自動駕駛資料架構 (ADDF) 安全和操作指南

AWS 規定指引

AWS 規定指引: 自動駕駛資料架構 (ADDF) 安全和操作指南

Table of Contents

自動駕駛資料架構 (ADDF) 安全和操作指南

Andreas Falkenberg、Junjie Tang、Torsten Reitemeyer 和 Srinivas Reddy Cheruku，Amazon Web Services (AWS)

2022 年 11 月 (文件歷史記錄)

自動駕駛資料架構 (ADDF) 是一個開放原始碼專案，旨在為想要實作進階駕駛輔助系統 (ADAS) 常見任務的汽車團隊提供可重複使用的模組化程式碼成品，例如，設定集中式資料儲存、資料處理管道、視覺化機制、搜尋介面、模擬工作負載、分析介面和預先建置的儀表板。使用 ADDF，您可以共用、修改或建立完全可自訂的模組，從而減少建立和部署這些解決方案所需的工作量。

本指南旨在協助您了解在 AWS 雲端 中安全部署和操作 ADDF 的最佳實務。將探討下列主題：

- 架構和術語 – 檢閱一般架構、工作流程和重要術語。
- 共同責任模式 – 了解您的角色以及 AWS 在保護 ADDF 部署和雲端資源方面的角色。
- 安全審查程序 – 由於 ADDF 是一個開放原始碼專案，因此檢閱 AWS 和貢獻者如何完成安全審查。
- 內建安全功能 – 檢閱如何將安全最佳實務和功能內建到 ADDF 開放原始碼專案及其部署架構中。
- 安全設定和操作 – 了解如何在 AWS 雲端 中部署和操作 ADDF。

目標對象

本指南適用於負責評估、部署、自訂和操作 ADDF 的開發營運 (DevOps) 團隊、基礎設施工程師、管理員、IT 安全人員和事件回應團隊。您可以將本指南中的建議套用於概念驗證或生產環境。

本指南假設您先前不了解 ADDF。但是，我們建議您在繼續之前先閱讀 ADDF 讀我 (GitHub)。

目標業務成果

本指南旨在協助您在開發和生產環境中更自信、安全地設定和操作 ADDF。

ADDF 架構和術語

在了解本指南中的安全和操作主題之前，請務必先深入了解自動駕駛資料架構 (ADDF) 的術語、元件和架構。本節包含下列主題：

ADDF 術語

ADDF 的重要術語如下：

- ADDF 模組 – 模組是基礎設施即程式碼 (IaC)，用於在進階駕駛輔助系統 (ADAS) 中實作常見任務。常見任務包括設定集中式資料儲存、資料處理管道、視覺化機制、搜尋介面、模擬工作負載、分析介面和預先建置的儀表板。您可以根據自己的需求建立模組，也可以重複使用或自訂現有的模組。

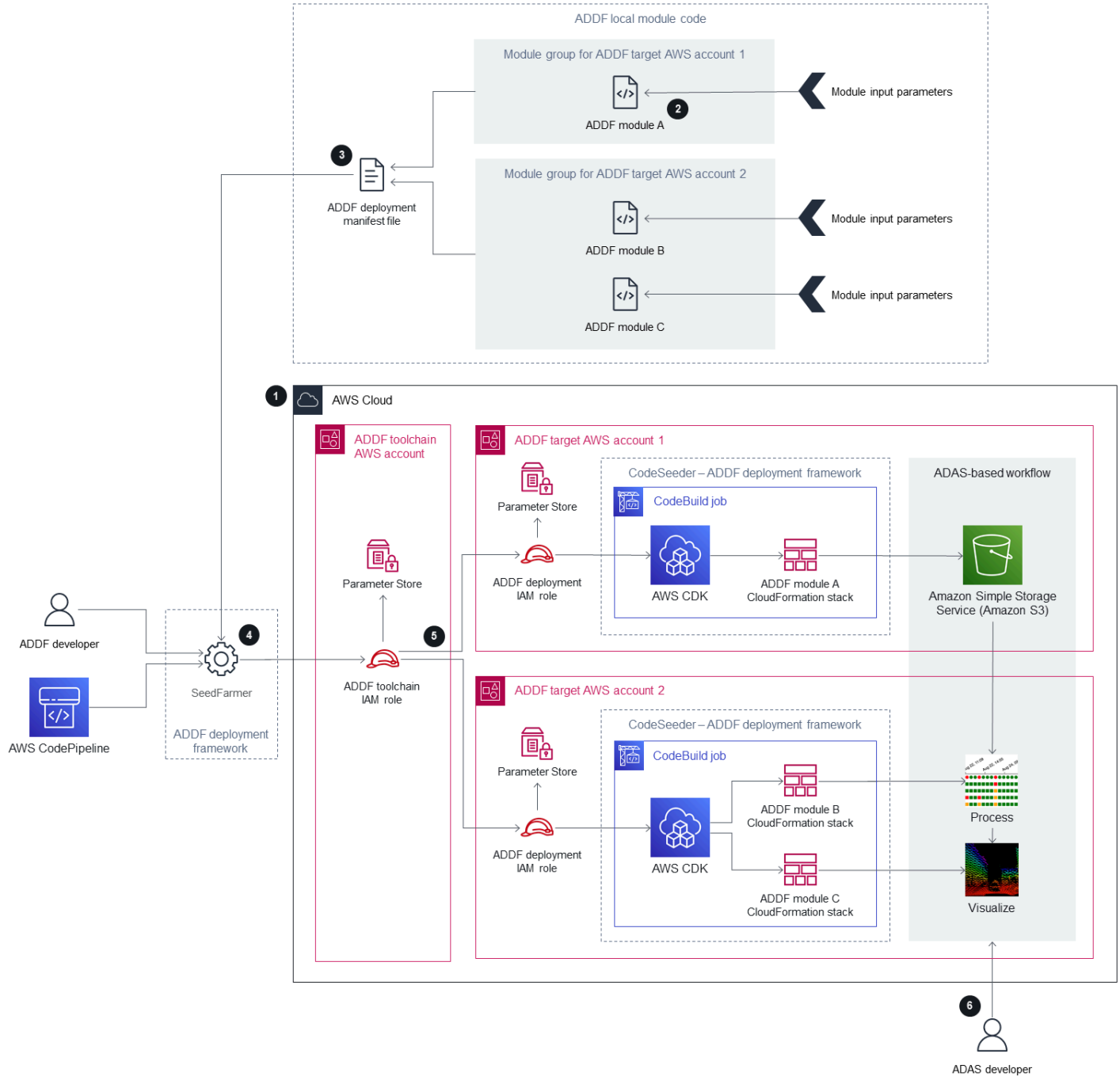
  您可以使用 AWS Cloud Development Kit (AWS CDK) 定義 ADDF 模組，或者您可以使用任何常見的 IaC 架構 (例如 Hashicorp Terraform 或 AWS CloudFormation) 來實作 ADDF 模組。模組具有一組輸入參數。輸入參數可能取決於來自其他模組的輸出值。ADDF 模組是 ADDF 目標 AWS 帳戶 的最小部署單位。
- ADDF 部署清單檔案 – 此檔案定義獨立 ADDF 模組的協同運作。協同運作指的是模組的部署順序。在 ADDF 部署清單檔案中，您可以使用 ADDF 群組將相關模組分組在一起。在此檔案中，您也可以定義 ADDF 工具鏈 AWS 帳戶、ADDF 目標 AWS 帳戶 和目標 AWS 區域。
- ADDF 部署架構 – 此架構根據 ADDF 部署清單檔案中定義的協同運作將 ADDF 模組部署到 ADDF 目標 AWS 帳戶 中。ADDF 部署架構透過使用下列 AWS 開放原始碼專案實作：
  - SeedFarmer (GitHub) – SeedFarmer 是用於 ADDF 部署的 CLI 工具。其管理每個模組狀態、準備並封裝模組程式碼、為 ADDF 部署角色建立最低權限政策，同時提供 CodeSeeder 用於部署的語意指令。您可以直接與 SeedFarmer 互動以執行 ADDF 部署，也可以將其整合到持續整合和持續部署 (CI/CD) 管道中。
  - CodeSeeder (GitHub) – CodeSeeder 透過 AWS CodeBuild 作業部署任意基礎設施即程式碼套件。SeedFarmer 會自動協同運作和執行 CodeSeeder。只有 SeedFarmer 直接與 CodeSeeder 互動。

  ADDF 部署架構旨在支援單一帳戶和多帳戶架構中的部署。根據您組織的要求，由您決定是否需要單一帳戶或多帳戶架構。

- ADF 工具鏈 AWS 帳戶– 此帳戶協調和管理模組部署到 ADF 目標 AWS 帳戶，根據 ADF 部署資訊清單檔案中的定義。一個 ADDF 部署只能具有一個 ADDF 工具鏈 AWS 帳戶。在單一帳戶架構中，ADDF 工具鏈 AWS 帳戶 也是 ADDF 目標 AWS 帳戶。此帳戶包含 AWS Identity and Access Management (IAM) 角色，稱為 ADDF 工具鏈 IAM 角色，在 ADDF 部署過程中由 SeedFarmer 擔任。在本指南中，我們將 ADDF 工具鏈 AWS 帳戶 稱為工具鏈帳戶。
- ADDF 目標 AWS 帳戶 – 這些是您要在其中部署 ADDF 模組的目標帳戶。您可以具有一或多個目標帳戶。這些帳戶包含 ADDF 部署清單檔案及其映射模組中描述的資源和應用程式邏輯。在單一帳戶架構中，ADDF 目標 AWS 帳戶 也是 ADDF 工具鏈 AWS 帳戶。每個 ADDF 目標帳戶都包含 IAM 角色，稱為 ADDF 部署 IAM 角色，在部署過程中由 CodeSeeder 擔任。在本指南中，我們將 ADDF 目標 AWS 帳戶 稱為目標帳戶。
- ADDF 執行個體 – 當您在雲端中部署 ADDF 和模組時 (如 ADDF 部署清單檔案中所定義)，這將成為 ADDF 執行個體。ADDF 執行個體可以具有單一帳戶或多帳戶架構，且您可以部署多個 ADDF 執行個體。如需有關選擇執行個體數量和為您的使用案例設計帳戶架構的詳細資訊，請參閱 定義您的 ADDF 架構。

ADDF 架構

下圖顯示了 AWS 雲端 中 ADDF 執行個體的高層級架構。它顯示了多帳戶架構，包括一個專用工具鏈帳戶和兩個目標帳戶。本指南討論使用 ADDF 將資源部署至目標帳戶的端對端程序。



