使用 Amazon 設計和實作日誌記錄和監控 CloudWatch

AWS 規定指引

AWS 規定指引: 使用 Amazon 設計和實作日誌記錄和監控 CloudWatch

Table of Contents

使用 Amazon 設計和實作日誌記錄和監控 CloudWatch

尼扎米, Amazon Web Services (AWS)

2023 年 4 月 (文件歷史記錄)

本指南可協助您針對使用 Amazon 彈性運算雲端 (Amazon CloudWatch EC2AWS) 執行個體、Amazon Web Services 服務 (Amazon ECS)、亞馬遜彈性容器服務 (Amazon ECS) 、Amazon 彈性 Kubernetes 服務 (Amazon EKS) 和現場部署伺服器的工作負載，設計和實作記錄和監控 () 管理和控管服務。AWS Lambda本指南適用於在AWS雲端管理 DevOps 工作負載的作業團隊、工程師和應用程式工程師。

您的記錄和監控方法應以AWS Well-Architected 的架構的六個支柱為基礎。這些支柱包括卓越營運、安全性、可靠性、效能效率和成本最佳化。架構良好的監控和警示解決方案可協助您主動分析和調整基礎架構，進而改善可靠性和效能。

本指南不會廣泛討論安全性或成本最佳化的記錄和監控，因為這些主題需要深入評估。有許多支持安全日誌記錄和監控的AWS服務 AWS CloudTrailAWS Config，包括 Amazon Inspector，亞馬遜 Detective，Amazon Macie GuardDuty，亞馬遜和 AWS Security Hub. 您還可以使用AWS Cost ExplorerAWS預算和CloudWatch 計費指標進行成本優化。

下表概述了您的記錄和監控解決方案應該解決的六個區域。

擷取和擷取記錄檔和指標	識別、設定系統和應用程式記錄檔和指標，並將其傳送至不同來源的AWS服務。
搜尋和分析記錄檔	搜尋並分析作業管理、問題識別、疑難排解和應用程式分析的記錄。
監控指標和警報	識別工作負載中的觀察和趨勢並採取行動。
監控應用程式和服務可用	透過持續監控服務可用性，減少停機時間並提高符合服務等級目標的能力。
追蹤應用	追蹤系統中的應用程式要求和外部相依性，以微調效能、執行根本原因分析，以及疑難排解問題。
建立儀表板和視覺化	建立專注於系統和工作負載的相關指標和觀察的儀表板，有助於持續改善並主動探索問題。

CloudWatch 可滿足大部分的記錄和監控需求，提供可靠、可擴展且彈性的解決方案。除了用於監視和分析的 CloudWatch 記錄整合外，許多AWS服務還會自動提供 CloudWatch 指標。 CloudWatch 也提供代理程式和記錄驅動程式，以支援各種運算選項，例如伺服器 (雲端和內部部署)、容器和無伺服器運算。本指南也涵蓋下列與記錄和監視搭配使用的AWS服務：

- AWS Systems Manager代理商、Systems Manager 狀態管理員和 Systems Manager 自動化，可自動化、設定和更新 EC2 執行個體和現場部署伺服器的 CloudWatch 代理程式
- 適 OpenSearch 用於進階日誌彙總、搜尋和分析的 Amazon 服務
- Amazon Route 53 運作狀態檢查和 CloudWatchSynthetics，可監控應用程式和服務的可
- Amazon thethethethethethethetheteus 的 Amazon Managed thethetheteus 的 Amazon Managed the
- AWS X-Ray應用程式追蹤和執行階段分析
- 亞馬遜託管 Grafana 可視化和分析來自多個來源的數據（例如 CloudWatch，亞馬遜 OpenSearch 服務和 Amazon Timestream）

您選擇的AWS運算服務也會影響記錄和監控解決方案的實作和組態。例如， CloudWatchAmazon EC2，亞馬遜 ECS，亞馬遜 EKS 和 Lambda 的實施和配置不同。

應用程式和工作負載擁有者通常會忘記記錄和監控，或設定和實作不一致。這表示工作負載在可觀察性有限的情況下進入生產環境，這會導致識別問題的延遲，並增加疑難排解和解決問題所花費的時間。除了應用程式記錄檔和指標的應用程式層之外，您的記錄和監控解決方案至少必須解決作業系統 (OS) 層級記錄和指標的系統層。本指南提供針對不同運算類型 (包括下表所述的三種運算類型) 來解決這兩個層級的建議方法。

長時間執行且不可變的 EC2 執行個體	跨多個AWS區域或帳戶中多個作業系統 (OS) 的系統和應用程式記錄檔和指標。
容器	Amazon ECS 和 Amazon EKS 叢集的系統和應用程式日誌和指標，包括不同組態的範例。
無伺服器	Lambda 函數的系統和應用程式日誌和指標，以及自訂的考量事項。

本指南提供記錄與監控解決方案，可處理下列領域的相關AWS服務： CloudWatch

- 規劃您的 CloudWatch 部署— 規劃 CloudWatch 部署的考量，以及集中化 CloudWatch 組態的指導。
- 設定 CloudWatch EC2 執行個體和內部部署伺服器的代理程式— 系統層級和應用程式層級記錄和測量結果的 CloudWatch 組態詳細資料。
- 適用於 Amazon EC2 和本地服務器的 CloudWatch 代理安裝方法— 安裝 CloudWatch 代理程式的方法，包括跨多個區域和帳戶使用 Systems Manager 的自動部署。
- 在 Amazon ECS 上進行記錄和監控 — 在 Amazon ECS 中 CloudWatch 設定叢集層級和應用程式層級記錄和指標的指引。
- Amazon EKS 上的記錄和監控 — 在 Amazon EKS 中 CloudWatch 設定叢集層級和應用程式層級記錄和指標的指導。
- Amazon EKS 上的 Prometheus 監控— 介紹並比較適用於 Prometheus 的 Amazon 受管服務與 Prometheus 的 CloudWatch 容器洞察監控。
- 記錄和指標AWS Lambda— 設定 CloudWatch Lambda 函數的指引。
- 搜尋和分析記錄 CloudWatch— 使用 Amazon CloudWatch 應用程式洞察、日誌深入解析，以及將 CloudWatch 日誌分析延伸到 Amazon OpenSearch 服務來分析日誌的方法。
- CloudWatch 警示選項— 引入 CloudWatch 警報和 CloudWatch 異常偵測，並提供有關警報建立和設定的指導。
- 監控應用程序和服務可用性— 引入和比較 CloudWatch Synthetics 和 Route 53 健康檢查，以實現自動可用性監控。
- 使用跟蹤應用程序AWS X-Ray— 使用 X-Ray 進行亞馬遜 EC2，亞馬遜 ECS，亞馬遜 EKS 和 Lambda 進行應用程序跟踪的簡介和設置
- 使用雲手錶的儀錶板和可視化— CloudWatch 儀表板簡介，以提高AWS工作負載的可觀察性。
- 與 CloudWatch 整合AWS服務— 說明如何 CloudWatch 與各種AWS服務整合。
- 用於儀表板和可視化的亞馬遜託管 Grafana— 介紹和比較亞馬遜託管 Grafana 與 CloudWatch 儀表板和可視化。

本指南中使用的實作範例涵蓋這些領域，也可從 AWSSamples GitHub 存放庫取得。

目標業務成果

創建日誌記錄和監控解決方案，專為AWS雲是實現雲計算的六大優勢。您的日誌記錄和監控解決方案應有助於您的 IT 組織實現業務成果，使您的業務流程、業務合作夥伴、員工和客户受益。在實施日誌記錄和監控解決方案與AWSWell-Architected：

加快操作就緒性

啟用日誌記錄和監控解決方案是為生產支持和使用準備工作負載的重要組成部分。如果您過於依賴手動流程，運營就緒性會很快成為瓶頸，並且還可以縮短 IT 投資的價值實現時間 (TTV)。無效的方法還會導致工作負載的可觀察性有限。這可能會增加長期停機、客户不滿和業務流程失敗的風險。

您可以使用本指南的方法來標準化和自動化您的日誌記錄和監控AWS雲端 因此，新的工作負載需要最少的手動準備和幹預來進行生產記錄和監控。這還有助於減少為跨多個客户和區域的不同工作負載大規模創建日誌記錄和監控標準所需的時間和步驟。

提高操作效能

本指南為日誌記錄和監控提供了多種最佳實踐，可幫助不同的工作負載滿足業務目標和操作效能。本指南還提供了詳細示例和可重複使用的開源模板，您可以與基礎架構即代碼 (iAC) 方法一起使用，以實現結構良好的日誌記錄和監控解決方案，使用AWS服務。提高卓越運營是反覆性的，需要持續改進。本指南就如何不斷改進記錄和監測做法提供了建議。

增強操作可見性

您的業務流程和應用程序可能受到不同 IT 資源的支持，並託管在不同的計算類型上，無論是在本地還是AWS雲端 您的運營可見性可能會受到不一致和不完整的日誌記錄和監控策略實施的限制。採用全面的日誌記錄和監控方法可幫助您快速識別、診斷和響應工作負載中的問題。本指南可幫助您設計和實施各種方法，以提高完整的操作可視性，並縮短平均解決 (MTTR) 故障的時間。全面的日誌記錄和監控方法還可幫助您的組織提高服務質量、增強最終用户體驗並滿足服務級別協議 (SLA) 要求。

擴展運營並降低間接成本

您可以根據本指南擴展日誌記錄和監控實踐，以支持多個區域和帳户、短期資源和多個環境。本指南提供了自動執行手動步驟的方法和示例（例如，安裝和配置代理、監控指標以及在出現問題時通知或採取

措施）。當您的雲採用成熟和增長，並且您需要在不增加雲管理活動或資源的情況下擴展運營能力時，這些方法非常有用。

規劃您的 CloudWatch 部署

記錄和監控解決方案的複雜性和範圍取決於以下幾個因素，包括：

- 使用了多少環境、區域和帳戶，以及此數目可能如何增加。
- 現有工作負載和架構的種類和類型。
- 必須記錄和監控的運算類型和作業系統。
- 是否同時有內部部署位置和 AWS 基礎結構。
- 多個系統和應用程序的聚合和分析要求。
- 防止未經授權暴露日誌和指標的安全要求。
- 必須與您的記錄和監控解決方案整合的產品和解決方案，以支援作業流程。

您必須使用新的或更新的工作負載部署定期檢閱和更新記錄和監控解決方案。在觀察到問題時，應識別並套用記錄、監控和警示的更新。然後可以主動識別這些問題，並在 future 預防。

您必須確保持續安裝和設定軟體和服務，以擷取和擷取記錄檔和指標。已建立的記錄和監控方法會針對不同網域 (例如安全性、效能、網路 AWS 或分析)，使用多個或獨立的軟體廠商 (ISV) 服務和解決方案。每個網域都有自己的部署和組態需求。

我們建議您使 CloudWatch 用擷取和擷取多個作業系統和運算類型的記錄和指標。許多 AWS 服務用 CloudWatch 於記錄、監視和發佈記錄和指標，無需進一步設定。 CloudWatch 提供可針對不同作業系統和環境安裝和設定的軟體代理程式。下列各節概述如何針對多個帳戶、區域和組態部署、安裝和設定 CloudWatch 代理程式：

主題

- CloudWatch 在集中式或分散式帳戶中使用
- 管理 CloudWatch 代理程式組態檔

CloudWatch 在集中式或分散式帳戶中使用

雖然設計 CloudWatch 用來監控一個帳戶和區域中的 AWS 服務或資源，但您可以使用中央帳戶從多個帳戶和區域擷取記錄和指標。如果您使用多個帳戶或區域，則應評估是使用集中式帳戶方法還是使用個別帳戶來擷取記錄和指標。一般而言，多帳戶和多區域部署需要混合式方法，以支援安全性、分析、作業和工作負載擁有者的需求。

下表提供選擇使用集中式、分散式或混合方法時要考慮的領域。

帳戶結構	您的組織可能有數個單獨的帳戶 (例如，針對非生產和生產工作負載的帳戶)，或針對特定環境中的單一應用程式擁有數千個帳戶。建議您在執行工作負載的帳戶中維護應用程式記錄檔和指標，以便讓工作負載擁有者存取記錄檔和指標。這使他們能夠在記錄和監視中扮演積極的角色。我們也建議您使用個別的記錄帳戶來彙總所有工作負載記錄，以進行分析、彙總、趨勢和集中式作業。單獨的日誌記錄帳戶也可以用於安全性，歸檔和監視以及分析。
存取要求	團隊成員 (例如，工作負載擁有者或開發人員) 需要存取記錄檔和指標，才能進行疑難排解並進行改善。應該在工作負載的帳戶中維護記錄，以便更輕鬆地進行存取和疑難排解。如果記錄檔和指標是在與工作負載不同的帳戶中維護，則使用者可能需要定期在帳戶之間切換。 使用集中式帳戶可為授權的使用者提供記錄資訊，而不會授與工作負載帳戶的存取權 這可以簡化分析工作負載的存取需求，其中需要從多個帳戶中執行的工作負載進行彙總。集中式記錄帳戶也可以有替代的搜尋和彙總選項，例如 Amazon Ser OpenSearch vice 叢集。Amazon OpenSearch 服務可為您的日誌提供精細的存取控制，直到欄位層級。當您擁有需要特殊存取權和權限的敏感或機密資料時，精細的存取控制非常重要。
操作	許多組織都有一個集中的操作和安全團隊或外部組織來提供操作支持，這些組織需要訪問日誌以進行監視。集中式記錄和監控可讓您更輕鬆地識別所有帳戶和工作負載的趨勢、搜尋、彙總和執行分析。如果您的組織使用「您建置，您執行它」方法 DevOps，則工作負載擁有者需要在其帳戶中記錄和監視資訊。除了分散式工作負載擁有權之外，還需要使用混合式方法來滿足中央作業和分析。
Environment (環境)	您可以選擇將日誌和指標託管在生產帳戶的集中位置，並根據安全性需求和帳戶架構，將其他環境（例如開發或測試）的日誌和指標保存在相同或單獨的帳戶中。這有助於防止在生產過程中創建的敏感數據被更廣泛的受眾訪問。

