAC 包含資料列許可和資料欄遮罩。

RPO

請參閱復原點目標。

RTO

請參閱復原時間目標。

執行手冊

執行特定任務所需的一組手動或自動程序。這些通常是為了簡化重複性操作或錯誤率較高的程序而建置。

S

SAML 2.0

許多身分提供者 (IdP) 使用的開放標準。此功能會啟用聯合單一登入 (SSO)，讓使用者可以登入 AWS 管理主控台 或呼叫 AWS API 操作，而不必為您組織中的每個人在 IAM 中建立使用者。如需有關以 SAML 2.0 為基礎的聯合詳細資訊，請參閱 IAM 文件中的關於以 SAML 2.0 為基礎的聯合。

斯卡達

請參閱監督控制和資料擷取。

SCP

請參閱服務控制政策。

秘密

您以加密形式存放的 AWS Secrets Manager機密或限制資訊，例如密碼或使用者登入資料。它由秘密值及其中繼資料組成。秘密值可以是二進位、單一字串或多個字串。如需詳細資訊，請參閱 Secrets Manager 秘密中的內容？ Secrets Manager 文件中的 。

設計安全性

透過整個開發程序將安全性納入考量的系統工程方法。

安全控制

一種技術或管理防護機制，它可預防、偵測或降低威脅行為者利用安全漏洞的能力。安全控制有四種主要類型：預防性、偵測性、回應性和主動性。

安全強化

減少受攻擊面以使其更能抵抗攻擊的過程。這可能包括一些動作，例如移除不再需要的資源、實作授予最低權限的安全最佳實務、或停用組態檔案中不必要的功能。

安全資訊與事件管理 (SIEM) 系統

結合安全資訊管理 (SIM) 和安全事件管理 (SEM) 系統的工具與服務。SIEM 系統會收集、監控和分析來自伺服器、網路、裝置和其他來源的資料，以偵測威脅和安全漏洞，並產生提醒。

安全回應自動化

預先定義和程式設計的動作，旨在自動回應或修復安全事件。這些自動化可做為偵測或回應式安全控制，協助您實作 AWS 安全最佳實務。自動化回應動作的範例包括修改 VPC 安全群組、修補 Amazon EC2 執行個體或輪換登入資料。

伺服器端加密

由 AWS 服務 接收資料的 在其目的地加密資料。

服務控制政策 (SCP)

為 AWS Organizations中的組織的所有帳戶提供集中控制許可的政策。SCP 會定義防護機制或設定管理員可委派給使用者或角色的動作限制。您可以使用 SCP 作為允許清單或拒絕清單，以指定允許或禁止哪些服務或動作。如需詳細資訊，請參閱 AWS Organizations 文件中的服務控制政策。

服務端點

的進入點 URL AWS 服務。您可以使用端點，透過程式設計方式連接至目標服務。如需詳細資訊，請參閱 AWS 一般參考 中的 AWS 服務 端點。

服務水準協議 (SLA)

一份協議，闡明 IT 團隊承諾向客戶提供的服務，例如服務正常執行時間和效能。

服務層級指標 (SLI)

服務效能方面的測量，例如其錯誤率、可用性或輸送量。

服務層級目標 (SLO)

代表服務運作狀態的目標指標，由服務層級指標測量。

共同責任模式

描述您與 共同 AWS 承擔雲端安全與合規責任的模型。 AWS 負責雲端的安全，而 負責雲端的安全。如需詳細資訊，請參閱共同責任模式。

陰影 AI

在組織內受管頻道之外建置或使用的未授權 AI 應用程式。

SIEM

請參閱安全資訊和事件管理系統。

單一故障點 (SPOF)

應用程式的單一關鍵元件故障，可能會中斷系統。

SLA

請參閱服務層級協議。

SLI

請參閱服務層級指標。

SLO

請參閱服務層級目標。

先拆分後播種模型

擴展和加速現代化專案的模式。定義新功能和產品版本時，核心團隊會進行拆分以建立新的產品團隊。這有助於擴展組織的能力和服務，提高開發人員生產力，並支援快速創新。如需詳細資訊，請參閱 中的階段式應用程式現代化方法 AWS 雲端。

SPOF

請參閱單一故障點。

星狀結構描述

使用一個大型事實資料表來存放交易或測量資料的資料庫組織結構，並使用一或多個較小的維度資料表來存放資料屬性。此結構旨在用於資料倉儲或商業智慧用途。

Strangler Fig 模式

一種現代化單一系統的方法，它會逐步重寫和取代系統功能，直到舊式系統停止使用為止。此模式源自無花果藤，它長成一棵馴化樹並最終戰勝且取代了其宿主。該模式由 Martin Fowler 引入，作為重寫單一系統時管理風險的方式。如需有關如何套用此模式的範例，請參閱使用容器和 Amazon API Gateway 逐步現代化舊版 Microsoft ASP.NET (ASMX) Web 服務。

子網

您 VPC 中的 IP 地址範圍。子網必須位於單一可用區域。

監控控制和資料擷取 (SCADA)