1. 建立和引導 ADDF AWS 帳戶。

   為了正常運行，每個帳戶都必須引導至 ADDF 和 AWS CDK。如果這是新的 ADDF 部署，或者您要新增目標帳戶，請執行下列操作：

   a. 在工具鏈帳戶和每個目標帳戶中引導 AWS CDK。如需說明，請參閱引導 (AWS CDK 文件)。ADDF 使用 AWS CDK 部署基礎設施。

b. 在工具鏈帳戶和每個目標帳戶中引導 ADDF。如需說明，請參閱 ADDF 部署指南中的引導 AWS 帳戶。這將設定 SeedFarmer 和 CodeSeeder 所需的所有 ADDF 特定 IAM 角色。

> **Note**
>
> 只有當您最初部署 ADDF 或新增目標帳戶時，才需要執行此步驟。此步驟不是重複發生 ADDF 部署至已建立的 ADDF 執行個體的一部分。

2. 建立或自訂 ADDF 模組。

   根據您嘗試解決的特定問題建立或自訂 ADDF 模組。您的模組應代表一個獨立的任務或一組任務。視需要定義模組的輸入參數，並使用模組輸出值作為其他模組的輸入參數。

3. 在 ADDF 部署清單檔案中定義模組協同運作。

   在 ADDF 清單檔案中，將模組整理為群組並定義這些群組之間的部署順序和相依性。在此檔案中，您還為每個 ADDF 群組及其模組指定單一工具鏈帳戶和目標帳戶 (包括 AWS 區域)。

4. 評估 ADDF 部署清單檔案並建立部署範圍。

   ADDF 開發人員或 CI/CD 管道 (例如 AWS CodePipeline) 透過呼叫 CLI 工具 SeedFarmer 開始評估 ADDF 部署清單檔案。若要開始評估：

   - SeedFarmer 使用 ADDF 部署清單檔案作為評估的輸入參數。
   - 若要承擔 ADDF 工具鏈 IAM 角色，SeedFarmer 需要與步驟 1 中 ADDF 引導過程中定義的相同、有效的 IAM 角色或使用者憑證。

   如果 SeedFarmer 沒有正確的憑證來擔任 ADDF 工具鏈 IAM 角色或無法存取 ADDF 部署清單檔案，則評估不會啟動。

   如果 SeedFarmer 可以開始評估，則它將擔任工具鏈帳戶中的 ADDF 工具鏈 IAM 角色。從那裡，SeedFarmer 可以透過擔任該帳戶中的 ADDF 部署 IAM 角色來存取任何目標帳戶。然後，SeedFarmer 嘗試讀取工具鏈帳戶和目標帳戶中的任何 ADDF 中繼資料。以下其中一種情況將發生：

   - 如果沒有要讀取的 ADDF 中繼資料，表示這是新的 ADDF 執行個體。SeedFarmer 會確定部署範圍是整個 ADDF 部署清單檔案及其內容。
   - 如果 ADDF 中繼資料存在，SeedFarmer 會將 ADDF 部署清單檔案及其內容與目標帳戶中現有的已部署成品的 MD5 雜湊進行比較。如果偵測到可部署的變更，則此程序會繼續。如果未偵測到任何可部署的變更，則表示此程序已完成。

5. 將範圍內的 ADDF 模組部署至目標帳戶。

根據 ADDF 部署清單檔案和上一步的評估結果，CodeSeeder 現在具有要執行的部署的排序清單。根據該排序清單，CodeSeeder 在每個關聯的目標帳戶中擔任 ADDF 部署 IAM 角色。然後，它在 AWS CodeBuild 作業中執行 CodeSeeder，以建立或更新 ADDF 模組的個別 IaC 部署，例如 AWS CDK 應用程式。依預設，ADDF 使用 AWS CDK 作為 IaC 架構，但也支援其他常見 IaC 架構。每個目標帳戶的程序完成後，您將擁有一個完全部署的、跨帳戶的端對端 ADAS 型工作流程，如您在 ADDF 部署清單檔案中定義。

如果您使用單一帳戶架構，則工具鏈帳戶和目標帳戶是相同帳戶，且同一個帳戶具有所述的所有功能。

6. 使用 ADDF 部署的基礎設施。

ADAS 開發人員可以使用已部署的 ADAS 型工作流程，如您的使用案例所定義。

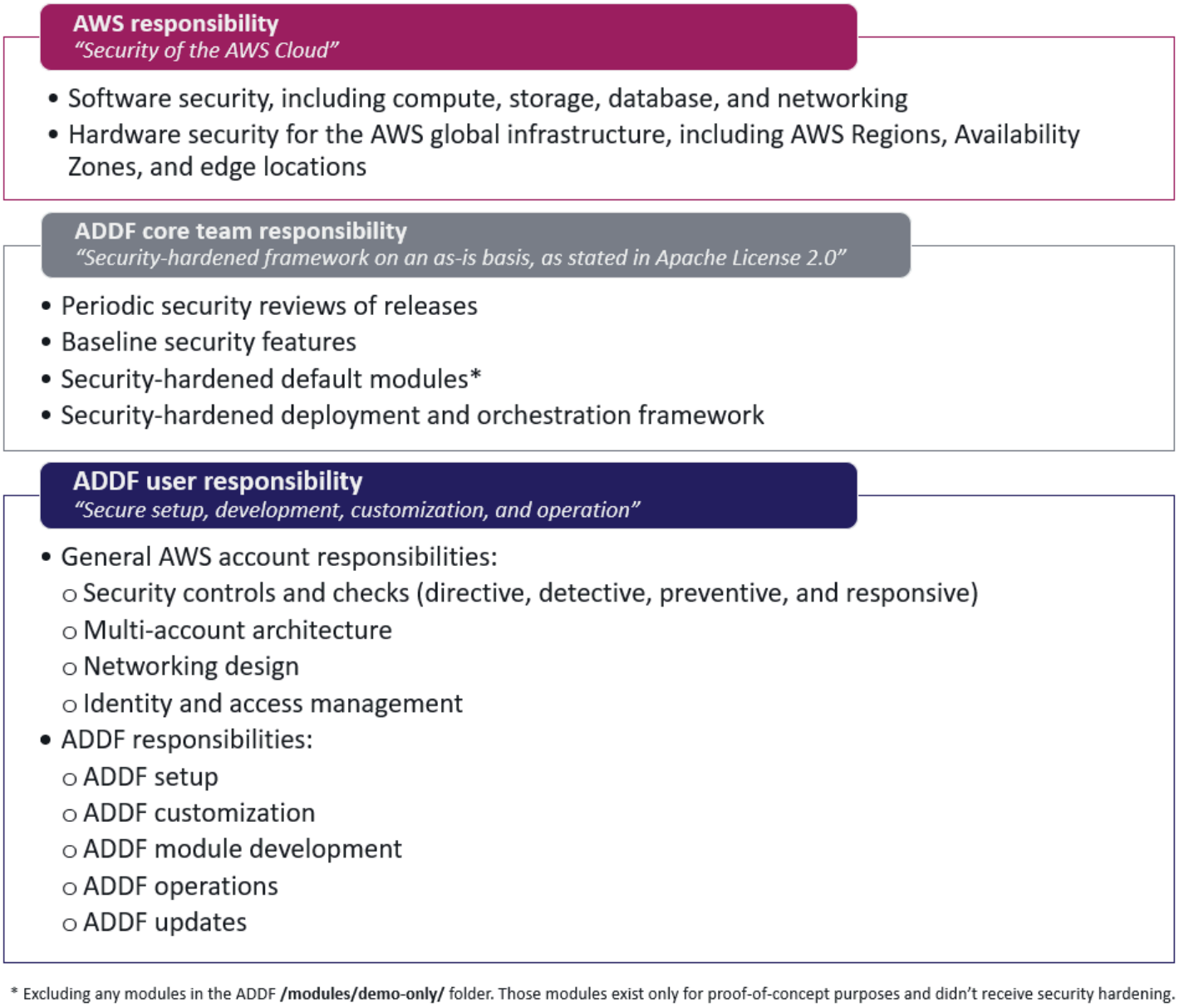
此工作流程描述 ADDF 多帳戶環境的單一執行個體的架構。根據您的開發、部署和操作模型，我們建議您在多階段環境中執行多個 ADDF 執行個體。典型設定可能包括專用 ADDF 執行個體，其中每個部署階段都具有專用 AWS 帳戶，例如用於開發、測試和生產的分支。您也可以在相同 AWS 區域的相同單一帳戶或多帳戶環境中執行多個 ADDF 執行個體，假設您為每個 ADDF 執行個體建立了唯一的資源命名空間。如需更多詳細資訊，請參閱 定義您的 ADDF 架構。

ADDF 共同責任模式

適用於 AWS 服務 的共同責任模式也適用於自動駕駛資料架構 (ADDF)。下列實體共同承擔保護 ADDF 的責任，如下圖所示：

- AWS – 提供 AWS 服務 的雲端基礎設施供應商。
- ADDF 核心團隊 – ADDF 核心團隊是在 ADDF 儲存庫 (GitHub) 中發佈 ADDF 版本的實體。
- ADDF 使用者 – ADDF 使用者包括但不限於：
  - ADDF 開發人員 – 變更、自訂或建立新 ADDF 模組程式碼的任何人。
  - ADF 操作員 – 設定和操作 ADDF 執行個體的任何人。
  - ADAS 開發人員 – ADDF 部署的資源的最終使用者或取用者。例如，ADAS 開發人員可以查詢建立為 ADDF 部署一部分的視覺化前端。

下圖總結了 AWS、ADDF 核心團隊與 ADDF 使用者之間的共同責任。



AWS 責任

AWS 負責保護執行 AWS 雲端 中提供的所有服務的基礎設施，如 AWS 共同責任模式中所定義。此基礎設施由執行 AWS 雲端 服務的硬體、軟體、聯網與設施組成。

ADDF 核心團隊責任

根據 Apache License 2.0 (GitHub)，ADDF 核心團隊盡力提供了一個本身安全的架構。ADDF 核心團隊負責下列工作：

- 發行版本的定期安全審查
- 基準安全功能
- 安全性強化的預設模組 (這不包括 /modules/demo-only/ 資料夾中的任何模組。這些模組僅用於概念驗證用途，不會接受安全性強化。)
- 安全性強化的部署和協同運作架構