CloudWatch 提供多種選項，可使用 CloudWatch 訂閱過濾器即時處理日誌。您可以使用訂閱篩選器將記錄即時串流至 AWS 服務，以進行自訂處理、分析和載入至其他系統。如果您採用混合式方法，除了集中式帳戶和區域之外，您還可以在個別帳戶和區域取得記錄和指標，這項功能會特別有用。下列清單提供可用於此功能的 AWS 服務範例：

- Amazon 資料防火軟管 — Firehose 提供串流解決方案，可根據所產生的資料量自動擴展和調整大小。您不需要管理 Amazon Kinesis 資料串流中的碎片數量，而且無需額外編碼即可直接連線到亞馬遜簡單儲存服務 (Amazon S3)、亞馬遜 OpenSearch 服務或 Amazon Redshift。如果您想將日誌集中在這 AWS 些服務中，Firehose 是一種有效的解決方案。
- Amazon Kinesis Data Streams — 如果您需要與 Firehose 不支援的服務整合並實作其他處理邏輯，則 Kinesis Data Streams 流是適當的解決方案。您可以在帳戶和區域中建立 Amazon CloudWatch Logs 目的地，以在中央帳戶中指定 Kinesis 資料串流，以及授予其在串流中放置記錄的權限的 AWS Identity and Access Management (IAM) 角色。Kinesis Data Streams 為您的記錄資料提供彈性、開放式的 landing zone，然後可供不同的選項使用。您可以將 Kinesis Data Streams 記錄資料讀入您的帳戶、執行預先處理，然後將資料傳送到您選擇的目的地。

  不過，您必須設定串流的碎片，以便針對所產生的記錄資料進行適當的大小調整。Kinesis Data Streams 可做為記錄資料的暫時中介或佇列，而且您可以將資料儲存在 Kinesis 串流中一到 365 天。Kinesis Data Streams 也支援重新顯示功能，這表示您可以重播未使用的資料。
- Amazon Ser OpenSearch vice — CloudWatch 日誌可以將日誌群組中的日誌串流到個別或集中式帳戶中的 OpenSearch 叢集。當您將日誌群組設定為將資料串流至 OpenSearch 叢集時，Lambda 函數會在與日誌群組相同的帳戶和區域中建立。Lambda 函數必須與 OpenSearch 叢集建立網路連線。除了將擷取自訂至 Amazon OpenSearch 服務之外，您還可以自訂 Lambda 函數以執行其他預處理。使用 Amazon Ser OpenSearch vice 進行集中式記錄，可讓您更輕鬆地分析、搜尋和疑難排解雲端架構中多個元件的問題。
- Lambda — 如果您使用 Kinesis Data Streams，則需要佈建和管理耗用串流資料的運算資源。為避免這種情況，您可以將日誌資料直接串流至 Lambda 進行處理，並根據邏輯將其傳送至目的地。這表示您不需要佈建和管理運算資源即可處理傳入的資料。如果您選擇使用 Lambda，請確定您的解決方案與 Lambda 配額相容。

您可能需要以檔案格式處理或共用 CloudWatch 記錄檔中儲存的記錄資料。您可以建立匯出任務，將日誌群組匯出到 Amazon S3 的特定日期或時間範圍。例如，您可以選擇每天將日誌匯出到 Amazon S3 以進行分析和稽核。Lambda 可用來自動化此解決方案。您也可以將此解決方案與 Amazon S3 複寫結合使用，將多個帳戶和區域的日誌交付和集中到一個集中式帳戶和區域。

CloudWatch 代理程式組態也可以在agent區credentials段中指定欄位。這會指定將指標和記錄傳送至其他帳戶時要使用的 IAM 角色。如果有指定，此欄位會包含role_arn參數。當您只需要在特定的集中式帳戶和區域中進行集中式記錄和監控時，可以使用此欄位。

您也可以使用 AWS SDK，以您選擇的語言撰寫自己的自訂處理應用程式、讀取帳戶中的日誌和指標，以及將資料傳送到集中式帳戶或其他目的地以進行進一步處理和監控。

管理 CloudWatch 代理程式組態檔

我們建議您建立標準 Amazon CloudWatch 代理程式組態，其中包含要在所有 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體和現場部署伺服器擷取的系統日誌和指標。您可以使用 CloudWatch 代理程式組態檔精靈來協助您建立組態檔。您可以多次執行組態精靈，為不同的系統和環境產生唯一的組態。您也可以使用組態檔案結構描述來修改組態檔案或建立變體。 CloudWatch 代理程式組態檔可以存放在 AWS Systems Manager Parameter Store 參數中。 如果您有多個 CloudWatch 代理程式組態檔，則可以建立個別的參數存放區參數。如果您使用多個 AWS 帳戶或 AWS 區域，則必須管理和更新每個帳戶和區域中的參數存放區參數。或者，您可以將 CloudWatch 組態集中管理為 Amazon S3 中的檔案或您選擇的版本控制工具。

CloudWatch代理程式隨附的指amazon-cloudwatch-agent-ctl令碼可讓您指定組態檔、參數存放區參數或代理程式的預設組態。預設組態會對齊預先定義的基本量度集，並設定代理程式以將記憶體和磁碟空間度量報告為。 CloudWatch但是，它不包括任何日誌文件配置。如果您使用 CloudWatch 代理程式的「Systems Manager 快速設定」，也會套用預設組態。

由於預設設定不包含記錄，且不會根據您的需求自訂，因此建議您根據需求建立並套用自己的組 CloudWatch 態。

管理 CloudWatch組態

依預設，組 CloudWatch 態可以儲存並套用為參數存放區參數或 CloudWatch 組態檔案。 最佳選擇取決於您的要求。 在本節中，我們討論了這兩個選項的利弊。 有關管理多個 AWS 帳戶和 AWS 區域的 CloudWatch 組態檔時，也會詳細介紹代表性的解決方案。

Systems Manager 參數儲存參數

如果您有要在一小 CloudWatch 組 AWS 帳戶和區域中套用和管理的單一標準 CloudWatch 代理程式組態檔案，則使用參數存放區參數管理組態很有效。將組 CloudWatch 態儲存為參數存放區參數時，您可以使用 CloudWatch 代理程式組態工具 (amazon-cloudwatch-agent-ctl在 Linux 上) 從參數存放區讀取並套用組態，而不需要將組態檔複製到執行個體。您可以使用 AmazonCloudWatch-ManageAgent Systems Manager 命令文件，在單次執行中更新多個 EC2 執行個體的 CloudWatch 組態。由於參數存放區參數是區域性的，因此您 CloudWatch 必須更新和維護每個 AWS 區域和 AWS

帳戶中的參數存放區參數。如果您有多個 CloudWatch 模型組態要套用至每個實例，您必須自訂 AmazonCloudWatch-ManageAgent Command 文件來包括這些參數。

CloudWatch 組態檔

如果您 CloudWatch 有許多 AWS 帳戶和區域，而且要管理多個 CloudWatch組態檔案，則以檔案形式管理您的組態可能會有效。使用此方法，您可以在資料夾結構中瀏覽、組織和管理它們。 您可以將安全性規則套用至個別資料夾或檔案，以限制和授與存取權限，例如更新和讀取權限。您可以在 AWS 之外共用和傳輸它們以進行協同合作。 您可以版本控制文件以跟踪和管理更改。 您可以將組 CloudWatch 態檔複製到 CloudWatch 代理程式組態目錄來統一套用組態，而不需個別套用每一個組態檔。對於 Linux， CloudWatch 組態目錄位於`/opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d`。對於視窗，您可以在中找到組態目錄`C:\ProgramData\Amazon\AmazonCloudWatchAgent\Configs`。

當您啟動代理程式時， CloudWatch 代理程式會自動附加在這些目錄中找到的每個檔案，以建立 CloudWatch 複合組態檔。組態檔案應存放在中央位置 (例如 S3 儲存貯體)，您所需的帳戶和區域可存取。 提供使用此方法的範例解決方案。

組織組 CloudWatch 態

無論用於管理組 CloudWatch 態的方法為何，都可以組織您的組 CloudWatch 態。您可以使用如下方法將組態組織到檔案或參數存放區路徑中。

/配置/標準/窗口/ec2	為 Amazon EC2 存放特定於 Windows 的標準 CloudWatch 組態檔案。您可以針對此資料夾下的不同 Windows 版本、EC2 執行個體類型和環境，進一步分類標準作業系統 (OS) 組態。
/配置/標準/窗戶/內部	儲存內部部署伺服器的標準 Windows 特定 CloudWatch 組態檔案。您也可以針對此資料夾下的不同 Windows 版本、伺服器類型和環境，進一步分類標準作業系統組態。
/配置/標準/鏈線/ec2	為 Amazon EC2 存放您的標準 Linux 特定 CloudWatch 組態檔案。您可以針對此資料夾下的不同 Linux 發行版、EC2 執行個體類型和環境，進一步分類標準作業系統組態。
/配置/標準/鏈條/內部	儲存內部部署伺服器的標準 Linux 特定 CloudWatch 組態檔案。您可以針對此資料夾下

	的不同 Linux 發行版、伺服器類型和環境，進一步分類標準作業系統組態。
/配置/ECS	如果您使用 Amazon ECS 容器執行個體，請儲存亞馬遜彈性容器服務 (Amazon ECS) 專用的 CloudWatch 組態檔案。這些組態可附加至標準 Amazon EC2 組態，以進行 Amazon ECS 特定系統層級記錄和監控。
/配置/ <application_name>	儲存應用程式特定的 CloudWatch 組態檔案。您可以使用環境和版本的其他資料夾和前置詞，進一步分類您的應用程式。

範例：將 CloudWatch 組態檔案儲存在 S3 儲存貯體

本節提供使用 Amazon S3 存放 CloudWatch組態檔案的範例，以及使用自訂 Systems Manager 執行手冊來擷取和套用 CloudWatch 組態檔案的範例。此方法可以解決使用 Systems Manager 參數存放區參數進行大規模 CloudWatch組態時所面臨的一些挑戰：

- 如果您使用多個區域，則必須同步每個區域的參數存放區中的 CloudWatch組態更新。參數存放區是區域服務，且必須在使用 CloudWatch 代理程式的每個區域中更新相同的參數。
- 如果您有多個組 CloudWatch 態，則必須初始化每個參數存放區組態的擷取和應用程式。您必須個別從「參數存放區」擷取每個 CloudWatch 組態，並在新增組態時更新擷取方法。相反地，CloudWatch 提供用於儲存組態檔的組態目錄，並套用目錄中的每個組態，而不需要個別指定它們。
- 如果您使用多個帳戶，則必須確保每個新帳戶在其參數存放區中具有必要的 CloudWatch 組態。您也必須確定 future 會將任何組態變更套用至這些帳戶及其區域。

您可以將 CloudWatch 組態存放在可從所有帳戶和區域存取的 S3 儲存貯體中。然後，您可以使用系統管理員自動化手冊和系統管理員狀態管理員，將這些 CloudWatch 組態從 S3 儲存貯體複製到組態目錄。您可以使用 cloudwatch-config-s3-bucket.yaml AWS CloudFormation 範本建立 S3 儲存貯體，該儲存貯體可從 AWS Organizations 中的組織內的多個帳戶存取。範本包含一個`OrganizationID`參數，可授與組織內所有帳戶的讀取權限。

本指南的「為代理程 CloudWatch 式部署和組態設定狀態管理員和代理商」一節中提供的增強型系統管理員執行手冊，設定為使用 cloudwatch-config-s3-buc ket.yaml AWS 範本建立的 S3 儲存貯體擷取檔案。 CloudFormation

GitHub或者，您可以使用版本控制系統 (例如 AWS CodeCommit) 來存放組態檔。如果您想要自動擷取儲存在版本控制系統中的組態檔案，您必須管理或集中化認證儲存，並更新用來擷取跨帳戶和區域認證的 Systems Manager Automation 執行手冊。

設定 CloudWatch EC2 執行個體和內部部署伺服器的代理程式

許多組織在物理伺服器和虛擬機器 (VM) 上運行負載。這些工作負載通常在不同的操作系統上運行，每個操作系統都有獨特的安裝和配置要求來捕獲和接收指標。

如果您選擇使用 EC2 實例，則可以對實例和操作系統配置進行高級別控制。但是，這種更高級別的控制和責任要求您監控和調整配置，以實現更高效的使用。通過建立日誌記錄和監控標準，並應用標準的安裝和配置方法來捕獲和攝取日誌和指標，您可以提高運營效率。

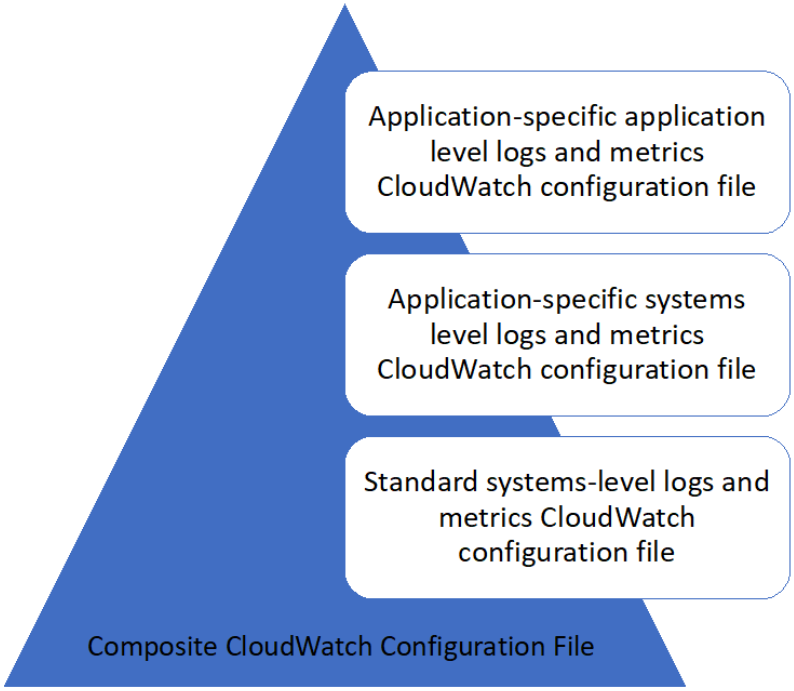
將其 IT 投資遷移或擴展到AWS雲可以利用 CloudWatch 來實現統一的日誌記錄和監控解決方案。CloudWatch 定價意味着您要捕獲的指標和日誌以遞增方式支付費用。您還可以通過使用類似的 CloudWatch 代理安裝過程與 Amazon EC2 相同。