在製造中，使用硬體和軟體來監控實體資產和生產操作的系統。

對稱加密

使用相同金鑰來加密及解密資料的加密演算法。

合成測試

以模擬使用者互動的方式測試系統，以偵測潛在問題或監控效能。您可以使用 Amazon CloudWatch Synthetics 來建立這些測試。

系統提示

一種向 LLM 提供內容、指示或指導方針以指示其行為的技術。系統提示有助於設定內容，並建立與使用者互動的規則。

T

標籤

做為中繼資料以組織 AWS 資源的鍵值對。標籤可協助您管理、識別、組織、搜尋及篩選資源。如需詳細資訊，請參閱標記您的 AWS 資源。

目標變數

您嘗試在受監督的 ML 中預測的值。這也被稱為結果變數。例如，在製造設定中，目標變數可能是產品瑕疵。

任務清單

用於透過執行手冊追蹤進度的工具。任務清單包含執行手冊的概觀以及要完成的一般任務清單。對於每個一般任務，它包括所需的預估時間量、擁有者和進度。

測試環境

請參閱 環境。

訓練

為 ML 模型提供資料以供學習。訓練資料必須包含正確答案。學習演算法會在訓練資料中尋找將輸入資料屬性映射至目標的模式 (您想要預測的答案)。它會輸出擷取這些模式的 ML 模型。可以使用 ML 模型，來預測您不知道的目標新資料。

tool

代理程式可以叫用以在外部系統中執行操作的函數或 API。

傳輸閘道

可以用於互連 VPC 和內部部署網路的網路傳輸中樞。如需詳細資訊，請參閱 AWS Transit Gateway 文件中的什麼是傳輸閘道。

主幹型工作流程

這是一種方法，開發人員可在功能分支中本地建置和測試功能，然後將這些變更合併到主要分支中。然後，主要分支會依序建置到開發環境、生產前環境和生產環境中。

受信任的存取權

將許可授予您指定的服務，以代表您在組織中 AWS Organizations 及其帳戶中執行任務。受信任的服務會在需要該角色時，在每個帳戶中建立服務連結角色，以便為您執行管理工作。如需詳細資訊，請參閱 文件中的 AWS Organizations 搭配使用 AWS Organizations 與其他 AWS 服務。

調校

變更訓練程序的各個層面，以提高 ML 模型的準確性。例如，可以透過產生標籤集、新增標籤、然後在不同的設定下多次重複這些步驟來訓練 ML 模型，以優化模型。

雙比薩團隊

兩個比薩就能吃飽的小型 DevOps 團隊。雙披薩團隊規模可確保軟體開發中的最佳協作。

U

不確定性

這是一個概念，指的是不精確、不完整或未知的資訊，其可能會破壞預測性 ML 模型的可靠性。有兩種類型的不確定性：認知不確定性是由有限的、不完整的資料引起的，而隨機不確定性是由資料中固有的噪聲和隨機性引起的。

未區分的任務

也稱為繁重工作，這是建立和操作應用程式的必要工作，但不為最終使用者提供直接價值或提供競爭優勢。未區分任務的範例包括採購、維護和容量規劃。

較高的環境

請參閱 環境。

V

清空

一種資料庫維護操作，涉及增量更新後的清理工作，以回收儲存並提升效能。

版本控制

追蹤變更的程序和工具，例如儲存庫中原始程式碼的變更。

VPC 對等互連

兩個 VPC 之間的連線，可讓您使用私有 IP 地址路由流量。如需詳細資訊，請參閱 Amazon VPC 文件中的什麼是 VPC 對等互連。

漏洞

危害系統安全性的軟體或硬體瑕疵。

W

暖快取

包含經常存取的目前相關資料的緩衝快取。資料庫執行個體可以從緩衝快取讀取，這比從主記憶體或磁碟讀取更快。

暖資料

不常存取的資料。查詢這類資料時，通常可接受中等緩慢的查詢。

視窗函數

SQL 函數，對與目前記錄在某種程度上相關的資料列群組執行計算。視窗函數適用於處理任務，例如根據目前資料列的相對位置計算移動平均值或存取資料列的值。

工作負載

提供商業價值的資源和程式碼集合，例如面向客戶的應用程式或後端流程。

工作串流

遷移專案中負責一組特定任務的功能群組。每個工作串流都是獨立的，但支援專案中的其他工作串流。例如，組合工作串流負責排定應用程式、波次規劃和收集遷移中繼資料的優先順序。組合工作串流將這些資產交付至遷移工作串流，然後再遷移伺服器和應用程式。

WORM

請參閱寫入一次，讀取許多。

WQF

請參閱AWS 工作負載資格架構。

寫入一次，讀取許多 (WORM)

儲存模型，可一次性寫入資料，並防止刪除或修改資料。授權使用者可以視需要多次讀取資料，但無法變更資料。此資料儲存基礎設施被視為不可變。

Z

零時差入侵

利用零時差漏洞的攻擊，通常是惡意軟體。

零時差漏洞

生產系統中未緩解的缺陷或漏洞。威脅行為者可以使用這種類型的漏洞來攻擊系統。開發人員經常因為攻擊而意識到漏洞。

零鏡頭提示

提供 LLM 執行任務的指示，但沒有可協助引導任務的範例 (快照)。LLM 必須使用其預先訓練的知識來處理任務。零鏡頭提示的有效性取決於任務的複雜性和提示的品質。另請參閱少量擷取提示。

殭屍應用程式

CPU 和記憶體平均使用率低於 5% 的應用程式。在遷移專案中，通常會淘汰這些應用程式。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。