這些安全責任僅擴充至 GitHub 儲存庫中提供的架構，無需任何修改或自訂。這包括除 modules/demo-only/ 資料夾中的 ADDF 模組之外的所有 ADDF 模組。此資料夾中的 ADDF 模組未經過安全性強化，不應部署在生產環境或任何具有敏感或受保護資料的環境中。包含這些模組是為了展示系統功能，您可以用其作為建立您自己的自訂、安全性強化模組的基礎。

> **Note**
>
> ADDF 作為架構依原狀交付。如 Apache License 2.0 (GitHub) 所述，它不隨附任何責任和保證。您應對 ADDF 進行自己的安全評定，並確認它符合您組織的特定安全要求。

ADDF 使用者責任

只有當 ADDF 以安全的方式設定、自訂和操作時，ADDF 及其模組才是安全的。ADDF 使用者對下列項目的安全負全部責任：

- 一般 AWS 帳戶 責任：
  - 安全控制和檢查 (指令、偵測性、預防性和回應)
  - 多帳戶架構
  - 聯網設計
  - 身分和存取管理
- ADDF 特定責任：
  - ADDF 設定
  - ADDF 自訂
  - ADDF 模組開發
  - ADDF 操作
  - ADDF 更新

一般 AWS 帳戶 責任

在將任何 ADDF 相關資源部署到 AWS 帳戶 之前，應根據 AWS Well-Architected Framework 中的最佳實務設定您的 AWS 帳戶。這包括指令、偵測性、預防性和回應安全控制。您應制定詳細的緩解程序，以防發生任何安全違規或事件。您組織的政策應包含集中管理身分、存取和聯網的要求。通常，這些要求和服務由專用登陸區域團隊處理。

ADDF 特定責任

安全的 ADDF 設定

ADDF 使用者的責任始於根據 ADDF 文件安全地設定 ADDF。我們強烈建議您遵循 ADDF 部署指南 (GitHub) 中的說明進行操作。如需有關安全地設定 ADDF 的詳細資訊，請參閱 定義您的 ADDF 架構 和 初始設定。

安全 ADF 自訂

如果對 ADDF 核心功能 (例如 CodeSeeder、SeedFarmer 和 ADDF 核心模組) 進行任何自訂，ADDF 使用者將對這些變更承擔全部責任。如需更多詳細資訊，請參閱 自訂 ADDF 部署架構程式碼。

安全 ADDF 模組開發

ADDF 使用者對使用 ADDF 部署的任何自訂模組負全部責任。此外，ADDF 使用者負責對 ADDF 提供的模組進行任何程式碼變更。如需更多詳細資訊，請參閱 在 ADDF 中編寫自訂模組。

安全 ADDF 更新和操作

隨著架構的發展，ADDF 會收到功能和安全更新。ADDF 使用者有責任定期檢查發佈至 GitHub 儲存庫的更新並長期安全地操作 ADDF。如需詳細資訊，請參閱 重複發生的 ADDF 部署、重複發生的安全稽核、ADDF 更新 和 解除委任。

ADF 安全審查程序

自動駕駛資料架構 (ADDF) 在建置時就考慮了安全性。在向公眾發行之前，AWS 對 ADDF 進行了初始內部安全審查並解決了任何已識別的安全問題。AWS 和開放原始碼社群都為架構的持續安全審查做出了貢獻。

AWS 定期進行安全審查

ADDF 在 AWS 擁有的 awslabs GitHub 組織下發佈。AWS 定期對此組織中的程式碼進行自動和手動安全審查，以竭盡全力確認安全。根據 AWS 政策，AWS 不會揭露有關安全審查頻率、方法或所使用工具的資訊。此外，AWS 不會發佈有關 ADDF 的任何內部稽核報告。但是，任何已識別的安全調查結果都透過提取請求修正和發佈，具有高度緊迫性。

> **Note**
>
> ADDF 作為架構依「原狀」交付，不附帶任何明示或默示之保證或條件，包括但不限於所有權、不侵權、適銷性或特定用途的適用性的任何保證或條件，如 Apache License 2.0 (GitHub) 所述。您應對 ADDF 進行自己的安全評定，並確認它符合您組織的特定安全要求，並根據 Apache License 2.0 中的規定，您全權負責確定使用或重新發佈 ADDF 的適當性，並承擔與您根據此類授權行使或取得許可關聯的任何風險。

開放原始碼安全審查和貢獻

ADDF 是一個歡迎貢獻的開放原始碼專案。我們邀請所有使用者對架構進行自己的安全審查，並透過報告任何與安全相關的調查結果做出貢獻。如果您發現程式碼中存在問題，請遵循安全問題通知 (ADDF 文件) 中的指導方針。

ADDF 內建安全功能

自動駕駛資料架構 (ADDF) 具有各種內建安全功能。依預設，這些功能旨在協助您設定安全架構，並協助您的組織符合常見企業安全要求。

以下是內建安全功能：

- ADDF 模組程式碼的最低權限
- 基礎設施即程式碼
- IaC 的自動安全檢查
- AWS CDK 部署角色的自訂最低權限政策
- 模組 deployspec 檔案的最低權限政策
- 資料加密
- 使用 Secrets Manager 的憑證儲存
- SeedFarmer 和 CodeSeeder 的安全審查
- CodeSeeder AWS CodeBuild 角色的許可界限支援
- AWS 多帳戶架構
- 多帳戶部署的最低權限許可

ADDF 模組程式碼的最低權限

最低權限是授予執行任務所需的最低許可的安全最佳實務。如需詳細資訊，請參閱套用最低權限許可。ADDF 提供的模組在其程式碼和部署的資源中嚴格遵循最低權限原則，如下所示：

- 為 ADDF 模組產生的所有 AWS Identity and Access Management (IAM) 政策都具有使用案例所需的最低許可。
- AWS 服務 根據最低權限原則進行設定和部署。ADDF 提供的模組僅使用特定使用案例所需的服務和服務功能。

基礎設施即程式碼

ADDF 作為架構，旨在將 ADDF 模組部署為基礎設施即程式碼 (IaC)。IaC 消除了手動部署程序，有助於防止手動程序可能導致的錯誤和組態錯誤。

ADDF 旨在使用任何常見的 IaC 架構來協調和部署模組。這包括但不限於：

- AWS Cloud Development Kit (AWS CDK)
- AWS CloudFormation
- Hashicorp Terraform

您可以使用不同的 IaC 架構編寫不同的模組，然後使用 ADDF 進行部署。

ADDF 模組使用的預設 IaC 架構是 AWS CDK。AWS CDK 是一個高層級物件導向的抽象概念，您可以使用它來以命令方式定義 AWS 資源。依預設，AWS CDK 已針對各種服務和案例強制執行安全最佳實務。透過使用 AWS CDK，降低了安全錯誤組態的風險。

IaC 的自動安全檢查

開放原始碼 cdk-nag 公用程式 (GitHub) 已整合到 ADDF。此公用程式會自動檢查基於 AWS CDK 的 ADDF 模組是否遵守一般和安全最佳實務。cdk-nag 公用程式使用規則和規則套件來偵測和報告違反最佳實務的程式碼。如需有關規則和完整清單的詳細資訊，請參閱 cdk-nag 規則 (GitHub)。

AWS CDK 部署角色的自訂最低權限政策

ADDF 廣泛使用 AWS CDK v2。您需要將所有 ADDF AWS 帳戶 引導至 AWS CDK。如需詳細資訊，請參閱引導 (AWS CDK 文件)。

依預設，AWS CDK 將寬鬆的 AdministratorAccess AWS 受管政策指派給在引導的帳戶中建立的 AWS CDK 部署角色。此角色的完整名稱為 `cdk-[CDK_QUALIFIER]-cfn-exec-role-[AWS_ACCOUNT_ID]-[REGION]`。作為 AWS CDK 部署程序的一部分，AWS CDK 使用此角色將資源部署到引導的 AWS 帳戶 中。

根據您組織的安全要求，`AdministratorAccess` 政策可能過於寬鬆。作為 AWS CDK 引導程序的的一部分，您可以根據自己的需要自訂政策和許可。您可以透過使用 `--cloudformation-execution-policies` 參數並使用新定義的政策重新引導帳戶來變更政策。如需詳細資訊，請參閱自訂引導 (AWS CDK 文件)。

> **Note**
>
> 雖然此安全功能不是 ADDF 特定的，但本節中列出了它，因為它可以提高 ADDF 部署的整體安全性。

模組 deployspec 檔案的最低權限政策

每個模組都包含稱為 deployspec.yaml 的部署規格檔案。此檔案定義了模組的部署指示。CodeSeeder 使用它來透過利用 AWS CodeBuild 將定義的模組部署在目標帳戶中。CodeSeeder 會將預設服務角色指派給 CodeBuild 以部署資源，如部署規格檔案中指示。此服務角色根據最低權限原則設計。它包括部署 AWS CDK 應用程式所需的所有許可，因為 ADDF 提供的所有模組都是作為 AWS CDK 應用程式建立的。

但是，如果您需要在 AWS CDK 之外執行任何階段命令，則需要建立自訂 IAM 政策，而不是使用 CodeBuild 的預設服務角色。例如，如果您使用的是 AWS CDK 以外的 IaC 部署架構 (例如 Terraform)，則需要建立 IAM 政策，以授予足夠的許可使該特定架構正常運作。需要專用 IAM 政策的另一種案例是當您在 `install`、`pre_build`、`build` 或 `post_build` 階段命令中包含對其他 AWS 服務 的直接 AWS Command Line Interface (AWS CLI) 呼叫時。例如，如果您的模組包含用於將檔案上傳至 S3 儲存貯體的 Amazon Simple Storage Service (Amazon S3) 命令，則您需要自訂政策。自訂 IAM 政策為 AWS CDK 部署以外的任何 AWS 命令提供精細控制。如需自訂 IAM 政策範例，請參閱 ModuleStack (SeedFarmer 文件)。為 ADDF 模組建立自訂 IAM 政策時，確保套用最低權限許可。

資料加密

ADDF 會儲存和處理潛在的敏感資料。為了協助保護此資料，SeedFarmer、CodeSeeder 和 ADDF 提供的模組會對所有使用的 AWS 服務 的靜態資料和傳輸中的資料進行加密 (除非對 `demo-only` 資料夾中的模組另有明確說明)。