在開始安裝和部署 CloudWatch 之前，請確保評估系統和應用程序的日誌記錄和指標配置。確保為要使用的操作系統定義了需要捕獲的標準日誌和指標。系統日誌和指標是日誌記錄和監視解決方案的基礎和標準，因為它們是由操作系統生成的，對於 Linux 和 Windows 而言是不同的。除了特定於 Linux 版本或發行版的指標和日誌文件之外，還有跨 Linux 發行版提供的重要指標和日誌文件。不同的 Windows 版本之間也會出現這種差異。

設定 CloudWatch 代理人

CloudWatch Amazon EC2 通過使用CloudWatch 代理程式組態檔案特定於每個操作系統的。我們建議您定義組織的標準指標和日誌捕獲配置，然後再開始安裝 CloudWatch 在您的帳户中規模代理。

您可以結合多個 CloudWatch 代理配置以形成複合 CloudWatch 代理程式組態。推薦的一種方法是在系統和應用程序級別定義和劃分日誌和指標的配置。下圖説明瞭如何將滿足不同要求的多種 CloudWatch 配置文件類型組合起來形成複合 CloudWatch 配置：



還可以針對特定環境或要求對這些日誌和指標進行進一步分類和配置。例如，您可以為不受管制的開發環境定義較小的日誌和指標子集，而且可以為受管制的生產環境定義更小的精度較小的日誌和指標子集。

配置 EC2 實例的日誌捕獲

默認情況下，Amazon EC2 不監視或捕獲日誌文件。相反，將捕獲日誌文件並將其攝入到 CloudWatch 日誌由 CloudWatch 代理程式組態軟件，AWSAPI，或AWS Command Line Interface(AWS CLI。我們建議使用 CloudWatch 代理將日誌文件提取到 CloudWatch Amazon EC2 和內部部署伺服器的日誌。

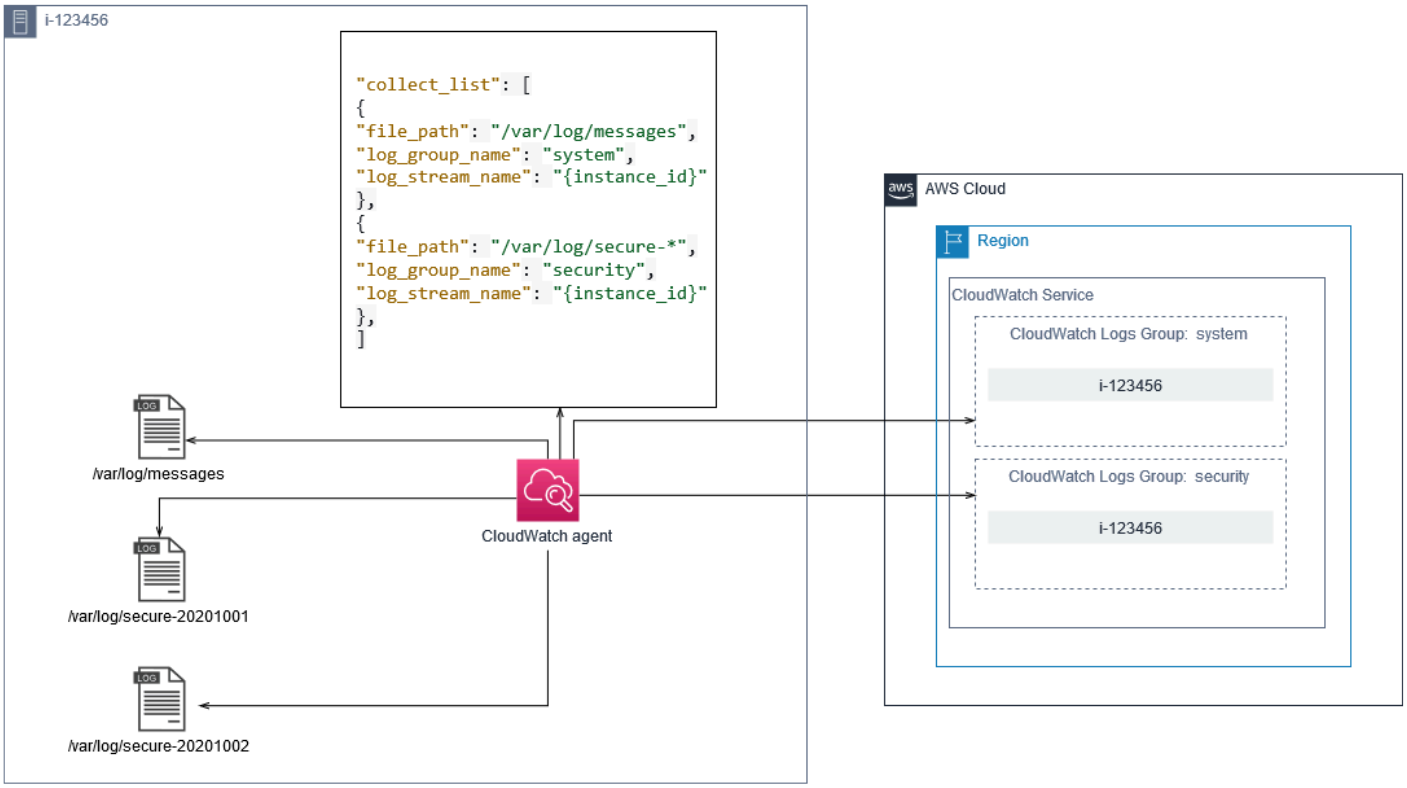
您可以搜索和篩選日誌，以及從 CloudWatch 中的日誌文件中提取指標和基於模式修補程序運行自動化。 CloudWatch 支持明文、空格分隔和 JSON 格式的過濾器和模式語法選項，使用 JSON 格式的日誌提供了最大的靈活性。要增加篩選和分析選項，應使用格式化的日誌輸出而不是純文本。

所以此 CloudWatch 代理使用定義要發送到 CloudWatch 的日誌和指標的配置文件。 CloudWatch 然後將每個日誌文件捕獲為日誌串流並將這些日誌流分組到日誌群組。這有助於您跨 EC2 實例的日誌執行操作，例如搜索匹配的字符串。

默認日誌流名稱與 EC2 實例 ID 相同，默認日誌組名稱與日誌文件路徑相同。日誌流的名稱必須在 CloudWatch 日誌羣組。您可以使用`instance_id`、`hostname`、`local_hostname`,

或ip_address用於日誌流和日誌組名稱中的動態替換，這意味着您可以使用相同的 CloudWatch 代理程式組態檔案。

下圖顯示 CloudWatch 用於捕獲日誌的代理配置。日誌組由捕獲的日誌文件定義，幷包含每個 EC2 實例的單獨日誌流，因為{instance_id}變量用於日誌流名稱，EC2 實例 ID 是唯一的。



日誌組定義其所包含的日誌流的保留期、標記、安全性、度量篩選器和搜索範圍。基於日誌文件名的默認分組行為可幫助您在賬户和區域中跨 EC2 實例中搜索、創建指標和警報特定於日誌文件的數據。您應評估是否需要進一步的日誌組細化。例如，您的帳户可能由多個業務部門共享，並且具有不同的技術或運營所有者。這意味着您必須進一步優化日誌組名稱以反映分離和所有權。通過此方法，您可以將分析和故障排除集中在相關 EC2 實例上。

如果多個環境使用一個帳户，則可以為每個環境中運行的工作負載分開日誌記錄。下表顯示了一個日誌組命名約定，其中包括業務部門、項目或應用程序以及環境。

日誌群組名稱	/<Business unit>/<Project or application name>/<Environment>/<Log file name>
日誌串流名稱	<EC2 instance ID>

您還可以將 EC2 實例的所有日誌文件分組到同一日誌組中。這樣就可以更輕鬆地跨單個 EC2 實例的一組日誌文件進行搜索和分析。如果您的大多數 EC2 實例服務一個應用程序或工作負載，並且每個 EC2 實例都有特定用途，則此功能非常有用。下表顯示瞭如何格式化日誌組和日誌流命名以支持此方法。

日誌群組名稱	`/<Business unit>/<Project or application name>/<Environment>/<EC2 instance ID>`
日誌串流名稱	`<Log file name>`

配置 EC2 實例的指標捕獲

默認情況下，您的 EC2 執行個體會啟用基本監控和標準指標集（例如，CPU、網絡或與存儲相關的指標）會自動發送到 CloudWatch 每五分鐘一次。 CloudWatch 指標可能因執行個體系而有所差異，例如爆量效能執行個體具有 CPU 積分的指標。Amazon EC2 標準指標包含在您的實例價格中。如果您啟用詳細監控，您可以在一分鐘內接收數據。週期頻率會影響您的 CloudWatch 成本，因此請確保評估是否需要對所有 EC2 實例進行詳細監控，還是僅需要對其中的某些實例進行詳細監控。例如，您可以啟用對生產工作負載的詳細監控，但對非生產工作負載使用基本監控。

本地服務器不包含 CloudWatch 並且必須使用 CloudWatch 代理程式,AWS CLI, 或AWS用於捕獲指標的 SDK。這意味着您必須定義要捕獲的指標（例如 CPU 利用率）在 CloudWatch 組態檔案。您可以建立唯一的 CloudWatch 配置文件，該文件包含您的本地服務器的標準 EC2 實例指標，並將其應用於您的標準 CloudWatch 組態。

指標在 CloudWatch 由指標名稱和零或多個維度進行唯一的定義，並且在指標命名空間中唯一地進行分組。提供的指標AWS服務的名稱空間以AWS（例如，AWS/EC2) 和非AWS指標會被視為自定義指標。您配置和捕獲的指標使用 CloudWatch 代理都被視為自定義指標。由於創建的指標數量會影響您的 CloudWatch 成本，您應評估您的所有 EC2 實例是否需要每個指標，還是僅需要部分指標。例如，您可以為生產工作負載定義一組完整的指標，但對非生產工作負載使用這些指標的較小子集。

`CWAgent`是發佈的指標的默認命名空間 CloudWatch 代理程式。與日誌組類似，指標命名空間組織了一組指標，以便在一個位置一起找到這些指標。您應該修改命名空間以反映業務部門、項目或應用程序以及環境（例如`/<Business unit>/<Project or application name>/<Environment>`。如果多個不相關的工作負載使用同一帳户，則此方法非常有用。您還可以將命名空間命名約定與 CloudWatch 日誌組命名約定。

衡量指標也由其維度標識，這有助於您根據一組條件對其進行分析，並且是記錄觀測值所依據的屬性。Amazon EC2 包括單獨指標的 EC2 執行個體`InstanceId`和`AutoScalingGroupName`維度。您

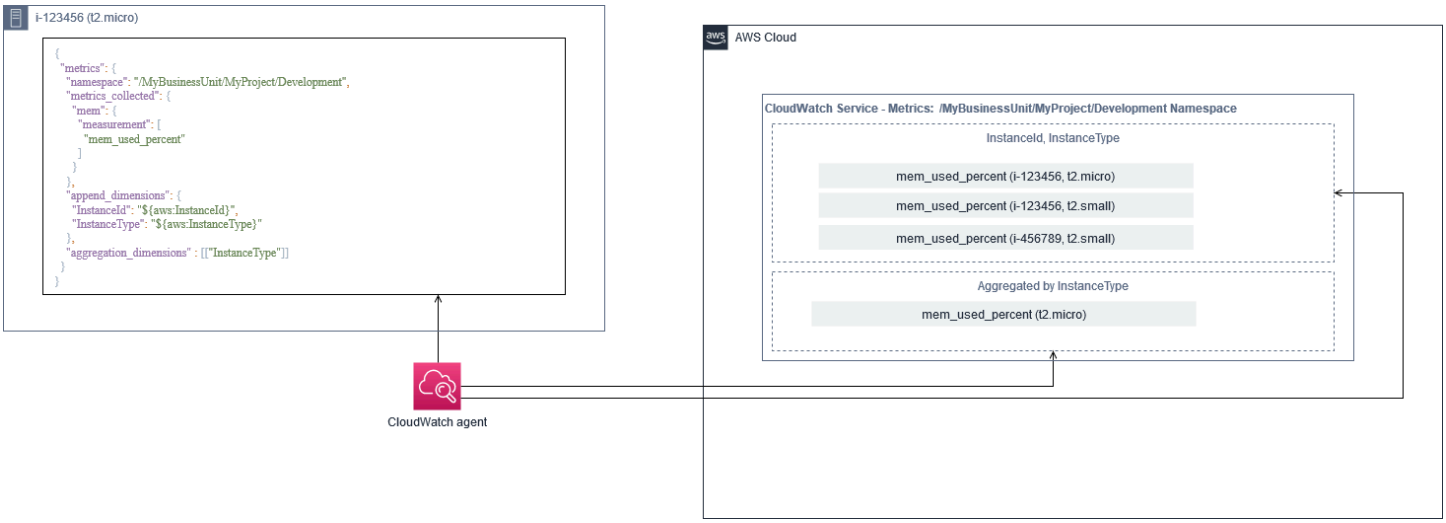
還會收到具有ImageId和InstanceType維度（如果啟用詳細監控）。例如，Amazon EC2 為 CPU 利用率提供了一個單獨的 EC2 實例指標，其中包含InstanceId維度，此外，除了單獨的 CPU 使用率指標InstanceType維度。這有助於您分析每個唯一 EC2 實例的 CPU 利用率，以及特定執行個體類型。

添加更多維度可增加您的分析能力，但也會增加您的總體成本，因為每個量度和唯一維度值組合都會生成一個新的指標。例如，如果您為內存利用率百分比創建了一個指標InstanceId維度，那麼這是每個 EC2 實例的新指標。如果您的組織運行數千個 EC2 實例，這會導致數千個指標並導致更高的成本。要控制和預測成本，請確保您確定度量的基數以及哪些維度增加最大值。例如，您可以為生產工作負載指標定義一組完整的維度，但為非生產工作負載定義這些維度的較小子集。