使用 Secrets Manager 的憑證儲存

ADDF 會處理不同服務的各種機密，例如 Docker Hub、JupyterHub 和 Amazon Redshift。ADDF 使用 AWS Secrets Manager 儲存任何與 ADF 相關的機密。這有助於您透過原始程式碼移除敏感資料。

Secrets Manager 機密僅儲存在目標帳戶中，以便該帳戶正常運作。依預設，工具鏈帳戶不包含任何機密。

SeedFarmer 和 CodeSeeder 的安全審查

SeedFarmer 和 CodeSeeder (GitHub 儲存庫) 用於部署 ADDF 及其 ADDF 模組。這些開放原始碼專案經歷與 ADDF 相同的定期 AWS 內部安全審查程序，如 ADF 安全審查程序 所述。

CodeSeeder AWS CodeBuild 角色的許可界限支援

IAM 許可界限界是一種常見安全機制，用於定義身分型政策可以授予 IAM 實體的許可上限。SeedFarmer 和 CodeSeeder 支援每個目標帳戶的 IAM 許可界限連接。許可界限限制 CodeSeeder 部署模組時 CodeBuild 使用的任何服務角色的許可上限。IAM 許可界限必須由安全團隊在 ADDF 外部建立。IAM 許可界限政策連接被接受為 ADDF 部署清單檔案 deployment.yaml 內的屬性。如需詳細資訊，請參閱許可界限支援 (SeedFarmer 文件)。

工作流程如下所示：

1. 您的安全團隊根據您的安全要求定義並建立 IAM 許可界限。必須在每個 ADDF AWS 帳戶 中個別建立 IAM 許可界限。輸出是許可界限政策 Amazon Resource Name (ARN) 清單。
2. 安全團隊會與您的 ADDF 開發人員團隊共用政策 ARN 清單。
3. ADDF 開發人員團隊會將政策 ARN 清單整合到清單檔案中。如需此整合的範例，請參閱 sample-permissionboundary.yaml (GitHub) 和部署清單檔案 (SeedFarmer 文件)。
4. 成功部署後，許可界限將附接至 CodeBuild 用於部署模組的所有服務角色。
5. 安全團隊監控是否視需要已套用許可界限。

AWS 多帳戶架構

如 AWS Well-Architected Framework 的安全支柱中定義，根據您組織的要求將資源和工作負載分成多個 AWS 帳戶 被認為是最佳實務。這是因為 AWS 帳戶 充當隔離界限。如需詳細資訊，請參閱 AWS 帳戶 管理和區隔。此概念的實作稱為多帳戶架構。與單一帳戶架構相比，正確設計的 AWS 多帳戶架構可提供工作負載分類，並減少發生安全漏洞時的影響範圍。

ADDF 以原生方式支援 AWS 多帳戶架構。您可以視需要將您的 ADDF 模組分佈在盡可能多的 AWS 帳戶 上，以符合您組織的安全和職責分離要求。您可以將 ADDF 部署到單一 AWS 帳戶 中，結合工具鏈和目標帳戶功能。或者，您可以為 ADDF 模組或模組群組建立個別目標帳戶。

您需要考慮的唯一限制是 ADDF 模組代表每個 AWS 帳戶 的最小部署單位。

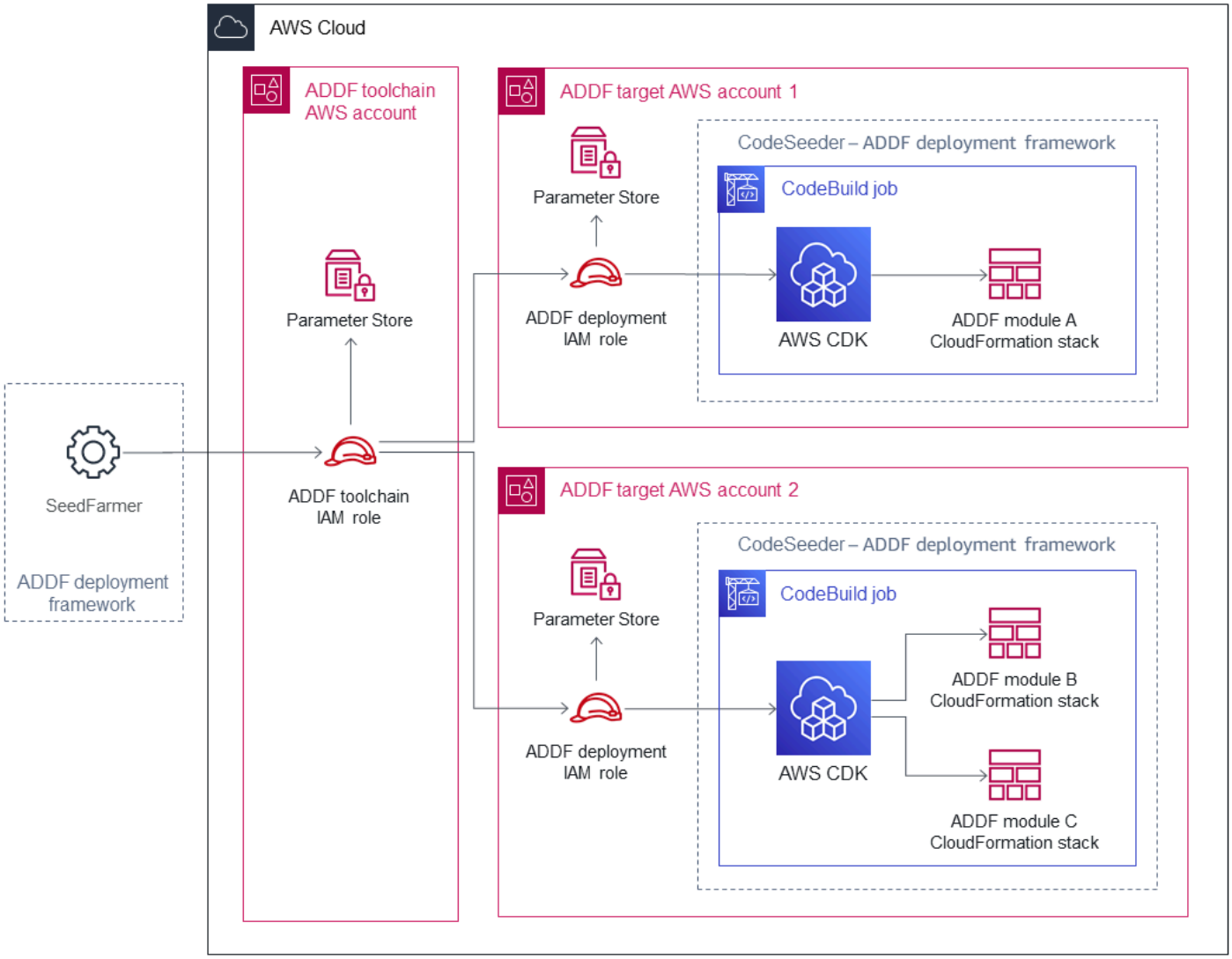
對於生產環境，建議您使用由一個工具鏈帳戶和至少一個目標帳戶組成的多帳戶架構。如需更多詳細資訊，請參閱 ADDF 架構。

多帳戶部署的最低權限許可

如果您使用多帳戶架構，SeedFarmer 需要存取目標帳戶來執行下列三個動作：

1. 將 ADDF 模組中繼資料寫入工具鏈帳戶和目標帳戶。
2. 從工具鏈帳戶和目標帳戶讀取目前 ADDF 模組中繼資料。
3. 在目標帳戶中啟動 AWS CodeBuild 作業，以部署或更新模組。

下圖顯示跨帳戶關係，包括擔任 ADDF 特定 AWS Identity and Access Management (IAM) 角色的操作。



這些跨帳戶操作透過使用明確定義的 assume-role 操作來實現。

- ADDF 工具鏈 IAM 角色部署在工具鏈帳戶中。SeedFarmer 擔任此角色。此角色有權執行 `iam:AssumeRole` 動作，且可以在每個目標帳戶中擔任 ADDF 部署 IAM 角色。此外，ADDF 工具鏈 IAM 角色可以執行本機 AWS Systems Manager Parameter Store 操作。

- ADDF 部署 IAM 角色會部署在每個目標帳戶中。只能透過使用 ADDF 工具鏈 IAM 角色從工具鏈帳戶擔任此角色。此角色有權執行本機 AWS Systems Manager Parameter Store 操作，且有權執行透過 CodeSeeder 啟動和描述 CodeBuild 作業的 AWS CodeBuild 動作。

這些 ADDF 特定的 IAM 角色是作為 ADDF 引導程序的一部分建立的。如需詳細資訊，請參閱 ADDF 部署指南 (GitHub) 中的引導 AWS 帳戶。

所有跨帳戶許可都根據最低權限原則設定。如果一個目標帳戶遭入侵，對其他 ADDF AWS 帳戶 的影響很小或沒有影響。

若是 ADDF 的單一帳戶架構，角色關係保持不變。其只是折疊成單一 AWS 帳戶。

ADDF 安全設定和操作

自動駕駛資料架構 (ADDF) 應視為自訂軟體，需要組織中的專用 DevOps 和安全團隊進行持續維護和保養。本節說明可協助您在 ADDF 整個生命週期中設定和操作的與安全相關的常見任務。

本節包含下列任務：

- 定義您的 ADDF 架構
- 初始設定
- 自訂 ADDF 部署架構程式碼
- 在 ADDF 中編寫自訂模組
- 重複發生的 ADDF 部署
- 重複發生的安全稽核
- ADDF 更新
- 解除委任

定義您的 ADDF 架構

ADDF 執行個體的安全性僅與其所部署的 AWS 帳戶 環境一樣。此 AWS 帳戶 環境的設計必須符合特定使用案例的安全和操作需求。例如，在概念驗證 (PoC) 環境中設定 ADDF 執行個體的安全和操作相關任務及考量與在生產環境中設定 ADDF 的任務及考量事項不同。

在 PoC 環境中執行 ADDF

如果您打算在 PoC 環境中使用 ADDF，我們建議您為 ADDF 建立一個不包含任何其他工作負載的專用 AWS 帳戶。這有助於在您探索 ADDF 及其功能時確保您的帳戶安全。以下是此方法的優點：

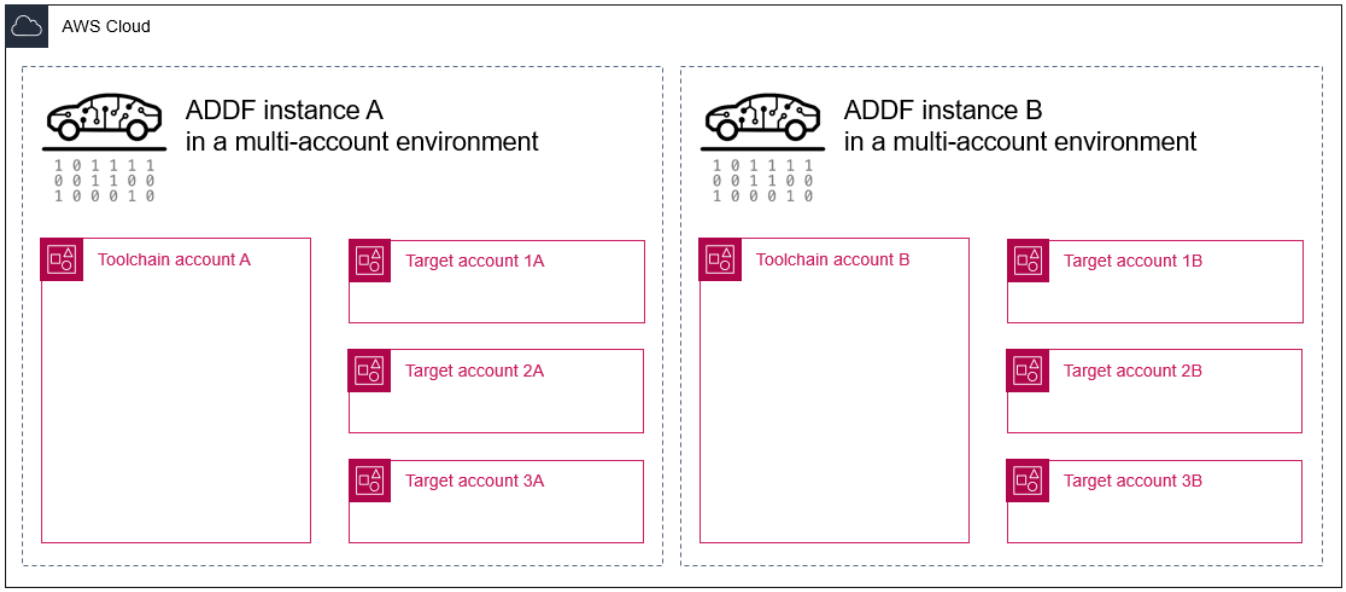
- 如果發生嚴重的 ADDF 組態錯誤，則不會對其他工作負載造成不利影響。
- 不存在可能對 ADDF 設定造成不利影響的任何其他工作負載組態錯誤的風險。

即使對於 PoC 環境，我們仍然建議您盡可能遵循 在生產環境中執行 ADDF 中列出的最佳實務。

在生產環境中執行 ADDF

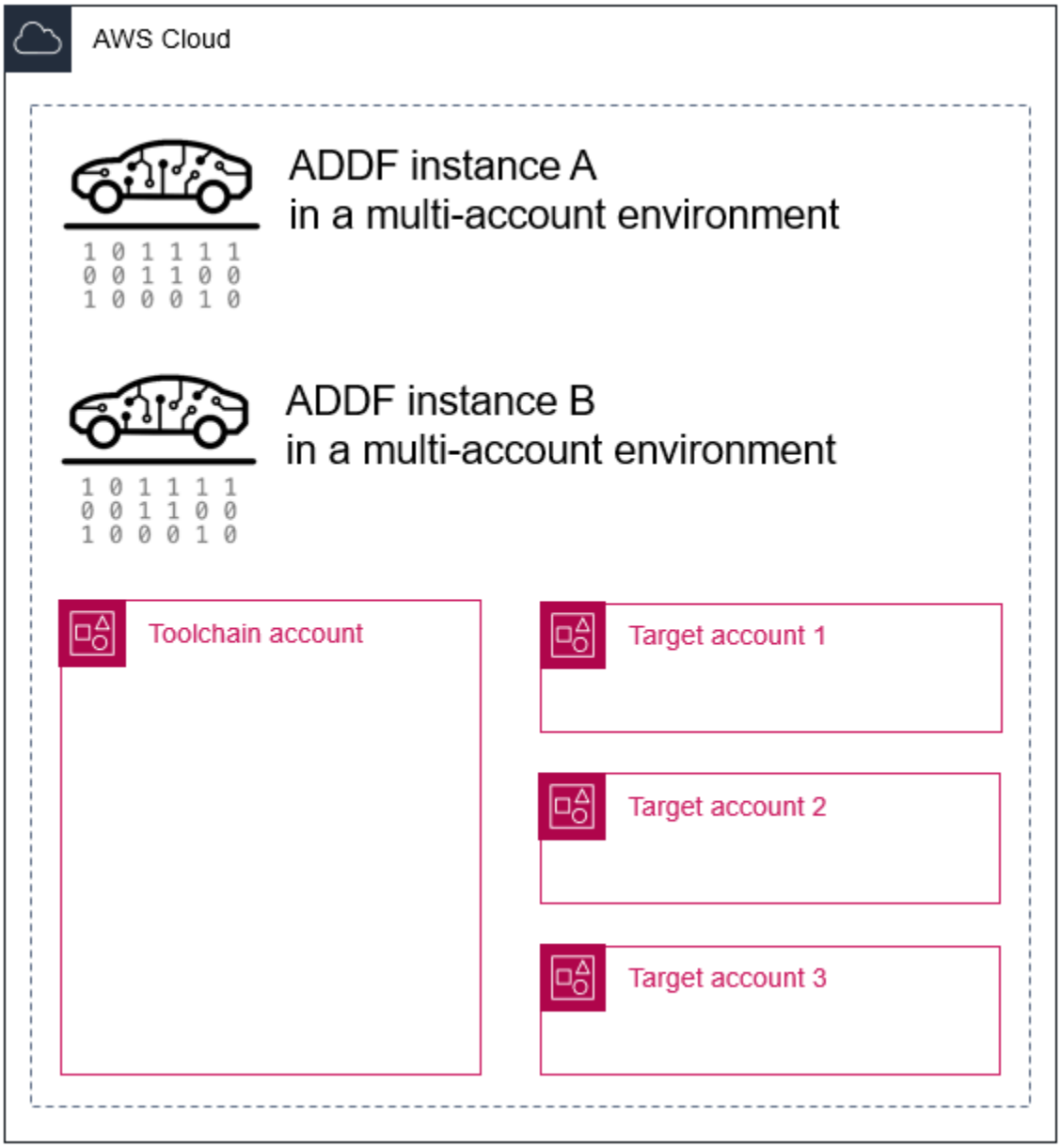
如果您打算在企業生產環境中使用 ADDF，我們強烈建議考慮您組織的安全最佳實務並相應地實作 ADDF。除了您組織的安全最佳實務之外，我們建議您實作下列項目：

- 建立一個長期受認可的 ADDF DevOps 團隊 – ADDF 需要被視為自訂軟體。它需要由專用 DevOps 團隊進行持續的維護和保養。開始在生產環境中執行 ADDF 之前，應定義一個具有足夠規模和能力的 DevOps 團隊，並承諾提供完整的資源，直到 ADDF 部署的生命週期結束。
- 使用多帳戶架構 – 每個 ADDF 執行個體都應部署在自己的專用 AWS 多帳戶環境中，沒有任何其他不相關的工作負載。如 AWS 帳戶管理和區隔 (AWS Well-Architected Framework) 中定義，根據您組織的要求將資源和工作負載分成多個 AWS 帳戶 被認為是最佳實務。這是因為 AWS 帳戶 充當隔離界限。與單一帳戶架構相比，正確設計的 AWS 多帳戶架構可提供工作負載分類，並減少發生安全漏洞時的影響範圍。使用多帳戶架構還可協助您的帳戶保持在 AWS 服務 配額內。視需要將您的 ADDF 模組分佈在盡可能多的 AWS 帳戶 上，以符合您組織的安全和職責分離要求。
- 部署多個 ADDF 執行個體 – 視需要設定任意多個獨立的 ADDF 執行個體，以根據您組織的軟體開發程序正確開發、測試和部署 ADDF 模組。設定多個 ADDF 執行個體時，您可以使用下列其中一種方法：
  - 不同的 AWS 多帳戶環境中的多個 ADDF 執行個體 – 您可以使用獨立的 AWS 帳戶 來隔離不同的 ADDF 執行個體。例如，如果您的組織具有專用的開發、測試和生產階段，您可以為每個階段建立獨立的 ADDF 執行個體和專用帳戶。這提供了許多好處，例如降低跨階段傳播任何錯誤的風險、協助您實作核准程序以及限制使用者僅存取特定環境。下列影像顯示部署在個別多帳戶環境中的兩個 ADDF 執行個體。



  - 相同的 AWS 多帳戶環境中的多個 ADDF 執行個體 – 您可以建立共用相同的 AWS 多帳戶環境的多個 ADDF 執行個體。這有效地在相同 AWS 帳戶 中建立了獨立的分支。例如，如果不同的開發人員平行工作，開發人員可以在相同的 AWS 帳戶 中建立專用 ADDF 執行個體。這有助於開發人員在獨立的分支中工作，以進行開發和測試。如果您使用此方法，對於每個 ADDF 執行個體，您的 ADDF 資源必須具有唯一的資源名稱。依預設，ADDF 預先提供的模組支援此功能。只要不超過

AWS 服務 配額，您就可以使用此方法。下列影像顯示部署在共用多帳戶環境中的兩個 ADDF 執行個體。



- 相同 AWS 單一帳戶環境中的多個 ADDF 執行個體 - 此架構與上一個範例非常相似。不同之處在於，多個 ADDF 執行個體部署在單一帳戶環境中，而非多帳戶環境中。此架構可以適合非常簡單的 ADDF 使用案例，這些使用案例的範圍非常有限，且多個開發人員同時在不同的分支上工作。