您可以使用append_dimensions屬性將維度添加到 CloudWatch 組態。您還可以動態地追加ImageId、InstanceId、InstanceType，和AutoScalingGroupName中的所有指標 CloudWatch 組態。或者，您可以通過使用append_dimensions屬性。 CloudWatch 還可以聚合指標維度的統計信息，這些指標維度使用aggregation_dimensions屬性。

例如，您可以將使用的內存聚合到InstanceType維度查看每個實例類型的所有 EC2 實例使用的平均內存。如果您使用t2.micro實例時，您可以確定是否使用t2.micro類過度利用或未充分利用提供的內存。未充分利用可能是使用具有不需要內存容量的 EC2 類的工作負載的標誌。相比之下，過度利用可能是使用內存不足的 Amazon EC2 類的工作負載的標誌。

下圖顯示示例 CloudWatch 度量配置，該配置使用自定義命名空間、添加的維度和聚合InstanceType。



系統層級 CloudWatch 配置

系統級別的指標和日誌是監控和日誌記錄解決方案的核心組成部分， CloudWatch 代理具有適用於 Windows 和 Linux 的特定配置選項。

建議您使用CloudWatch 配置文件嚮導或配置文件架構來定義 CloudWatch 代理配置文件，以便您計劃支持的每個操作系統。其他特定於工作負載的操作系統級日誌和指標可以在單獨的 CloudWatch 配置文件並附加到標準配置中。這些唯一的配置文件應單獨存儲在 S3 存儲桶中，您的 EC2 實例可在該存儲桶中檢索它們。為此目的設置 S3 存儲桶的示例，請參閱管理 CloudWatch組態本指南的一節。您可以使用狀態管理器和分發服務器自動檢索和應用這些配置。

設定系統層級日誌

系統級日誌對於診斷和故障排除本地或AWS雲端。日誌捕獲方法應包括操作系統生成的任何系統和安全日誌。操作系統生成的日誌文件可能因操作系統版本而有差異。

所以此 CloudWatch 代理支持通過提供事件日誌名稱來監視 Windows 事件日誌。您可以選擇要監視的 Windows 事件日誌（例如`System`、`Application`, 或`Security`。

Linux 系統的系統、應用程序和安全日誌通常存儲在`/var/log`目錄。下表定義了您應該監視的常見默認日誌文件，但您應檢查`/etc/rsyslog.conf`或者`/etc/syslog.conf`文件來確定系統日誌文件的特定設置。

Fedora 分佈	`/var/log/boot.log*` — 啟動日誌
（亞馬遜 Linux、CentOS、Red Hat Enterprise Linux）	`/var/log/dmesg` — 核心日誌
	`/var/log/secure` — 安全和身份驗證日誌
	`/var/log/messages` — 常規系統日誌
	`/var/log/cron*` — Cron 日誌
	`/var/log/cloud-init-output.log` — 輸出自`Userdata`啟動程式腳本
Debian	`/var/log/syslog` — 啟動日誌
(Ubuntu)	

		/var/log/cloud-init-output.log — 輸出自Userdata啟動程式腳本
		/var/log/auth.log — 安全和身份驗證日誌
		/var/log/kern.log — 核心日誌

您的組織可能還有生成要監視的日誌的其他代理或系統組件。您應評估並決定這些代理或應用程序生成的日誌文件，並通過標識其文件位置將它們包含在配置中。例如，您應該包含 Systems Manager 和 CloudWatch 代理登錄到配置中。下表提供了適用於 Windows 和 Linux 的這些代理日誌的位置。

Windows	CloudWatch 代理程式	$Env:ProgramData\Amazon\AmazonCloudWatchAgent\Logs\amazon-cloudwatch-agent.log
	Systems Manager 程式代理	%PROGRAMDATA%\Amazon\SSM\Logs\amazon-ssm-agent.log %PROGRAMDATA%\Amazon\SSM\Logs\errors.log %PROGRAMDATA%\Amazon\SSM\Logs\audits\amazon-ssm-agent-audit-YYYY-MM-DD
Linux	CloudWatch 代理程式	/opt/aws/amazon-cloudwatch-agent/logs/amazon-cloudwatch-agent.log
	Systems Manager 程式代理	/var/log/amazon/ssm/amazon-ssm-agent.log

		/var/log/amazon/ssm/errors.log /var/log/amazon/ssm/audits/amazon-ssm-agent-audit-YYYY-MM-DD

如果日誌文件是在 CloudWatch 代理配置，但未找到。當您希望為 Linux 維護單個日誌配置時，這非常有用，而不是為每個發行版單獨配置。當日誌文件在代理或軟件應用程序開始運行之前不存在時，它也很有用。

設定系統層級指標

Amazon EC2 提供的標準指標中不包括內存和磁盤空間利用率。若要包括這些指標，您必須先安裝和設定 CloudWatch 代理。所以此 CloudWatch 代理程式組態嚮導會建立 CloudWatch 使用的組態預先定義指標，您可以根據需要添加或移除指標。請確保您查看預定義的指標集以確定所需的適當級別。

最終用户和工作負載所有者應根據服務器或 EC2 實例的特定要求發佈其他系統指標。這些指標定義應該存儲、版本化和維護在單獨的 CloudWatch 代理配置文件，並在中心位置（例如 Amazon S3）共享，以便重複使用和自動化。

標準 Amazon EC2 指標不會在本地服務器中自動捕獲。這些指標必須定義於 CloudWatch 本地實例使用的代理配置文件。您可以為具有 CPU 利用率等指標的本地實例創建單獨的指標配置文件，並將這些指標附加到標準指標配置文件中。

應用程式層級 CloudWatch 配置

應用程序日誌和指標由運行的應用程序生成，並且特定於應用程序。確保定義了充分監視組織經常使用的應用程序所需的日誌和指標。例如，您的組織可能已針對基於 Web 的應用程序在 Microsoft 互聯網信息服務器 (IIS) 上進行了標準化。您可以創建標準日誌和指標 CloudWatch 配置，也可以在您的組織中使用。特定於應用程序的配置文件可以存儲在集中位置（例如 S3 存儲桶）中，由工作負載所有者或通過自動檢索進行訪問，然後複製到 CloudWatch 組態目錄。所以此 CloudWatch 代理自動將每個 EC2 實例或服務器的配置文件目錄中找到的 CloudWatch 配置文件合併到複合 CloudWatch 組態。最終結果是 CloudWatch 配置，其中包括組織的標準系統級配置以及所有相關的應用程序級別 CloudWatch 配置。

工作負載所有者應為所有關鍵應用程序和組件確定並配置日誌文件和衡量指標。

配置應用程序級日誌

應用程序級日誌記錄取決於應用程序是否為商業 off-the-shelf （COTS）或定製開發的應用程序。COTS 應用程序及其組件可能為日誌配置和輸出提供多種選項，例如日誌詳細信息級別、日誌文件格式和日誌文件位置。但是，大多數 COTS 或第三方應用程序不允許您從根本上更改日誌記錄（例如，更新應用程序的代碼以包含其他日誌語句或不可配置的格式）。至少，您應該為 COTS 或第三方應用程序配置日誌記錄選項，以記錄警告和錯誤級別信息，最好採用 JSON 格式。

您可以將定製開發的應用程序與 CloudWatch 通過將應用程序的日誌文件包含在 CloudWatch 組態。自定義應用程序提供更好的日誌質量和控制，因為除了包括任何其他所需的詳細信息外，您還可以自定義日誌輸出格式、分類和分離組件輸出到單獨的日誌文件。確保您查看和標準化日誌記錄庫以及組織所需的數據和格式設置，以便更輕鬆地進行分析和處理。

您也可以寫入 CloudWatch 日誌串流 CloudWatch 日誌`PutLogEvents`API 調用或使用AWS開發套件。您可以使用 API 或 SDK 滿足自定義日誌記錄要求，例如，跨一組分佈式組件和服務器將日誌記錄協調到單個日誌流。但是，最容易維護和最廣泛適用的解決方案是將應用程序配置為寫入日誌文件，然後使用 CloudWatch 代理讀取日誌文件並將其流式傳輸到 CloudWatch。

您還應該考慮要從應用程序日誌文件中衡量的指標類型。您可以使用指標篩選器來測量、繪製圖形和警報 CloudWatch 日誌羣組。例如，您可以使用指標篩選器通過在日誌中標識失敗的登錄嘗試來計算失敗的登錄嘗試。

您還可以為自定義開發的應用程序創建自定義指標，方法是使用CloudWatch 內嵌指標格式在應用程序日誌文件中。

設定應用程式指標

自定義指標是指不直接由AWS的服務 CloudWatch ，並且它們發佈在 CloudWatch 指標。所有應用程序指標被視為自定義 CloudWatch 指標。應用程序指標可能與 EC2 實例、應用程序組件、API 調用甚至業務函數相對應。您還必須考慮為指標選擇的維度的重要性和基本性。基數較高的維度會生成大量的自定義指標，並且可能會增加 CloudWatch 成本：

CloudWatch 幫助您通過多種方式捕獲應用程序級指標，包括以下方式：

- 通過定義要從常規程式外包。
- 應用程序將指標發佈到 Windows 性能監視器，並且此度量在 CloudWatch 組態。
- 指標篩選器和模式會針對 CloudWatch 中的應用程序的日誌應用。
- 應用程序寫入 CloudWatch 日誌中使用 CloudWatch 內嵌指標格式。

- 應用程序將指標發送到 CloudWatch 或AWS開發套件。
- 應用程序將指標發送到集合或者StatsD守護進程配置的 CloudWatch 代理程式。

通過 CloudWatch 代理，您可以使用 prostat 監視和測量關鍵應用程序進程。如果您的應用程序不再運行關鍵進程，這有助於您發出警報並採取措施（例如，通知或重新啟動過程）。您還可以測量應用程序進程的性能特徵，並在特定進程運作異常時發出警報。

如果您無法使用其他自定義指標更新 COTS 應用程序，Procstat 監控也非常有用。例如，您可以建立`my_process`度量，用於測量`cpu_time`並且包含一個自定義的`application_version`維度。您也可以使用 CloudWatch 代理配置文件（如果您針對不同的指標具有不同的維度）。

如果您的應用程序在 Windows 上運行，則應評估它是否已將指標發佈到 Windows 性能監視器。許多 COTS 應用程序與 Windows 性能監視器集成，可幫助您輕鬆監控應用程序指標。 CloudWatch 還與 Windows 性能監視器集成，您可以捕獲其中已經可用的任何指標。

確保您查看應用程序提供的日誌記錄格式和日誌信息，以確定可以使用指標篩選器提取哪些指標。您可以查看應用程序的歷史日誌，以確定如何表示錯誤消息和異常關閉。您還應查看以前報告的問題，以確定是否可以捕獲量度以防止問題再次發生。您還應查看應用程序的文檔，並要求應用程序開發人員確認如何識別錯誤消息。

對於自定義開發的應用程序，請與應用程序的開發人員合作，定義重要的指標，這些指標可以通過使用 CloudWatch 內嵌指標格式AWS或AWSAPI。建議的方法是使用嵌入指標格式。您可以使用AWS提供了開源嵌入式指標格式庫，以幫助您以所需格式編寫語句。您還需要將特定於的應用程式 CloudWatch 配置以包括嵌入指標格式代理程式代理程式。這會導致在 EC2 實例上運行的代理充當本地嵌入式指標格式終端節點，該終端節點將嵌入式指標格式指標發送到 CloudWatch。

如果您的應用程序已支持將指標發佈到 collectd 或 statsd，您可以利用這些指標將指標引入到 CloudWatch 中。

適用於 Amazon EC2 和本地服務器的 CloudWatch 代理安裝方法

自動化 CloudWatch 代理的安裝過程可幫助您快速、一致地部署它並捕獲所需的日誌和指標。有幾種方法可以自動安裝 CloudWatch 代理，包括多帳户和多區域支持。討論了以下自動化安裝方法：

- 安裝 CloudWatch 代理使用 Systems Manager 分發商和 Systems Manager 狀態管理器— 如果 EC2 執行個體和內部部署伺服器正在運行 Systems Manager 代理程式，我們建議您使用此方法。這可確保 CloudWatch 代理保持更新，您可以報告並修復沒有 CloudWatch 代理程式。此方法還可擴展以支持多個賬户和區域。
- 部署 CloudWatch 代理作為 EC2 實例置備期間用户數據腳本的一部分— Amazon EC2 允許您定義首次引導或重新啟動時運行的啟動腳本。您可以定義腳本以自動執行代理的下載和安裝過程。這也可以包含在AWS CloudFormation和AWSService Catalog 產品。如果針對不符合標準的特定工作負載採用了自定義的代理安裝和配置方法，則此方法可能會根據需要適用。
- 在亞馬遜計算機映像 (AMI) 中包括 CloudWatch 代理程式— 您可以在適用於 Amazon EC2 的自定義 AMI 中安裝 CloudWatch 代理。使用 AMI 的 EC2 實例將自動安裝並啟動代理。但是，您必須確保代理及其配置定期更新。

安裝 CloudWatch 代理程式使用 Systems Manager 分發商和狀態管理器

您可以將 Systems Manager 狀態管理器狀態管理器與 Systems Manager 分發服務器自動安裝和更新 CloudWatch 服務器和 EC2 實例上的代理。經銷商包括`AmazonCloudWatchAgent` AWS託管軟件包，可安裝最新的 CloudWatch 代理版本。

此安裝方法有以下事前準備：

- 必須在您的伺服器或 EC2 執行個體上安裝並執行 Systems Manager 代理程式。Systems Manager 代理已預安裝在亞馬遜 Linux、亞馬遜 Linux 2 和某些 AMI 上。還必須在其他映像或本地 VM 和服務器上安裝和配置代理。
- 一個或多個 IAM 角色，具有必需的 CloudWatch 和 Systems Manager 權限必須附加到 EC2 執行個體或在內部部署伺服器的證書文件中定義。例如，您可以創建 IAM 角色，其中包含AWS受管政策：`AmazonSSMManagedInstanceCore`針對 Systems Manager 和`CloudWatchAgentServerPolicy`適用於 CloudWatch。您可以使用服務器-雲監視實例角

色 .yaml AWS CloudFormation模板部署 IAM 角色和執行個體配置文件，其中包含這兩個策略。此模板也可以修改為包含 EC2 實例的其他標準 IAM 權限。對於內部部署伺服器或 VM，應配置 CloudWatch 代理程式使用Systems Manager 服務角色，這是為內部部署伺服器設定的。如需此項目的詳細資訊，請參如何配置使用 Systems Manager 代理和統一 CloudWatch 代理只使用臨時證書？中的AWS知識中心.