由於 SeedFarmer 是控制 ADDF 執行個體部署的單一工具，因此您可以建置適合您組織的部署策略和 CI/CD 程序的任何環境和帳戶架構。

- 根據您組織的安全要求自訂 AWS Cloud Development Kit (AWS CDK) 引導程序 – 依預設，AWS CDK 在引導過程中指派 AdministratorAccess AWS 受管政策。此政策授予完整管理權限。如果此政策對於您組織的安全要求過於寬鬆，您可以自訂套用的政策。如需更多詳細資訊，請參閱 AWS CDK 部署角色的自訂最低權限政策。
- 在 IAM 中設定存取權時遵循最佳實務 – 建立結構化 AWS Identity and Access Management (IAM) 存取解決方案，讓您的使用者可以存取 ADDF AWS 帳戶。ADDF 架構的設計遵循最低權限原則。您的 IAM 存取模式也應遵循最低權限原則，應符合您組織的要求，並應遵循 IAM 中的安全最佳實務 (IAM 文件)。
- 根據您組織的最佳實務設定聯網 – ADDF 包括選用 聯網 AWS CloudFormation 堆疊，用於建立基本的公有或私有虛擬私有雲端 (VPC)。視您組織的組態而定，此 VPC 可能會直接向網際網路公開資源。我們建議您遵循組織的聯絡最佳實務，並建立自訂安全性強化的網路模組。
- 在 AWS 帳戶 層級部署安全防護、偵測和緩解措施 – AWS 提供各種安全服務，例如 Amazon GuardDuty、AWS Security Hub、Amazon Detective 和 AWS Config。在 ADDF AWS 帳戶 中啟用

這些服務，並整合您組織的安全防護、偵測、緩解和事件處理程序。我們建議您遵循安全、身分與合規的最佳實務 (AWS 架構中心) 以及該服務文件中包含的任何服務特定的建議。如需詳細資訊，請參閱 AWS 安全文件。

ADDF 不處理其中任何主題，因為實作和組態詳細資訊在很大程度上取決於組織特定的要求和程序。相反，處理這些主題是您的組織的核心責任。通常，管理您的 AWS 登陸區域的團隊可協助您規劃和實作 ADDF 環境。

初始設定

根據 ADDF 部署指南 (GitHub) 設定 ADDF。任何部署的起點是 autonomous-driving-data-framework Git Hub 儲存庫中的 `/manifest` 資料夾。`/manifest/example-dev` 資料夾包含用於示範用途的範例部署。使用此範例作為設計您自己的部署的起點。在該目錄中，有一個稱為 deployment.yaml 的 ADDF 部署清單檔案。它包含 SeedFarmer 管理、部署或刪除 ADDF 及其在 AWS 雲端 中的資源的所有資訊。您可以在專用檔案中建立 ADDF 模組的群組。core-modules.yaml 是核心模組群組的範例，它包含 ADDF 提供的所有核心模組。總而言之，deployment.yaml 檔案包含將部署至其目標帳戶的群組和模組的所有參考，並指定部署順序。

為了獲得安全且合規的組態，特別是在不用於概念驗證的環境中，我們建議您檢閱要部署的每個模組的原始程式碼。根據安全性強化的最佳實務，您應僅部署預期使用案例所需的模組。

> **Note**
>
> `modules/demo-only/` 資料夾中的 ADDF 模組未經過安全性強化，不應部署在生產環境或任何具有敏感或受保護資料的環境中。包含這些模組是為了展示系統功能，您可以用其作為建立您自己的自訂、安全性強化模組的基礎。

自訂 ADDF 部署架構程式碼

ADDF 部署架構及其協同運作和部署邏輯可完全自訂，以符合任何需求。但是，我們建議您不要進行自訂或盡量減少變更，原因如下：

- 保持上游相容性 – 上游相容性可讓您更輕鬆地更新 ADDF 以取得最新功能和安全更新。變更架構會破壞與 SeedFarmer、CodeSeeder 和任何 ADDF 核心模組的原生向後相容性。
- 安全後果 – 變更 ADDF 部署架構可能是一項複雜任務，可能會產生非預期的安全後果。在最壞的情況下，架構變更可能會造成安全漏洞。

如果可能，建置和自訂您自己的模組程式碼，而不是修改 ADDF 部署架構和 ADDF 核心模組程式碼。

> **Note**
>
> 如果您認為 ADDF 部署架構的特定部分需要改進或進一步強化安全性，請透過提取請求將您的變更貢獻給 ADDF 儲存庫。如需更多詳細資訊，請參閱 開放原始碼安全審查和貢獻。

在 ADDF 中編寫自訂模組

建立新的 ADDF 模組或擴充現有的模組是 ADDF 的核心概念。在建立或自訂模組時，我們建議您遵循一般 AWS 安全最佳實務和您組織的安全編碼最佳實務。此外，我們建議您根據您組織的安全要求進行初始和定期的內部或外部技術安全審查，以進一步降低安全問題的風險。

重複發生的 ADDF 部署

部署 ADDF 及其模組，如 ADDF 部署指南 (GitHub) 所述。為了支援可在目標帳戶中新增、更新或移除資源的重複發生的 ADDF 部署，SeedFarmer 使用儲存在工具鍊和目標帳戶的 Parameter Store 中的 MD5 雜湊，將目前部署的基礎設施與本機程式碼庫的清單檔案中定義的基礎設施進行比較。

此方法遵循 GitOps 範例，其中您的來源儲存庫 (您操作 SeedFarmer 的本機程式碼庫) 是事實來源，其中明確聲明的基礎設施是您的部署的所需結果。如需有關 GitOps 的詳細資訊，請參閱什麼是 GitOps (GitLab 網站)。

重複發生的安全稽核

就像您組織中的任何其他軟體一樣，將 ADDF 和您的自訂 ADDF 模組程式碼整合到安全風險管理、安全審查和安全稽核週期中。

ADDF 更新

ADDF 會定期接收更新，作為持續開發工作的一部分。這包括功能更新以及與安全相關的改進和修正。我們建議您定期檢查是否有新的架構版本，並及時套用更新。如需詳細資訊，請參閱更新 ADDF 的步驟 (ADDF 文件)。

解除委任

如果不再需要 ADDF，請從您的 AWS 帳戶 中刪除 ADDF 及其所有相關資源。任何無人值守和未使用的基礎設施都會產生不必要的成本並帶來潛在的安全風險。如需詳細資訊，請參閱銷毀 ADDF 的步驟 (ADDF 文件)。

後續步驟

本指南回顧了在 AWS 雲端 環境中部署自動駕駛資料架構 (ADDF) 時的安全和操作最佳實務和考量事項。本指南回顧 ADDF 使用者、ADDF 核心團隊和 AWS 之間的共同責任模式，以便您了解安全設定和操作 ADDF 的角色和責任。它還包括在整個生命週期中安全操作 ADDF 的建議，包括環境特定的建議。

我們建議您熟悉 資源 一節中的資源。準備好後，您可以根據 ADDF 部署指南 (GitHub) 中的說明設定 ADDF。

在設定和操作 ADDF 時，如果您認為部署架構需要改進或進一步強化安全性，請透過提取請求將您的變更貢獻給 ADDF 儲存庫。如需更多詳細資訊，請參閱 開放原始碼安全審查和貢獻。

資源

AWS 文件

- 使用 ADDF on AWS 開發和部署自訂工作流程 (AWS 部落格文章)
- AWS 安全服務文件
- IAM 中的安全最佳實務
- AWS 帳戶管理和區隔
- AWS CDK 的引導
- AWS 共同責任模式
- AWS Well-Architected Framework

開放原始碼資源

- ADDF 儲存庫 (GitHub)
- ADDF 部署指南 (GitHub)
- CodeSeeder 儲存庫 (GitHub)
- SeedFarmer 儲存庫 (GitHub)

注意

客戶有責任對本文件中的資訊進行自己的獨立評定。本文件：(a) 僅供參考之用；(b) 代表目前 AWS 產品供應與實務，如有變更恕不另行通知；以及 (c) 不構成 AWS 及其附屬公司、供應商或授權人的任何承諾或保證。AWS 產品或服務均以「原樣」提供，不作任何形式的明示或暗示的保證、陳述或條件。

AWS 對其客戶的責任與義務應由 AWS 協議管轄，本文並非 AWS 與其客戶之間的任何協議的一部分，也並非上述協議的修改。

文件歷史記錄

下表描述了本指南的重大變更。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	—	2022 年 11 月 15 日

AWS 規定指引詞彙

以下是 AWS 規範性指引所提供的策略、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 資料庫遷移到與 Amazon Aurora PostgreSQL 相容的版本。
- 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：將您的現場部署 Oracle 資料庫遷移到 Amazon Relational Database Service 服務 (Amazon RDS)，適用於 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 資料庫 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應用程式移轉至 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

ABAC

請參閱以屬性為基礎的存取控制。

抽象的服務

請參閱受管理服務。

酸

請參閱原子性、一致性、隔離性、耐用性。

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。它比主動-被動遷移更具彈性，但需要更多的工作。

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，但只有來源資料庫處理來自連接應用程式的交易，同時將資料複寫至目標資料庫。目標資料庫在遷移期間不接受任何交易。

聚合函數

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

AI

請參閱人工智慧。

艾奧運

請參閱人工智慧作業。

匿名化

永久刪除資料集中個人資訊的程序。匿名化可以幫助保護個人隱私。匿名資料不再被視為個人資料。

反模式

一種經常使用的解決方案，用於解決方案的生產力適得其反，效果不佳或效果低於替代方案。

應用控制

一種安全性方法，只允許使用核准的應用程式，以協助保護系統免受惡意軟體的攻擊。

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 AWS中的 ABAC。

授權資料來源

儲存資料主要版本的位置，被認為是最可靠的資訊來源。您可以將授權資料來源中的資料複製到其他位置，以便處理或修改資料，例如匿名化、編輯或將其虛擬化。

可用區域

一個獨立的位置，與其他 AWS 區域 可用區域中的故障隔離，並為相同區域中的其他可用區域提供廉價、低延遲的網路連線能力。

AWS 雲端採用架構 (AWS CAF)

指導方針和最佳做法的架構，可協 AWS 助組織制定有效率且有效的計畫，以順利移轉至雲端。AWS CAF 將指導組織到六個重點領域，稱為觀點：業務，人員，治理，平台，安全性和運營。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。針對此觀點，AWS CAF 為人員開發、訓練和通訊提供指導，以協助組織為成功採用雲端做好準備。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)

可評估資料庫移轉工作負載、建議移轉策略並提供工作預估的工具。 AWS WQF 包含在 AWS Schema Conversion Tool ()AWS SCT中。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

壞機器人

旨在破壞或對個人或組織造成傷害的機器人。

BCP

請參閱業務連續性規劃。

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 「位元順序」。

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

建立兩個獨立但相同環境的部署策略。您可以在一個環境中執行目前的應用程式版本 (藍色)，而在另一個環境 (綠色) 中執行新的應用程式版本。此策略可協助您以最小的影響快速回復。

機器人

透過網際網路執行自動化工作並模擬人類活動或互動的軟體應用程式。某些漫遊器是有用的或有益的，例如用於索引 Internet 上信息的網絡爬蟲。其他一些機器人（稱為不良機器人）旨在破壞或對個人或組織造成傷害。

殭屍網絡

受惡意軟件感染並受到單一方（稱為機器人牧民或機器人操作員）控制的機器人網絡。殭屍網絡是擴展機器人及其影響的最著名機制。

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)。

防碎玻璃訪問

在特殊情況下，並透過核准的程序，使用者可以快速取得他 AWS 帳戶 們通常沒有存取權限的存取權。如需詳細資訊，請參閱 AWS Well-Architected 指南中的實作防破玻璃程序指標。

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

咖啡

請參閱AWS 雲端採用架構。

金絲雀部署

向最終用戶發行版本的緩慢和增量版本。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

請參閱雲端卓越中心。

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來測試系統的彈性。您可以使用 AWS Fault Injection Service (AWS FIS) 執行實驗來 stress 您的 AWS 工作負載並評估其回應。

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。

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。雲計算通常連接到邊緣計算技術。

雲端運作模式

在 IT 組織中，這是用來建置、成熟和最佳化一或多個雲端環境的作業模型。如需詳細資訊，請參閱建立您的雲端作業模型。

採用雲端階段

組織移轉至下列四個階段時通常會經歷 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

這些階段是 Stephen Orban 在 AWS 雲端 企業策略部落格部落格文章「邁向雲端優先的旅程與採用階段」中所定義的。如需其與 AWS 移轉策略之間關聯的詳細資訊，請參閱移轉準備指南。

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。常見的雲存儲庫包括 GitHub 或 AWS CodeCommit。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。查詢此類資料時，通常可以接受緩慢的查詢。將此資料移至效能較低且成本較低的儲存層或類別可降低成本。

計算機視覺（CV）

一個 AI 領域，它使用機器學習來分析和從數字圖像和視頻等視覺格式中提取信息。例如，提 AWS Panorama 供將 CV 添加到現場部署攝像機網絡的設備，Amazon 為 CV SageMaker 提供圖像處理算法。

配置漂移

對於工作負載，組態會從預期的狀態變更。這可能會導致工作負載變得不合規，而且通常是漸進且無意的。

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

AWS Config 規則和補救動作的集合，您可以組合這些動作來自訂合規性和安全性檢查。您可以使用 YAML 範本，將一致性套件部署為 AWS 帳戶 和區域中的單一實體，或跨組織部署。如需詳細資訊，請參閱文件中的AWS Config 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 架構中安全性支柱的一個組成部分。如需詳細資訊，請參閱資料分類。

資料漂移

生產資料與用來訓練 ML 模型的資料之間有意義的變化，或輸入資料隨著時間的推移有意義的變化。資料漂移可降低機器學習模型預測中的整體品質、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

透過集中式管理和控管，提供分散式、分散式資料擁有權的架構架構。

資料最小化

僅收集和處理絕對必要的數據的原則。在中執行資料最小化 AWS 雲端 可降低隱私權風險、成本和分析碳足跡。

資料周長

您 AWS 環境中的一組預防性護欄，可協助確保只有受信任的身分正在存取來自預期網路的受信任資源。若要取得更多資訊，請參閱〈在上建立資料周長〉 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。

數據來源

在整個生命週期中追蹤資料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 (例如分析) 的資料管理系統。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。

defense-in-depth

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。在上採用此策略時 AWS，您可以在 AWS

Organizations 結構的不同層加入多個控制項，以協助保護資源。例如，一 defense-in-depth 種方法可能會結合多因素驗證、網路分段和加密。

委派的管理員

在中 AWS Organizations，相容的服務可以註冊成 AWS 員帳戶，以管理組織的帳戶並管理該服務的權限。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。

部署

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。

發展價值流映射

用於識別限制並排定優先順序，對軟體開發生命週期中的速度和品質產生不利影響的程序。DVSM 擴展了最初為精益生產實踐而設計的價值流映射流程。它著重於創造和通過軟件開發過程中移動價值所需的步驟和團隊。

數字雙胞胎

真實世界系統的虛擬表現法，例如建築物、工廠、工業設備或生產線。數位雙胞胎支援預測性維護、遠端監控和生產最佳化。

維度表

在 star 結構描述中，較小的資料表包含事實資料表中定量資料的相關資料屬性。維度表格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、篩選和結果集標籤。

災難

防止工作負載或系統在其主要部署位置達成其業務目標的事件。這些事件可能是自然災害、技術故障或人為行為造成的結果，例如意外設定錯誤或惡意軟體攻擊。

災難復原 (DR)

您使用的策略和程序，將因災難造成的停機時間和資料遺失降到最低。如需詳細資訊，請參閱 AWS Well-Architected 的架構中的雲端中的工作負載的災難復原 AWS：雲端復原。

DML

請參閱資料庫操作語言。

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。

博士

請參閱災難復原。

漂移檢測

追蹤基線組態的偏差。例如，您可以用 AWS CloudFormation 來偵測系統資源中的漂移，也可以用 AWS Control Tower 來偵測 landing zone 中可能會影響法規遵循治理要求的變更。

DVSM

請參閱開發價值流映射。

E

EDA

請參閱探索性資料分析。

邊緣運算

提升 IoT 網路邊緣智慧型裝置運算能力的技術。與雲計算相比，邊緣計算可以減少通信延遲並縮短響應時間。

加密

一種計算過程，將純文本數據（這是人類可讀的）轉換為密文。

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。您可以使用其他或 (IAM) 主體建立端點服務， AWS PrivateLink 並將權限授予其他 AWS 帳戶 或 AWS Identity and Access Management (IAM) 主體。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

可自動化並管理企業關鍵業務流程 (例如會計、MES 和專案管理) 的系統。

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 安全史詩包括身份和訪問管理，偵探控制，基礎結構安全性，數據保護和事件響應。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

事實表

星型架構中的中央表格。它存儲有關業務運營的定量數據。事實資料表通常包含兩種類型的資料欄：包含計量的資料欄，以及包含維度表格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減少開發生命週期的理念。這是敏捷方法的關鍵部分。

故障隔離邊界

在中 AWS 雲端，可用區域、 AWS 區域控制平面或資料平面等界限，可限制故障的影響，並協助改善工作負載的彈性。如需詳細資訊，請參閱AWS 錯誤隔離邊界。

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，請參閱機器學習模型可解釋性：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。

FGAC

請參閱精細的存取控制。

精細的存取控制 (FGAC)

使用多個條件來允許或拒絕訪問請求。

閃切遷移

一種資料庫移轉方法，透過變更資料擷取使用連續資料複寫，在最短的時間內移轉資料，而不是使用階段化方法。目標是將停機時間降至最低。

G

地理阻塞

請參閱地理限制。

地理限制 (地理封鎖)

在 Amazon 中 CloudFront，防止特定國家/地區的使用者存取內容分發的選項。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 工作流程被認為是遺留的，基於主幹的工作流程是現代的首選方法。

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。它們是通過使用 AWS Config，Amazon AWS Security Hub GuardDuty， AWS Trusted Advisor亞馬遜檢查 Amazon Inspector 和自定義 AWS Lambda 檢查來實現的。

H

公頃

查看高可用性。

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，無需干預。HA 系統的設計可自動容錯移轉、持續提供高品質的效能，以及處理不同的負載和故障，並將效能影響降到最低。

歷史學家現代化

一種用於現代化和升級操作技術 (OT) 系統的方法，以更好地滿足製造業的需求。歷史學家是一種類型的數據庫，用於收集和存儲工廠中的各種來源的數據。

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。

熱數據

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的儲存層或類別，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，修補程式通常是在典型的 DevOps 發行工作流程之外進行。

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

查看基礎結構即程式碼。

身分型政策

附加至一或多個 IAM 主體的政策，用於定義其在 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。

不可變基礎設施

為生產工作負載部署新基礎結構的模型，而不是更新、修補或修改現有基礎結構。不可變的基礎架構本質上比可變基礎架構更加一致、可靠且可預測。如需詳細資訊，請參閱 Well-Architected 的架構中的使用不可變基礎結 AWS 構進行部署最佳作法。

傳入 (輸入) VPC

在 AWS 多帳戶架構中，VPC 可接受、檢查和路由來自應用程式外部的網路連線。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

Klaus Schwab 於 2016 年推出的一個術語，指的是透過連線能力、即時資料、自動化、分析和 AI/ML 的進步來實現製造流程的現代化。

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 VPC（相同或不同 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 AWS 的機器學習模型可解釋性。

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

標籤式存取控制 (LBAC)

強制存取控制 (MAC) 的實作，其中每個使用者和資料本身都明確指派一個安全性標籤值。使用者安全性標籤與資料安全性標籤之間的交集決定了使用者可以看到哪些列與欄。

登陸區域

landing zone 是一個架構良好的多帳戶 AWS 環境，具有可擴展性和安全性。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。

大型遷移

遷移 300 部或更多伺服器。

LBAC

請參閱以標示為基礎的存取控制。

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

見 7 盧比

小端序系統

首先儲存最低有效位元組的系統。另請參閱 「位元順序」。

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

旨在危及計算機安全性或隱私的軟件。惡意軟件可能會破壞計算機系統，洩漏敏感信息或獲得未經授權的訪問。惡意軟體的例子包括病毒、蠕蟲、勒索軟體、特洛伊木馬程式、間諜軟體和鍵盤記錄程式。

受管理服務

AWS 服務 用於 AWS 操作基礎架構層、作業系統和平台，並且您可以存取端點以儲存和擷取資料。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

用於跟踪，監控，記錄和控制生產過程的軟件系統，可在現場將原材料轉換為成品。

MAP

請參閱 Migration Acceleration Program。

機制

一個完整的過程，您可以在其中創建工具，推動工具的採用，然後檢查結果以進行調整。機制是一個循環，它加強和改善自己，因為它運行。如需詳細資訊，請參閱 AWS Well-Architected 的架構中建置機制。

成員帳戶

屬於 AWS 帳戶 中組織的管理帳戶以外的所有帳戶 AWS Organizations。一個帳戶一次只能是一個組織的成員。

MES

請參閱製造執行系統。

郵件佇列遙測傳輸 (MQTT)

以發佈/訂閱模式為基礎的輕量型 machine-to-machine (M2M) 通訊協定，適用於資源受限 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 AWS 無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，請參閱上 AWS的實作微服務。