下面的列表提供了使用 Systems Manager 分發服務器和狀態管理器方法來安裝和維護 CloudWatch 代理程式：

- 自動安裝多個操作系統— 您無需為每個操作系統編寫和維護腳本即可下載和安裝 CloudWatch 代理。
- 自動更新檢查— 狀態管理器會自動定期檢查每個 EC2 實例是否具有最新的 CloudWatch 版本。
- 合規性報告— Systems Manager 合規性儀錶板顯示哪些 EC2 實例未能成功安裝分銷商軟件包。
- 新啟動的 EC2 實例的自動安裝— 在您的賬户中啟動的新 EC2 實例會自動接收 CloudWatch 代理程式。

但是，在選擇此方法之前，還應考慮以下三個方面：

- 與現有關聯的衝突— 如果另一個關聯已安裝或配置 CloudWatch 代理，那麼這兩個關聯可能會互相幹擾並可能導致問題。使用此方法時，應刪除安裝或更新 CloudWatch 代理和配置的所有現有關聯。
- 更新定製代理程式組態檔案— 分發服務器使用默認配置文件執行安裝。如果您使用自定義配置文件或多個 CloudWatch 配置文件，則必須在安裝後更新配置。
- 多區域或多賬户設置— 必須在每個帳户和地區中設置狀態管理器關聯。必須更新多賬户環境中的新帳户以包含狀態管理器關聯。您需要集中或同步 CloudWatch 配置，以便多個帳户和地區可以檢索和應用您所需的標準。

設置狀態管理器和分銷商 CloudWatch 代理程式部署和組態

您可以使用Systems Manager 快速設定快速配置 Systems Manager 功能，包括自動安裝和更新 CloudWatch 代理。快速安裝程序會部署AWS CloudFormation堆棧，根據您的選擇部署和配置 Systems Manager 資源。

下面的列表提供了兩個重要的操作，這兩個操作由快速安裝為自動執行 CloudWatch 代理安裝和更新：

1. 創建 Systems Manager 自定義文檔— 快速安裝創建以下 Systems Manager 文檔以與狀態管理器一起使用。文檔名稱可能會有所不同，但內容保持不變：

- `CreateAndAttachIAMToInstance`— 建立`AmazonSSMRoleForInstancesQuickSetup`角色和實例配置文件（如果它們不存在）並附加`AmazonSSMManagedInstanceCore`政策添加到角色。這不包括所需的`CloudWatchAgentServerPolicy`IAM 政策。您必須更新此策略並更新此 Systems Manager 文檔以包含此策略，如以下部分所述。
- `InstallAndManageCloudWatchDocument`— 安裝 CloudWatch 代理，並使用默認的 CloudWatch 使用代理程式組態`AWS-ConfigureAWSPackage`Systems Manager 文件。
- `UpdateCloudWatchDocument`— 更新 CloudWatch 代理程式安裝最新的 CloudWatch 代理程式，使用`AWS-ConfigureAWSPackage`Systems Manager 文件。更新或卸載代理不會刪除現有 CloudWatch 來自 EC2 執行個體的組態檔案。

2. 建立狀態管理員關聯— 創建狀態管理器關聯並將其配置為使用自定義創建的 Systems Manager 文檔。狀態管理器關聯名稱可能會有所不同，但配置保持不變：
   - `ManageCloudWatchAgent`— 執行`InstallAndManageCloudWatchDocument`Systems Manager 為每個 EC2 實例提供一次文檔。
   - `UpdateCloudWatchAgent`— 執行`UpdateCloudWatchDocument`Systems Manager 每 30 天為每個 EC2 實例提供一次文檔。
   - 執行`CreateAndAttachIAMToInstance`Systems Manager 為每個 EC2 實例提供一次文檔。

您必須增強和自定義已完成的快速安裝配置，以包含 CloudWatch 權限並支持自定義 CloudWatch 配置。尤其
是，`CreateAndAttachIAMToInstance`與`InstallAndManageCloudWatchDocument`文檔將需要更新。您可以手動更新由「快速設定」創建的 Systems Manager 文件。或者，您可以使用自己的 CloudFormation 模板以使用必要的更新置備相同的資源，以及配置和部署其他 Systems Manager 資源，而不是使用快速設置。

> **Important**
>
> 快速設定會創建AWS CloudFormation堆棧根據您的選擇部署和配置 Systems Manager 資源。如果您更新了快速設置選項，則可能需要手動重新更新 Systems Manager 文檔。

以下各節介紹如何手動更新由快速安裝創建的 Systems Manager 資源，以及如何使用您自己的AWS CloudFormation模板來執行更新的快速設置。建議您使用自己的AWS CloudFormation模板，以避免手動更新由快速安裝和AWS CloudFormation。

使用 Systems Manager 快速設置並手動更新創建的 Systems Manager 資源

必須更新快速設置方法創建的 Systems Manager 資源，以包括所需的 CloudWatch 代理權限並支持多個 CloudWatch 組態檔案。本節介紹如何更新 IAM 角色和 Systems Manager 文檔以使用包含 CloudWatch 配置，可從多個帳户訪問。建立 S3 儲存貯體來存放 CloudWatch 設定文件會在管理 CloudWatch組態章節。

更新**`CreateAndAttachIAMToInstance`**Systems Manager 文件

由快速安裝創建的此 Systems Manager 文檔檢查 EC2 實例是否附加了現有 IAM 實例配置文件。如果這樣做，它會附加`AmazonSSMManagedInstanceCore`策略設定為現有角色。這樣可以防止您的現有 EC2 執行個體丟失AWS權限，這些權限可以通過現有實例配置文件分配。您需要在本文檔中添加一個步驟，以附加`CloudWatchAgentServerPolicy`IAM 策略添加到已附加實例配置文件的 EC2 實例。如果 IAM 角色不存在且 EC2 實例沒有附加實例配置文件，則 Systems Manager 文檔還會創建該 IAM 角色。您必須更新文檔的此部分，以便還包含`CloudWatchAgentServerPolicy`IAM 政策。

檢已完成創建和立町實例 .Yaml示例文檔，並將其與快速設置創建的文檔進行比較。編輯現有文檔以包含所需步驟和更改。根據您的「快速設置」選項，「快速設置」創建的文檔可能與提供的示例文檔不同，因此請確保進行所需的調整。示例文檔包括「快速安裝」選項選項，用於每天掃描缺少的修補程序的實例，因此包括 Systems Manager 修補程序管理器的策略。

更新**`InstallAndManageCloudWatchDocument`**Systems Manager 文件

由快速安裝創建的此 Systems Manager 文檔將安裝 CloudWatch 代理，並使用默認 CloudWatch 代理程式組態。預設的 CloudWatch 配置與基本的預定義指標集對齊。您必須替換默認配置步驟並添加步驟以下載 CloudWatch 配置文件 CloudWatch 設定 S3 儲存貯體。

檢已完成安裝和管理觀察文檔 .Yaml更新文檔，並將其與快速設置創建的文檔進行比較。快速設置創建的文檔可能不同，因此請確保您已進行了所需的調整。編輯現有文檔以包含必要的步驟和更改。

使用AWS CloudFormation而不是快速設定

您可以使用AWS CloudFormation配置 Systems Manager。此方法允許您根據您的特定需求定製 Systems Manager 配置。此方法還可以避免手動更新由快速安裝程序創建的已配置的 Systems Manager 資源，以支持自定義 CloudWatch 配置。

快速設置功能還使用AWS CloudFormation並創建一個AWS CloudFormation堆棧設置為根據您的選擇部署和配置 Systems Manager 資源。在您可以使用AWS CloudFormation堆棧集，您必須創建AWS CloudFormation StackSets 以支持跨多個帳户或區域的部署。快速設置可創建支持多區域或多賬户部

署所需的角色。AWS CloudFormationStackSets。您必須完成AWS CloudFormation StackSets 如果您想要在多個區域或從單個帳户和地區的多個帳户配置和部署 Systems Manager 資源。如需此項目的詳細資訊，請參堆疊集操作的先決條件中的AWS CloudFormation文件中)。

請參AWS-快速設置-網絡主機管理。Yaml AWS CloudFormation模板來定製快速設定。

您應該查看AWS CloudFormation模板，並根據您的需求進行調整。您應該使用版本控制AWS CloudFormation模板，並以遞增方式測試更改以確認所需結果。此外，您還應執行雲安全審核，以確定是否存在根據組織的要求進行的任何策略調整。

您應該部署AWS CloudFormation堆棧在單個測試帳户和區域中，並執行任何必要的測試用例來自定義和確認所需的結果。然後，您可以將部署完成到單個帳户中的多個區域，然後到多個帳户和多個區域。

在單一帳號和區域中進行定製快速設定AWS CloudFormation堆

如果您只使用單個帳户和區域，則可以將完整的示例部署為AWS CloudFormation堆疊，而不是AWS CloudFormation堆疊集。但是，如果可能，我們建議您使用多賬户、多區域堆棧設置方法，即使只使用單個帳户和地區。使用AWS CloudFormation StackSets 可以更輕鬆地 future 擴展到其他賬户和地區。

使用下列步驟來部署AWS-快速設置-網絡主機管理。Yaml AWS CloudFormation模板作為AWS CloudFormation在單一帳户和區域中堆棧：

1. 下載模板並將其檢入到您首選的版本控制系統（例如AWS CodeCommit。
2. 自定義默認AWS CloudFormation參數值基於組織的要求。
3. 定製狀態管理器關聯計劃。
4. 定製 Systems Manager 文件，並使用`InstallAndManageCloudWatchDocument`邏輯 ID。確認 S3 存儲桶前綴與包含 CloudWatch 組態。
5. 檢索並記錄 Amazon Cource Name (ARN)，其中包含 CloudWatch 配置。如需此項目的詳細資訊，請參管理 CloudWatch組態章節。範例雲監視配置 S3 桶 .yaml AWS CloudFormation模板可用，該模板包含一個存儲桶策略，用於提供AWS Organizations帳户。
6. 部署自定義的快速設置AWS CloudFormation模板添加到 S3 存儲段相同的賬户：
   - 對於`CloudWatchConfigBucketARN`參數中輸入 S3 儲存貯體的 ARN。
   - 根據要為 Systems Manager 啟用的功能，對參數選項進行調整。
7. 部署帶有或不帶 IAM 角色的測試 EC2 實例，以確認 EC2 實例與 CloudWatch 配合使用。

- 套用AttachIAMToInstance狀態管理員關聯。這是一個 Systems Manager Runbook，配置為按計劃執行。使用 Runbook 的狀態管理器關聯不會自動應用於新的 EC2 實例，並且可以配置為按計劃運行。如需詳細資訊，請參閱「」使用狀態管理器執行觸發執行自動化在 Systems Manager 文件中。
- 確認 EC2 實例附加了所需的 IAM 角色。
- 通過確認 EC2 實例在 Systems Manager 中可見，確認 Systems Manager 代理工作正常。
- 確認 CloudWatch 代理正常工作，方法是查看 CloudWatch 日誌和基於 CloudWatch 來自 S3 儲存貯體的組態。

在多個區域和多個帳户中進行定製快速設定AWS CloudFormationStackSets

如果您使用的是多個帳户和區域，則可以部署AWS-快速設置-網絡主機管理。Yaml AWS CloudFormation模板作為堆棧集。您必須完成AWS CloudFormationStackSet 先決條件，然後再使用堆棧集。根據您是否使用自我管理或者服務管理permissions。

我們建議您部署具有服務管理權限的堆棧集，以便新帳户自動接收自定義的快速安裝程序。您必須部署一個服務管理堆棧集，從AWS Organizations管理帳户或委派管理員帳户。您應該從具有委派管理員權限的用於自動化的集中式帳户部署堆棧集，而不是AWS Organizations管理帳户。我們還建議您測試堆棧集部署，方法是將測試組織單位 (OU) 定位在一個區域中具有單個或少量帳户數量的測試組織單位 (OU)。

1. 完成步驟 1 到 5，從在單一帳號和區域中進行定製快速設定AWS CloudFormation堆章節。
2. 登入AWS Management Console，開啟AWS CloudFormation控制台，然後選擇建立 StackSet：
   - 選擇範本已就緒和上傳範本檔案。上傳AWS CloudFormation模板，您可以根據您的需求進行定製。
   - 指定堆棧集詳細信息：
     - 輸入堆棧集名稱，例如StackSet-SSM-QuickSetup。
     - 根據要為 Systems Manager 啟用的功能，對參數選項進行調整。
     - 對於CloudWatchConfigBucketARN參數 ARN 輸入 CloudWatch 設定的 S3 儲存貯體。
     - 指定堆棧集選項，選擇是否將服務管理權限與AWS Organizations或自我管理許可。
       - 如果選擇自我管理權限，請輸入Aws雲形式堆棧集管理角色和Aws雲形成堆棧規則角色IAM角色詳細信息。帳户中必須存在管理員角色，並且每個目標帳户中必須存在執行角色
     - 適用於服務管理具有的許可AWS Organizations，我們建議您首先部署到測試 OU，而不是整個組織。

- 選擇您要啟用自動部署。建議您選擇Enabled。對於帳户刪除行為，建議的設置為刪除堆疊。
- 適用於自我管理權限，輸入AWS您要設定的帳户 ID。如果您使用自管理權限，則必須為每個新帳户重複此過程。
- 輸入您將使用的地區 CloudWatch 和 Systems Manager。
- 通過查看操作 和堆疊執行個體選項卡中的堆棧集。
- 測試 Systems Manager 和 CloudWatch 在已部署的帳户中正常工作，請按照在單一帳號和區域中進行定製快速設定AWS CloudFormation堆章節。