Migration Acceleration Program (MAP)

提供諮詢支援、訓練和服務的 AWS 計畫，協助組織為移轉至雲端建立穩固的營運基礎，並協助抵消移轉的初始成本。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。移轉工廠團隊通常包括營運、業務分析師和擁有者、移轉工程師、開發人員和 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。移轉中繼資料的範例包括目標子網路、安全性群組和 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 AWS 用應用程式遷移服務將遷移重新託管到 Amazon EC2。

遷移組合評定 (MPA)

這是一種線上工具，可提供驗證要移轉至的商業案例的 AWS 雲端資訊。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)。所有 AWS 顧問和 APN 合作夥伴顧問均可免費使用 MPA 工具 (需要登入)。

遷移準備程度評定 (MRA)

使用 AWS CAF 獲得有關組織雲端準備狀態的見解、識別優勢和弱點，以及建立行動計劃以縮小已識別差距的過程。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

將工作負載移轉至 AWS 雲端. 如需詳細資訊，請參閱本詞彙表中的 7 Rs 項目，並參閱動員您的組織以加速大規模移轉。

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. 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，請參閱評估應用程式的現代化準備程度 AWS 雲端。

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

請參閱移轉組合評估。

MQTT

請參閱佇列遙測傳輸的郵件。

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？」

可變的基礎

一種模型，用於更新和修改生產工作負載的現有基礎結構。為了提高一致性，可靠性和可預測性，AWS Well-Architected 框架建議使用不可變的基礎結構作為最佳實踐。

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

請參閱作業整合。

OLA

請參閱作業層級協定。

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。

OPCA

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 提供數據加密，身份驗證和授權方案的互操作性標準。

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)。

操作準備程度檢討 (ORR)

問題和相關最佳做法的檢查清單，可協助您瞭解、評估、預防或減少事件和可能的故障範圍。如需詳細資訊，請參閱 AWS Well-Architected 的架構中的作業準備檢閱 (ORR)。

操作技術

可與實體環境搭配使用的硬體和軟體系統，以控制工業作業、設備和基礎設施。在製造業中，整合 OT 和資訊技術 (IT) 系統是工業 4.0 轉型的關鍵焦點。

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

由建立的追蹤 AWS CloudTrail 記錄中組織 AWS 帳戶 中所有人的所有事件 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文件中的為組織建立追蹤。

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 移轉策略中，這個架構稱為人員加速，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)

在中 CloudFront，限制存取權限以保護 Amazon Simple Storage Service (Amazon S3) 內容的增強選項。OAC 支援所有 S3 儲存貯體 AWS 區域、伺服器端加密 AWS KMS (SSE-KMS)，以及 S3 儲存貯體的動態PUT和DELETE請求。

原始存取身分 (OAI)

在中 CloudFront，用於限制存取以保護 Amazon S3 內容的選項。當您使用 OAI 時， CloudFront 會建立 Amazon S3 可用來進行驗證的主體。經驗證的主體只能透過特定散發存取 S3 儲存 CloudFront 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

請參閱作業整備檢閱。

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的網路連線的 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。

個人識別資訊 (PII)

直接查看或與其他相關數據配對時，可用於合理推斷個人身份的信息。PII 的範例包括姓名、地址和聯絡資訊。

PII

請參閱個人識別資訊。

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。

公司

請參閱可編程邏輯控制器

PLM

查看產品生命週期管理。

政策

可以定義權限 (請參閱以身分識別為基礎的策略)、指定存取條件 (請參閱以資源為基礎的策略) 或定義組織中所有帳戶的最大權限的物件 AWS Organizations (請參閱服務控制策略)。

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。如需詳細資訊，請參閱在微服務中啟用資料持久性。

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`，通常位於子`WHERE`句中。

謂詞下推

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

隱私設計

一種系統工程方法，在整個工程過程中將隱私權納入考量。

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

一種安全控制項，旨在防止部署不符合規範的資源。這些控制項會在資源佈建之前進行掃描。如果資源不符合控制項，則不會佈建該資源。如需詳細資訊，請參閱 AWS Control Tower 文件中的控制項參考指南，並參閱實作安全性控制中的主動控制 AWS。

產品生命週期管理 (PLM)

在產品的整個生命週期中管理資料和流程，從設計、開發、上市到成長與成熟度，再到下降和移除。

生產環境

請參閱環境。

可編程邏輯控制器 (PLC)

在製造業中，一台高度可靠且適應性強的計算機，可監控機器並自動化製造過程。

化名化

以預留位置值取代資料集中的個人識別碼的程序。化名化有助於保護個人隱私。假名化數據仍被認為是個人數據。

發布/訂閱 (發布/訂閱)

一種模式，可在微服務之間實現非同步通訊，以提高延展性和回應能力 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通道。系統可以在不變更發佈服務的情況下新增微服務。

Q

查詢計劃

一系列步驟，如指示，用來存取 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

拉齐矩阵

請參閱負責任，負責，諮詢，通知（RACI）。

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。

拉西矩陣

請參閱負責任，負責，諮詢，通知（RACI）。

RCAC

請參閱列與欄存取控制。

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。

重新建築師

見 7 盧比

復原點目標 (RPO)

自上次資料復原點以來可接受的時間上限。這決定了最後一個恢復點和服務中斷之間可接受的數據丟失。

復原時間目標 (RTO)

服務中斷與恢復服務之間的最大可接受延遲。

重構

見 7 盧比

區域

地理區域中的 AWS 資源集合。每個都 AWS 區域 是隔離和獨立於其他的，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 您的帳戶可以使用的項目。

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。

重新主持

見 7 盧比

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

見 7 盧比

再平台

見 7 盧比

買回

見 7 盧比

彈性

應用程式抵抗或從中斷中復原的能力。在規劃備援時，高可用性和災難復原是常見的考量因素。AWS 雲端如需詳細資訊，請參閱AWS 雲端 復原力。

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

定義移轉活動和雲端作業所涉及之所有各方的角色與責任的矩陣。矩陣名稱衍生自矩陣中定義的責任型別：負責 (R)、負責 (A)、諮詢 (C) 及通知 (I)。支撐（S）類型是可選的。如果您包含支援，則該矩陣稱為 RASCI 矩陣，如果您將其排除，則稱為 R ACI 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

見 7 盧比

退休

見 7 盧比

旋轉

定期更新密碼以使攻擊者更難以存取認證的程序。

資料列與資料行存取控制 (RCAC)

使用已定義存取規則的基本、彈性 SQL 運算式。RCAC 由資料列權限和資料行遮罩所組成。

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

許多身份提供者（IdPs）使用的開放標準。此功能可啟用聯合單一登入 (SSO)，因此使用者可以登入 AWS Management Console 或呼叫 AWS API 作業，而不必為組織中的每個人在 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

請參閱服務控制策略。

秘密

您以加密形式儲存的機密或受限制資訊，例如密碼或使用者認證。 AWS Secrets Manager它由秘密值及其中繼資料組成。密碼值可以是二進位、單一字串或多個字串。如需詳細資訊，請參閱「Secrets Manager 碼中有什麼內容？ 在 Secrets Manager 文檔中。

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。安全性控制有四種主要類型：預防性、偵測、回應式和主動式。

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

預先定義且程式化的動作，其設計用來自動回應或修復安全性事件。這些自動化作業可做為偵探或回應式安全控制項，協助您實作 AWS 安全性最佳實務。自動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

在其目的地的數據加密，通 AWS 服務 過接收它。

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 文件中的服務控制原則。

服務端點

的進入點的 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。

服務等級指示器 (SLI)

對服務效能層面的測量，例如錯誤率、可用性或輸送量。

服務等級目標 (SLO)

代表服務狀況的目標測量結果，由服務層次指示器測量。

共同責任模式

描述您在雲端安全性和合規方面共享的責任的模型。 AWS AWS 負責雲端的安全性，而您則負責雲端的安全性。如需詳細資訊，請參閱共同責任模式。

暹

請參閱安全性資訊和事件管理系統。

單點故障 (SPF)

應用程式的單一重要元件發生故障，可能會中斷系統。

SLA

請參閱服務等級協議。

SLI

請參閱服務層級指示器。

SLO

請參閱服務等級目標。

split-and-seed 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，請參閱中的應用程式現代化的階段化方法。 AWS 雲端

痙攣

請參閱單一故障點。

星型綱要

使用一個大型事實資料表來儲存交易或測量資料，並使用一或多個較小的維度表格來儲存資料屬性的資料庫組織結構。這種結構是專為在數據倉庫中使用或用於商業智能目的。

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。

監督控制與資料擷取 (SCADA)

在製造業中，使用硬體與軟體來監控實體資產與生產作業的系統。

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

以模擬使用者互動以偵測潛在問題或監控效能的方式測試系統。您可以使用 Amazon CloudWatch Synthetics 來創建這些測試。

T

標籤

作為組織 AWS 資源的中繼資料的索引鍵值配對。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。

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 文件中的傳輸閘道是什麼。

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

授與權限給您指定的服務，以代表您在組織內 AWS Organizations 及其帳戶中執行工作。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 AWS Organizations 文件中的AWS Organizations 與其他 AWS 服務搭配使用。

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

一個小 DevOps 團隊，你可以餵兩個比薩餅。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。如需詳細資訊，請參閱量化深度學習系統的不確定性指南。

無差別的任務

也稱為繁重工作，是創建和操作應用程序所必需的工作，但不能為最終用戶提供直接價值或提供競爭優勢。無差異化作業的範例包括採購、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

會危及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。

溫暖的數據

不常存取的資料。查詢此類資料時，通常可以接受中度緩慢的查詢。

視窗功能

一種 SQL 函數，可對以某種方式與當前記錄相關的一組行執行計算。視窗函數對於處理工作非常有用，例如計算移動平均值或根據目前列的相對位置存取列的值。

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。

蠕蟲

看到寫一次，多讀。

WQF

請參閱 AWS 工作負載鑑定架構。

寫一次，多讀（WORM）

一種儲存模型，可單次寫入資料並防止資料遭到刪除或修改。授權用戶可以根據需要多次讀取數據，但無法更改數據。這種數據存儲基礎設施被認為是不可變的。

Z

零日漏洞

一種利用零時差漏洞的攻擊，通常是惡意軟件。

零時差漏洞

生產系統中未緩解的瑕疵或弱點。威脅參與者可以利用這種類型的漏洞攻擊系統。由於攻擊，開發人員經常意識到該漏洞。

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。