配置內部部署伺服器的注意事項

所以此 CloudWatch 代理使用類似於 EC2 實例的方法來安裝和配置。但是，下表提供了在安裝和配置 CloudWatch 在內部部署伺服器和 VM 上安裝代理程式。

指向 CloudWatch 代理設置為用於 Systems Manager 的相同臨時證書。	在包含本地服務器的混合環境中設置 Systems Manager 時，您可以使用 IAM 角色激活 Systems Manager。您應該使用為 EC2 實例創建的角色，該角色包括`CloudWatchAgentServerPolicy` 和`AmazonSSMManagedInstanceCore` 政策。 這會導致 Systems Manager 代理檢索臨時證書並將其寫入本地憑證文件。您可以指出您的 CloudWatch 代理配置添加到同一個文件中。您可以使用將使用 Systems Manager 代理和統一 CloudWatch 代理的本地服務器配置為僅使用臨時憑據中的AWS知識中心. 您還可以通過定義單獨的 Systems Manager 自動化運行簿和狀態管理器關聯，並使用標籤定位您的本地實例，從而自動執行此過程。建立Systems Manager 激活，則應包含一個標籤，將實例標識為本地實例。
考慮使用具有 VPN 或AWS Direct Connect存取和AWSPrivateLink。	您可以使用AWS Direct Connect或者AWS Virtual Private Network(AWS VPN），在內部部署網絡和虛擬私有雲端 (VPC) 之間建立私有

	連接。AWSPrivateLink 接建立一個私有連接 CloudWatch 使用界面 VPC 端點記錄。如果您有限制阻止數據通過公共互聯網發送到公共服務終端節點，則此方法非常有用。
所有指標必須包含在 CloudWatch 組態檔案。	Amazon EC2 包含標準指標（例如 CPU 利用率），但必須為本地實例定義這些指標。您可以使用單獨的平台配置文件為本地服務器定義這些指標，然後將配置附加到標準 CloudWatch 指標配置。

臨時 EC2 實例的注意事項

EC2 實例是臨時實例，或僅有一日生命的，如果它們是 Amazon EC2 Auto Scaling、Amazon EMR、Amazon EC2 競價型實例, 或AWS Batch。臨時 EC2 實例可能會導致非常大量的 CloudWatch 流，而無需關於其運行時源的其他信息。

如果您使用臨時 EC2 實例，請考慮在日誌組和日誌流名稱中添加其他動態上下文信息。例如，您可以包含競價型實例請求 ID、Amazon EMR 集羣名稱或 Auto Scaling 組名稱。對於新啟動的 EC2 實例，此信息可能會有所不同，您可能必須在運行時檢索和配置它。您可以通過編寫 CloudWatch 代理配置文件，並重新啟動代理以包含更新的配置文件。這樣可以使用動態運行時信息將日誌和指標傳遞到 CloudWatch。

您還應確保您的指標和日誌由 CloudWatch 代理，然後再終止臨時 EC2 實例。所以此 CloudWatch 代理包含`flush_interval`參數，可配置為定義刷新日誌和度量緩衝區的時間間隔。您可以根據工作負載降低此值，並停止 CloudWatch 代理，並在 EC2 實例終止之前強制緩衝區刷新。

使用自動化解決方案部署 CloudWatch 代理人

如果您使用自動化解決方案（例如 Ansible 或 Chef），您可以利用它來自動安裝和更新 CloudWatch 代理程式。如果使用此方法，則必須評估以下注意事項：

- 驗證自動化是否涵蓋您支持的操作系統和操作系統版本。如果自動化腳本不支持組織的所有操作系統，則應為不受支持的操作系統定義替代解決方案。
- 驗證自動化解決方案是否定期檢查 CloudWatch 代理更新和升級。您的自動化解決方案應定期檢查 CloudWatch 代理，或定期卸載並重新安裝代理。您可以使用調度程序或自動化解決方案功能定期檢查和更新代理。

- 驗證是否可以確認代理安裝和配置符合性。您的自動化解決方案應使您能夠確定系統何時沒有安裝代理或代理無法正常工作。您可以在自動化解決方案中實施通知或警報，以便跟蹤失敗的安裝和配置。

部署 CloudWatch 代理在實例置備期間使用用户數據腳本

如果您不打算使用 Systems Manager，並且希望有選擇地將 CloudWatch 用於 EC2 實例，則可以使用此方法。通常，這種方法是一次性的，或者在需要專門配置時使用。AWS提供直接鏈接(針對) CloudWatch 代理，可以在啟動腳本或用户數據腳本中下載。代理安裝包可以在無需用户交互的情況下以靜默方式運行，這意味着您可以在自動部署中使用它們。如果使用此方法，則應評估以下注意事項：

- 增加了用户無法安裝代理或配置標準指標的風險。用户可以置備實例而不包括安裝 CloudWatch 代理程式。他們還可能錯誤配置代理，這可能會導致日誌記錄和監視不一致。
- 安裝腳本必須特定於操作系統並適用於不同的操作系統版本。如果您打算同時使用 Windows 和 Linux，則需要單獨的腳本。Linux 腳本還應根據發行版具有不同的安裝步驟。
- 您必須定期更新 CloudWatch 代理的新版本（如果可用）。如果將 Systems Manager 與狀態管理器一起使用，則可以自動執行此操作，但您也可以配置用户數據腳本以在實例啟動時重新運行。所以此 CloudWatch 代理會在每次重新啟動時更新並重新安裝。
- 您必須自動檢索和應用標準 CloudWatch 配置。如果將 Systems Manager 與狀態管理器一起使用，則可以自動執行此操作，但也可以配置用户數據腳本以在引導時檢索配置文件並重新啟動 CloudWatch 代理程式。

包括 CloudWatch AMI 中的代理程式

使用此方法的優點是您不必等待 CloudWatch 代理，您可以立即開始日誌記錄和監視。這有助於您更好地監控實例配置和啟動步驟，以防實例無法啟動。如果您不打算使用 Systems Manager 代理，此方法也適用。如果使用此方法，則應評估以下注意事項：

- 必須存在更新過程，因為 AMI 可能不包含最新的 CloudWatch 代理程式版本。所以此 CloudWatch 代理僅在上次創建 AMI 時為最新時間。您應該包括一個額外的方法，用於定期更新代理以及在預配 EC2 實例時更新代理。如果您使用 Systems Manager，您可以使用安裝 CloudWatch 代理程式使用 Systems Manager 分發商和狀態管理器解決方案。如果您不使用 Systems Manager，則可以在實例啟動和重新啟動時使用用户數據腳本更新代理。
- 您的 CloudWatch 必須在實例啟動時檢索代理配置文件。如果不使用 Systems Manager，則可以配置用户數據腳本以在引導時檢索配置文件，然後重新啟動 CloudWatch 代理程式。
- 所以此 CloudWatch 代理必須在 CloudWatch 設定已更新。

- AWS憑據不能保存在 AMI 中。確保沒有本地AWS憑證會儲存在 AMI 中。如果您使用 Amazon EC2，則可以將必要的 IAM 角色應用於您的實例並避免使用本地證書。如果您使用本地實例，則應在啟動 CloudWatch 代理程式。

在 Amazon ECS 上進行記錄和監控

Amazon Elastic Container Service (Amazon ECS) 為執行容器提供兩種啟動類型，並決定託管任務和服務的基礎設施類型；這些啟動類型是 AWS Fargate 和 Amazon EC2。兩種啟動類型都與整合，CloudWatch 但組態和支援會有所不同。

以下各節可協助您了解如何在 Amazon ECS 上使用 CloudWatch 記錄和監控。

主題

- CloudWatch 使用 EC2 啟動類型進行設定
- 適用於 EC2 和 Fargate 啟動類型的 Amazon ECS 容器日誌
- 使用 Amazon ECS FireLens 的自訂日誌路由
- Amazon ECS 的指標

CloudWatch 使用 EC2 啟動類型進行設定

使用 EC2 啟動類型時，您可以佈建使用 CloudWatch代理程式進行記錄和監控的 EC2 執行個體的 Amazon ECS 叢集。Amazon ECS 最佳化 AMI 已預先安裝在 Amazon ECS 容器代理程式中，並為 Amazon ECS 叢集提供 CloudWatch 指標。

這些預設指標包含在 Amazon ECS 的成本中，但 Amazon ECS 的預設組態不會監控日誌檔或其他指標 (例如，可用磁碟空間)。您可以使用 AWS Management Console 來佈建具有 EC2 啟動類型的 Amazon ECS 叢集，這會建立一個使用啟動組態部署 Amazon EC2 Auto Scaling 群組的 AWS CloudFormation 堆疊。但是，這種方法意味著您無法選擇自訂 AMI 或使用不同的設定或其他開機指令碼自訂啟動組態。

若要監控其他日誌和指標，您必須在 Amazon ECS 容器執行個體上安裝 CloudWatch 代理程式。您可以從本指南的一安裝 CloudWatch 代理程式使用 Systems Manager 分發商和狀態管理器節中使用 EC2 執行個體的安裝方法。不過，Amazon ECS AMI 不包含必要的系統管理員代理程式。建立 Amazon ECS 叢集時，您應該使用自訂啟動組態與安裝系統管理員代理程式的使用者資料指令碼搭配使用。這可讓您的容器執行個體向 Systems Manager 註冊，並套用狀態管理員關聯來安裝、設定和更新 CloudWatch 代理程式。當狀態管理器執行和更新您的 CloudWatch 代理程式組態時，它也會套用 Amazon EC2 的標準化系統層級 CloudWatch 組態。您也可以將 Amazon ECS 的標準化 CloudWatch 組 CloudWatch 態存放在 S3 儲存貯體中，以供您的組態使用狀態管理員自動套用。

您應確保套用至 Amazon ECS 容器執行個體的 IAM 角色或執行個體設定檔包含必要的`CloudWatchAgentServerPolicy`和`AmazonSSMManagedInstanceCore`政策。您可以使用模

板佈建以 Linux 為基礎的 Amazon ECS 叢集。 AWS CloudFormation 此範本建立具有自訂啟動組態的 Amazon ECS 叢集，該叢集會安裝 Systems Manager 並部署自訂 CloudWatch 組態以監控 Amazon ECS 特定的日誌檔。

您應該為 Amazon ECS 容器執行個體以及標準 EC2 執行個體日誌擷取下列日誌：

- Amazon ECS 代理程式啟動輸出 — `/var/log/ecs/ecs-init.log`
- Amazon ECS 代理程式輸出 — `/var/log/ecs/ecs-agent.log`
- IAM 登入資料提供者請求記錄 — `/var/log/ecs/audit.log`

如需有關輸出層級、格式化和其他組態選項的詳細資訊，請參閱 Amazon ECS 文件中的 Amazon ECS 日誌檔案位置。

> **Important**
>
> Fargate 啟動類型不需要代理程式安裝或組態，因為您不執行或管理 EC2 容器執行個體。

Amazon ECS 容器執行個體應該使用最新的 Amazon ECS 最佳化 AMI 和容器代理程式。 AWS 使用 Amazon ECS 最佳化 AMI 資訊 (包括 AMI ID) 儲存公用 Systems Manager 參數存放區參數。您可以使用 Amazon ECS 最佳化 AMI 的參數存放區參數格式，從參數存放區擷取最新最新最佳化的 AMI。您可以參考參照參考範本中最新 AMI 或特定 AMI 發行版 AWS CloudFormation 本的公用參數存放區參數。

AWS 在每個支援的區域中提供相同的參數存放區參數。這表示參考這些參數的 AWS CloudFormation 範本可以跨區域和帳戶重複使用，而不需要更新 AMI。您可以透過參考特定版本來控制組織中較新的 Amazon ECS AMI 的部署，這有助於您在測試之前防止使用新的 Amazon ECS 最佳化 AMI。

適用於 EC2 和 Fargate 啟動類型的 Amazon ECS 容器日誌

Amazon ECS 使用任務定義將容器部署和管理為任務和服務。您可以在任務定義內設定要啟動到 Amazon ECS 叢集的容器。記錄是使用容器層級的記錄驅動程式設定。多個記錄驅動程式選項會根據您使用 EC2 還是 Fargate 啟動類型 `awslogs` `fluentd` `gelf` `json-file` `journald` `logentries` `splunk` `syslog`，為您的容器提供不同的記錄系統 (例如、、、、、、或`awsfirelens`)。Fargate 啟動類型提供下列記錄驅動程式選項的子集：`awslogs`、`splunk`和`awsfirelens`。 AWS 提供記`awslogs`錄驅動程式，以擷取容器輸出並將其傳輸至 CloudWatch 記錄檔。記錄驅動程式設定可讓您自訂記錄群組、區域和記錄串流前置詞以及許多其他選項。

記錄群組的預設命名以及上的「自動設定 CloudWatch 記錄檔」選項使用的 AWS Management Console 選項為。/ecs/<task_name>Amazon ECS 使用的日誌流名稱具有以下<awslogs-stream-prefix>/<container_name>/<task_id>格式。我們建議您使用群組名稱，根據組織的需求將記錄分組。在下表中，image_name和包含image_tag在記錄資料流的名稱中。

記錄群組名稱	/<Business unit>/<Project or application name>/<Environment>/<Cluster name>/<Task name>
記錄串流名稱前置詞	/<image_name>/<image_tag>

此資訊也可在任務定義中找到。但是，任務會定期更新為新的修訂版本，這意味著任務定義使用的可能image_name與image_tag任務定義目前使用的不同，也可能使用的版本不同。如需詳細資訊和命名建議，請參規劃您的 CloudWatch 部署閱本指南章節。

如果您使用持續整合和持續傳遞 (CI/CD) 管線或自動化程序，您可以透過每個新的 Docker 映像組建，為應用程式建立新的任務定義修訂版本。例如，您可以將 Docker 映像名稱、影像標籤、 GitHub修訂或其他重要資訊納入任務定義修訂版本和記錄組態中，作為 CI/CD 程序的一部分。

使用 Amazon ECS FireLens 的自訂日誌路由

FireLens Amazon ECS 可協助您將日誌路由到 Fluentd 或 Flu ent Bit，以便您可以將容器日誌直接傳送到 AWS 服務和 AWS 合作夥伴網路 (APN) 目的地，以及支援日誌傳送至日誌。 CloudWatch

AWS 使用 Amazon Kinesis 資料串流、亞馬遜資料 Firehose 和日誌的預先安裝外掛程式，為 Fluent Bit 提供 Docker 影像。 CloudWatch 您可以使用記 FireLens 錄驅動程式而非記awslogs錄驅動程式來進行更多自訂和控制傳送至記錄檔的 CloudWatch 記錄檔。

例如，您可以使用 FireLens 記錄驅動程式來控制記錄格式輸出。這表示 Amazon ECS 容器的 CloudWatch 日誌會自動格式化為 JSON 物件，並包含ecs_cluster、、、ecs_task_arnecs_task_definitioncontainer_id、container_name和的 JSON 格式屬性。ec2_instance_id當您指定驅動程式時，流暢的主機會透過FLUENT_HOST和FLUENT_PORT環境變數公開給您的容awsfirelens器。這意味著您可以使用流利的記錄器庫從代碼直接登錄到日誌路由器。例如，您的應用程式可能會包含要記錄到 Fluent Bit 的程式fluent-logger-python庫，方法是使用環境變數提供的可用值。

如果您選擇用 FireLens 於 Amazon ECS，您可以設定與awslogs日誌驅動程式相同的設定，並使用其他設定。例如，您可以使用 ecs-task-nginx-firelense.json Amazon ECS 任務定義來啟動設定用

FireLens 於記錄的 NGINX 伺服器。 CloudWatch它還啟動了一個 FireLens Fluent Bit 容器作為日誌記錄的邊車。

Amazon ECS 的指標

Amazon ECS 透過 Amazon ECS 容器代理程式在叢集和服務層級為 EC2 和 Fargate 啟動類型提供標準 CloudWatch 指標 (例如 CPU 和記憶體使用率)。您也可以使用 CloudWatch Container Insights 擷取服務、工作和容器的指標，或使用內嵌指標格式擷取您自己的自訂容器指標。

Container Insights 是一項 CloudWatch 功能，可在叢集、容器執行個體、服務和工作層級提供 CPU 使用率、記憶體使用率、網路流量和儲存體等指標。Container Insights 也會建立自動儀表板，協助您分析服務和工作，並查看容器層級的平均記憶體或 CPU 使用率。Container Insights 會將自訂指標發佈到`ECS/ContainerInsights`自訂命名空間，供您用於繪製圖形、警示和儀表板。

您可以為每個個別 Amazon ECS 叢集啟用容器洞見，以開啟容器洞察指標。如果您也想要查看容器執行個體層級的指標，可以在 Amazon ECS 叢集上以精靈容器的身分啟動 CloudWatch 代理程式。您可以使用 cwagent-ecs-instance-metric-cfn.yaml AWS CloudFormation 範本將 CloudWatch 代理程式部署為 Amazon ECS 服務。重要的是，此範例假設您建立了適當的自訂 CloudWatch代理程式組態，並使用金鑰將其儲存在參數存放區中`ecs-cwagent-daemon-service`。

部署為容器深入解析的協助程式 CloudWatch 容器的CloudWatch代理程式包括其他磁碟、記憶體`instance_cpu_reserved_capacity`和 `instance_memory_reserved_capacity` CPU 度量，例如維度和`ClusterName``ContainerInstanceId`、`InstanceId`維度。容器執行個體層級的指標是由容器深入解析使用 CloudWatch 內嵌指標格式來實作。您可以使用本指南各設置狀態管理器和分銷商 CloudWatch 代理程式部署和組態節中的方法，為 Amazon ECS 容器執行個體設定其他系統層級指標。

在 Amazon ECS 中建立自訂應用程式指標

您可以使用CloudWatch內嵌指標格式，為應用程式建立自訂指標。`awslogs`記錄驅動程式可以解譯 CloudWatch 內嵌的指標格式陳述式。

以下範例中的`CW_CONFIG_CONTENT`環境變數設定為「`cwagentconfig`Systems Manager 參數存放區」參數的內容。您可以使用此基本組態執行代理程式，將其設定為內嵌度量格式端點。但是，它不再是必要的。

```
{
"logs": {
```

```
    "metrics_collected": {
      "emf": { }
    }
  }
}
```

如果您有跨多個帳戶和區域的 Amazon ECS 部署，則可以使用 AWS Secrets Manager 秘密存放組 CloudWatch 態，並設定秘密政策以與組織共用。您可以使用任務定義中的祕密選項來設定`CW_CONFIG_CONTENT`變數。

您可以在應用程式中使用 AWS 提供的開放原始碼內嵌指標格式程式庫，並指定`AWS_EMF_AGENT_ENDPOINT`環境變數，以連接到 CloudWatch 代理程式附屬容器，做為內嵌度量格式端點。例如，您可以使用 ecs_cw_emf_example 範例 Python 應用程式，將內嵌度量格式的度量傳送至設定為內嵌度量格式端點的 CloudWatch 代理程式附屬容器。

的 Fluent Bit 外掛程式也 CloudWatch 可用於傳送內嵌度量格式訊息。您也可以使用 ecs_firelense_emf_example 範例 Python 應用程式，將內嵌度量格式的指標傳送到適用於 Amazon ECS 附屬容器的防火鏡。

如果您不想使用內嵌指標格式，可以透過 AWS API 或 AWS SDK 建立和更新 CloudWatch 指標。除非您有特定的使用案例，否則我們不建議使用這種方法，因為它會增加程式碼的維護和管理額外負荷。

Amazon EKS 上的記錄和監控

Amazon Elastic Kubernetes Service (Amazon EKS) 與 CloudWatch 庫伯奈特斯控制飛機的日誌。控制面由 Amazon EKS 作為託管服務提供，您可以打開日誌記錄而不安裝 CloudWatch 代理。所以此 CloudWatch 代理來捕獲 Amazon EKS 節點和容器日誌。Fluent Bit 和 Fluentd也支持將容器日誌發送到 CloudWatch 日誌。

CloudWatch Container Insights 可透過這些資料，在叢集、節點、Pod、任務和服務層級提供針對 Amazon EKS 的綜合指標監控解決方案。亞馬遜 EKS 還支持多個選項，用於使用Prometheus。Amazon EKS 控制平面提供度量終端節點，以 Prometheus 格式顯示指標。您可以將 Prometheus 部署到您的 Amazon EKS 集羣中以使用這些指標。

您也可以設定 CloudWatch 代理程式以湊集 Prometheus 指標並建立 CloudWatch 指標，以及消耗其他 Prometheus 終端節點。Prometheus 的容器洞察監控還可以從受支持的容器化工作負載和系統中自動發現和捕獲 Prometheus 指標。

您可以安裝和設定 CloudWatch 代理的方法類似於 Amazon EC2 與分銷商和狀態管理器使用的方法，以使您的 Amazon EKS 節點與標準系統日誌記錄和監控配置保持一致。

Amazon EKS 的記錄

Kubernetes 日誌記錄可分為控制平面日誌記錄、節點日誌記錄和應用程序日誌記錄。所以此Kubernetes 控制平面是一組組件，用於管理 Kubernetes 羣集並生成用於審核和診斷目的的日誌。使用亞馬遜 EKS，您可以打開不同控制平面組件的日誌並將其發送到 CloudWatch。

Kubernetes 還運行系統組件，如`kubelet`和`kube-proxy`在運行窗格的每個 Kubernetes 節點上。這些組件在每個節點中寫入日誌，您可以配置 CloudWatch 和容器見解來捕獲每個 Amazon EKS 節點的這些日誌。

容器被分組為豆莢，並計劃在您的 Kubernetes 節點上運行。大多數容器化應用程序寫入標準輸出和標準錯誤，容器引擎會將輸出重定向到日誌記錄驅動程序。在庫伯內提斯，容器日誌可以在`/var/log/pods`目錄。您可以設定 CloudWatch 和容器見解，為您的每個 Amazon EKS 容器捕獲這些日誌。

Amazon EKS 控制平面記錄

Amazon EKS 羣集由 Kubernetes 羣集和運行容器的 Amazon EKS 節點的高可用性單租户控制平面組成。控制平面節點在由AWS。Amazon EKS 叢集控制平面節點與 CloudWatch ，您可以打開特定控制平面組件的日誌記錄。

為每個 Kubernetes 控制平面組件實例提供日誌。AWS管理控制平面節點的運行狀況，並提供Kubernetes 端點的服務級別協議 (SLA)。

Amazon EKS 節點和應用程式記錄

建議您使用CloudWatch Container Insights捕獲亞馬遜 EKS 的日誌和指標。容器見解實現集羣、節點和容器級別的指標，並使用 CloudWatch 代理，以及用於日誌捕獲到 CloudWatch 的流利位或 Fluentd。Container Insights 也提供自動儀表板，其中包含捕獲的分層視圖 CloudWatch 指標。容器見解部署為 CloudWatch DaemonSet 和流利位 DaemonSet ，在每個 Amazon EKS 節點上運行。容器見解不支持 Fargate 節點，因為這些節點由AWS並且不支持 DaemonSets。本指南中單獨介紹了亞馬遜 EKS 的 Fargate 日誌記錄。

下表顯示 CloudWatch 日誌組和日誌默認 Fluentd 或 Fluent Bit 日誌捕獲設定適用於 Amazon EKS 的。

`/aws/containerinsights/Cluster_Name/application`	中所有日誌檔`/var/log/containers` 。此目錄提供了符號鏈接，指向`/var/log/pods` 目錄結構。這將捕獲寫入到`stdout`或者`stderr`。它還包括 Kubernetes 系統容器的日誌，如`aws-vpc-cni-init` 、`kube-proxy` ，以及`coreDNS`。
`/aws/containerinsights/Cluster_Name/host`	來自 的日誌`/var/log/dmesg` 、`/var/log/secure` ，以及`/var/log/messages` 。
`/aws/containerinsights/Cluster_Name/dataplane`	`/var/log/journal` 中適用於`kubelet.service` 、`kubeproxy.service` 和 `docker.service` 的日誌。

如果您不想將容器見解與 Fluent Bit 或 Fluentd 結合使用日誌記錄，則可以使用 CloudWatch Amazon EKS 節點上安裝的代理程式。Amazon EKS 節點是 EC2 實例，這意味着您應將它們包含在 Amazon EC2 的標準系統級日誌記錄方法中。如果您安裝 CloudWatch 代理，則 Amazon EKS 節點也包含在 CloudWatch 代理安裝、配置和更新。

下表顯示了特定於 Kubernetes 的日誌，如果您不使用帶 Fluent 位或 Fluentd 的容器見解進行日誌記錄，則必須捕獲這些日誌。

`/var/log/containers`	此目錄提供了符號鏈接到所有 Kubernetes 容器日誌的`/var/log/pods` 目錄結構。這樣可以有效地捕獲寫入`stdout`或者`stderr`。這包括 Kubernetes 系統容器的日誌，例如`aws-vpc-cni-init` 、`kube-proxy` ，以及`coreDNS`。重要：如果您正在使用容器見解，則不需要此操作。

`var/log/aws-routed-eni/ipamd.log`
 	
`/var/log/aws-routed-eni/plugin.log`
	L-IPAM 守護程序的日誌可以在這裏找到

您必須確保 Amazon EKS 節點安裝並配置 CloudWatch 代理以發送相應的系統級日誌和度量。不過，Amazon EKS 優化的 AMI 不包含 Systems Manager 代理程式。通過使用啟動範本，您可以自動執行 Systems Manager 代理安裝和默認 CloudWatch 配置，通過用户數據部分實現的啟動腳本捕獲重要的 Amazon EKS 特定日誌。Amazon EKS 節點使用 Auto Scaling 組作為受管節點組或作為自我管理的節點。

對於託管節點組，您可以提供啟動範本，其中包括用户數據部分，以自動執行 Systems Manager 代理安裝和 CloudWatch 組態。您可以自定義和使用亞馬遜網站 _ 管理 _ 節點 _ 組啟動 _ 配置 AWS CloudFormation模板創建安裝 Systems Manager 代理的啟動模板 CloudWatch 代理，還將 Amazon EKS 特定的日誌記錄配置添加到 CloudWatch 設定目錄。此模板可用於將您的 Amazon EKS 託管節點組啟動模板更新為 infrastructure-as-code (國際諮詢委員會) 辦法. 每次更新AWS CloudFormation範本預配啟動範本的新版本。您可以更新節點組，使用新的範本並使用受管生命週期過程更新節點而不會停機。確保應用於託管節點組的 IAM 角色和實例配置文件包含`CloudWatchAgentServerPolicy`和`AmazonSSMManagedInstanceCore` AWS受管政策。

藉助自我管理的節點，您可以直接配置和管理 Amazon EKS 節點的生命週期和更新策略。自我管理的節點允許您在亞馬遜 EKS 羣集上運行 Windows 節點，Bottlerocket，以及其他選項。您可以使用AWS CloudFormation將自我管理的節點部署到您的 Amazon EKS 集羣中，這意味着您可以對 Amazon EKS 羣集使用 iAC 和託管變更方法。AWS提供亞馬遜鏈接點組 AWS CloudFormation模板，您可以依原狀使用或自訂。該模板為集羣中的 Amazon EKS 節點預置所有必需的資源（例如，單獨的 IAM 角色、安全組、Amazon EC2 Auto Scaling 組和啟動模板）。所以此亞馬遜鏈接點組 AWS CloudFormation模板

是一個更新版本，用於安裝所需的 Systems Manager 代理 CloudWatch 代理，還將 Amazon EKS 特定的日誌記錄配置添加到 CloudWatch 設定目錄。

登錄亞馬遜 EKS 在 Fargate

使用 Fargate 上的 Amazon EKS，您可以部署容器，而無需分配或管理您的 Kubernetes 節點。這樣就不需要為 Kubernetes 節點捕獲系統級日誌。要從 Fargate 窗格中捕獲日誌，您可以使用流利位將日誌直接轉發到 CloudWatch。這使您能夠自動將日誌路由到 CloudWatch 而無需進一步配置，也無需在 Fargate 上為您的亞馬遜 EKS 艙提供側車容器。如需此項目的詳細資訊，請參Fargate 記錄和 Amazon EKS 文件中的亞馬遜 EKS 的流利位在AWS部落格。此解決方案捕獲STDOUT和STDERR輸入/輸出 (I/O) 串流並將其發送至 CloudWatch 基於 Fargate 上為 Amazon EKS 羣集建立的流利位配置。

Amazon EKS 和 Kubernetes 的指標

Kubernetes 提供了一個指標 API，允許您訪問資源使用量指標（例如，節點和窗格的 CPU 和內存使用率），但 API 僅提供時間點信息，而不提供歷史指標。所以此庫伯內提斯指標服務器通常用於 Amazon EKS 和 Kubernetes 部署，以聚合指標、提供有關指標的短期歷史信息，以及支持Horizontal Pod Autoscaler。

亞馬遜 EKS 通過 Kubernetes API 服務器公開控制平面指標在 Prometheus 格式和 CloudWatch 可以捕獲和接收這些指標。 CloudWatch 和容器見解還可以配置為您的 Amazon EKS 節點和窗格提供全面的指標捕獲、分析和警報。

Kubernetes 控制平面指標

Kubernetes 公開控制平面指標以 Prometheus 格式通過使用`/metrics`HTTP API 終端節點。您應該安裝Prometheus，以使用 Web 瀏覽器繪製和查看這些指標。您也可以攝取公開的指標通過該 API 服務器傳輸到 CloudWatch 中。

Kubernetes 的節點和系統衡量指標

庫伯內提斯提供的 Prometheus指標-伺服器窗格，您可以部署和運行以獲取羣集、節點和容器級 CPU 和內存統計信息。這些指標與Horizontal Pod Autoscaler和Vertical Pod Autoscaler。 CloudWatch 也可以提供這些指標。

您應該安裝 Kubernetes 度量服務器，如果您使用Kubernetes 儀表板或水平和垂直容器自動縮放器。Kubernetes 儀錶板可幫助您瀏覽和配置 Kubernetes 羣集、節點、窗格和相關配置，並從 Kubernetes 度量服務器查看 CPU 和內存衡量指標。您可以按照部署 Kubernetes 儀表板在 Amazon EKS 文件中。

Kubernetes 度量服務器提供的指標不能用於非自動擴展目的（例如，監視）。這些指標適用於 point-in-time 分析而不是歷史分析. Kubernetes 儀錶板會部署`dashboard-metrics-scraper`以在短時間窗口內存儲來自 Kubernetes 度量服務器的度量。

容器見解使用容器化版本的 CloudWatch 在庫伯內提斯運行的代理 DaemonSet 查找集羣中所有正在運行的容器，並提供節點級別的指標。它可透過這些資料，收集性能堆疊的每一層級的性能資料。您可以使用AWS快速啟動或單獨配置容器見解。Quick Start 會設定指標監控使用 CloudWatch 代理和日誌記錄，因此您只需將其部署一次即可進行日誌記錄和監視。

由於 Amazon EKS 節點是 EC2 實例，因此除了容器洞察捕獲的指標外，您還應使用您為 Amazon EC2 定義的標準捕獲系統級別的指標。您可以使用設置狀態管理器和分銷商 CloudWatch 代理程式部署和組態部分安裝和設定 CloudWatch 適用於 Amazon EKS 羣集的代理程式。您可以更新特定於亞馬遜 EKS 的 CloudWatch 配置文件，以包括指標以及您的亞馬遜 EKS 特定日誌配置。

所以此 CloudWatch 代理可以自動發現和抓取 Prometheus 指標從受支持的容器化工作負載和系統。它攝取它們作為 CloudWatch 日誌以嵌入式度量格式進行分析 CloudWatch 記錄見解並自動創建 CloudWatch 指標。

> **Important**
>
> 您必須部署專用版本的 CloudWatch 代理程式來收集 Prometheus 指標。這是與 CloudWatch 為容器洞見部署的代理程式。您可以使用普羅米修斯示例 Java 應用程序，其中包括 CloudWatch 代理和 Amazon EKS 容器部署，以演示 Prometheus 指標發現。如需詳細資訊，請參閱「」設定適用於 Amazon EKS 和 Kubernetes 的 Java/JMX 範例工作負載在 CloudWatch 文檔中。您也可以設定 CloudWatch 代理從 Amazon EKS 集羣中運行的其他 Prometheus 目標捕獲指標。

應用程式指標

您可以使用CloudWatch 內嵌指標格式。要接收嵌入的指標格式語句，您需要將嵌入的度量格式條目發送到嵌入式度量格式終端節點。所以此 CloudWatch 代理可以配置為您的亞馬遜 EKS 容器中的旁邊集裝箱。所以此 CloudWatch 代理配置存儲為 Kubernetes ConfigMap 並通過您的 CloudWatch 代理側車容器啟動嵌入式指標格式終端節點。

您還可以將您的應用程序設置為 Prometheus 目標，並配置 CloudWatch 代理（Prometheus）支持，以發現、抓取並將您的指標引入到 CloudWatch 中。例如，您可以使用開源 JMX 出口商，以公開 JMX 豆以供 Prometheus 使用 CloudWatch 代理程式。

如果您不想使用嵌入式指標格式，還可以使用AWSAPI或者AWS SDK。不過，我們不建議這樣做，因為它混合了監控和應用程序邏輯。

亞馬遜 EKS 在 Fargate 上的指標

Fargate 會自動配置 Amazon EKS 節點以運行您的 Kubernetes 窗格，因此您無需監控和收集節點級別指標。但是，您必須監控 Fargate 上 Amazon EKS 節點上運行的容器的指標。容器見解目前不適用於 Fargate 上的 Amazon EKS，因為它需要以下當前不支持的功能：

- 目前不支援 DaemonSets。容器見解是通過運行 CloudWatch 代理程式作為 DaemonSet 在每個羣集節點上。
- 不支援 HostPath 永久卷。所以此 CloudWatch 代理容器使用 HostPath 持久卷作為收集容器度量數據的先決條件。
- Fargate 阻止特權容器和訪問主機信息。

您可以使用適用於 Fargate 的內置日誌路由器將嵌入指標格式語句發送到 CloudWatch。日誌路由器使用流利位，該位具有 CloudWatch 插件，可配置為支持嵌入式度量格式語句。

通過在 Amazon EKS 羣集中部署 Prometheus 服務器，從您的 Fargate 節點收集指標，您可以檢索和捕獲 Fargate 節點的容器級別指標。因為 Prometheus 需要持久存儲，如果您使用 Amazon Elastic File System (Amazon EFS) 進行持久存儲，您可以在 Fargate 上部署 Prometheus。您也可以在 Amazon EC2 支持的節點上部署 Prometheus。如需詳細資訊，請參閱「」監控 Amazon EKS 上的 AWS Fargate使用 Prometheus 和 Grafana在AWS部落格。

Amazon EKS 上的 Prometheus 監控

Amazon Managed Service for Prometheus提供了一個可擴展、安全、AWS託管服務的開源 Prometheus。您可以使用 Prometheus 查詢語言 (PromQL) 監控容器化工作負載的性能，而無需管理用於攝取、存儲和查詢操作指標的底層基礎架構。您 Prometheus 通過使用AWS發行版 OpenTelemetry （阿道特）或作為收集代理的 Prometheus 服務器。

CloudWatch Container Insights 監控可讓您設定和使用 CloudWatch 代理來發現來自亞馬遜雲服務器、亞馬遜 EKS 和 Kubernetes 工作負載的 Prometheus 指標，並將其作為 CloudWatch 指標引入。此解決方案適用於 CloudWatch 是您的主要可觀察性和監控解決方案。但是，以下列表概述了針對 Prometheus 的 Amazon 託管服務為接收、存儲和查詢 Prometheus 指標提供了更大的靈活性的使用案例：

- 面向 Prometheus 的 Amazon 託管服務使您能夠使用部署在 Amazon EKS 或自我管理的 Kubernetes 中的現有 Prometheus 服務器，並將其配置為寫入用於 Prometheus 的亞馬遜託管服務，而不是本地配置的數據存儲。這樣就消除了為您的 Prometheus 服務器及其基礎架構管理高可用性數據存儲所帶來的無與倫比的繁重工作。面向 Prometheus 的亞馬遜託管服務是一個合適的選擇，當您有一個成熟的 Prometheus 部署，您希望在AWS雲端。
- Grafana 直接支持 Prometheus 作為可視化的數據源。如果你想使用 Grafana 與 Prometheus，而不是 CloudWatch 用於集裝箱監控的儀錶板，然後面向 Prometheus 的亞馬遜託管服務可以滿足您的要求。面向 Prometheus 的亞馬遜託管服務與亞馬遜管理 Grafana 集成，提供託管開源監控和可視化解決方案。
- Prometheus 使您能夠使用 PromQL 查詢對您的運營指標進行分析。相比之下,該 CloudWatch 代理程式以內嵌指標格式引入 Prometheus 指標到 CloudWatch 導致 CloudWatch 指標。如需內嵌指標格式日誌，可使用 CloudWatch 記錄洞見。
- 如果您不打算使用 CloudWatch 進行監控和指標捕獲，那麼您應該將 Prometheus 的亞馬遜託管服務與 Prometheus 服務器和可視化解決方案（如 Grafana）配合使用。您需要將 Prometheus 服務器配置為從您的 Prometheus 目標抓取指標，並將服務器配置為遠程寫入 Amazon Managed Service of Prometheus。如果您使用亞馬遜管理的 Grafana，則可以通過使用附帶的插件，直接將亞馬遜管理 Grafana 與您的 Prometheus 數據源亞馬遜託管服務集成。由於指標數據存儲在針對 Prometheus 的 Amazon 託管服務中，因此沒有依賴於部署 CloudWatch 代理或要求將數據提取到 CloudWatch 中。所以此 CloudWatch 代理程式進行 Container Insights 監控 Prometheus。

您還可以使用 ADOT 收集器從 Prometheus 測量應用程序中刮取，並將指標發送到 Prometheus 的亞馬遜託管服務。如需 ADOT 收集器的詳細資訊，請參AWSOpen Distro文件中)。

記錄和指標AWS Lambda

拉姆達不再需要管理和監控工作負載的伺服器，並自動使用 CloudWatch指標和 CloudWatch 無需對應用程式程式碼進行進一步設定或檢測的記錄。本節可協助您瞭解 Lambda 所使用之系統的效能特性，以及您的組態選擇如何影響效能。它還可協助您記錄和監控 Lambda 函數，以進行效能最佳化和診斷應用程式層級問題。

Lambda 函數記錄

Lambda 會自動將標準輸出和標準錯誤訊息從 Lambda 函數串流至 CloudWatch 記錄檔，不需要記錄驅動程式。Lambda 也會自動佈建執行 Lambda 函數的容器，並將其設定為在不同的記錄串流中輸出記錄訊息。

Lambda 函數的後續叫用可以重複使用相同的容器，並輸出至相同的記錄串流。Lambda 也可以佈建新的容器，並將呼叫輸出至新的記錄串流。

Lambda 會在第一次叫用 Lambda 函數時自動建立日誌群組。Lambda 函數可以有多個版本，您可以選擇要執行的版本。Lambda 函數叫用的所有記錄都儲存在相同的記錄群組中。名稱無法變更，且位於`/aws/lambda/<YourLambdaFunctionName>`格式。系統會在每個 Lambda 函數執行個體的記錄群組中建立個別的記錄串流。Lambda 有一個標準的命名慣例，用於使用`YYYY/MM/DD/[<FunctionVersion>]<InstanceId>`格式。該`InstanceId`是由AWS以識別 Lambda 函數實例。

我們建議您將日誌消息格式化為 JSON 格式，因為您可以更輕鬆地查詢它們 CloudWatch 日誌見解。它們也可以更容易地過濾和導出。您可以使用日誌庫來簡化此過程或編寫自己的日誌處理函數。我們建議您使用記錄程式庫來協助格式化和分類記錄訊息。例如，如果您的 Lambda 函數是用 Python 編寫的，則可以使用日誌記錄模塊記錄訊息並控制輸出格式。Lambda 原生會針對以 Python 撰寫的 Lambda 函數使用 Python 記錄程式庫，而且您可以在 Lambda 函數中擷取和自訂記錄器。AWS實驗室創造了AWS蟒蛇的拉姆達電動工具開發人員工具組可讓您更輕鬆地利用冷啟動等關鍵資料來豐富記錄訊息。該工具包可用於 Python，Java，打字稿和 .NET。

另一個最佳做法是使用變數來設定記錄輸出層級，並根據環境和您的需求進行調整。除了使用的程式庫之外，Lambda 函數的程式碼還可以根據記錄輸出層級輸出大量記錄資料輸出。這可能會影響您的記錄成本並影響效能。

Lambda 可讓您為 Lambda 函數執行階段環境設定環境變數，而無需更新程式碼。例如，您可以建立`LAMBDA_LOG_LEVEL`環境變數，定義您可以從程式碼擷取的記錄輸出層級。下列範例會嘗試擷

取LAMBDA_LOG_LEVEL環境變量並使用該值來定義日誌記錄輸出。如果未設定環境變數，則預設為INFO水平。

```
import logging
from os import getenv

logger = logging.getLogger()
log_level = getenv("LAMBDA_LOG_LEVEL", "INFO")
level = logging.getLevelName(log_level)
logger.setLevel(level)
```

將記錄檔傳送至其他目的地 CloudWatch

您可以將日誌發送到其他目的地（例如，亞馬遜 OpenSearch 服務或 Lambda 函數) 使用訂閱篩選器。如果你不使用亞馬遜 OpenSearch 服務中，您可以使用 Lambda 函數來處理日誌並將其發送到AWS使用您選擇的服務AWS軟體開發套件。

您也可以將 SDK 用於以外的日誌目的地AWS在 Lambda 函數中使用雲端功能，將記錄陳述式直接傳送至您選擇的目的地。如果選擇此選項，建議您考慮延遲、額外處理時間、錯誤和重試處理，以及操作邏輯耦合至 Lambda 函數的影響。

Lambda 函數指標

Lambda 可讓您執行程式碼，而無需管理或擴展伺服器，這幾乎消除了系統層級稽核和診斷的負擔。不過，瞭解系統層級的 Lambda 函數的效能和叫用指標仍然很重要。這有助於您優化資源配置並提高代碼性能。有效監控和衡量效能可以透過適當調整 Lambda 函數的大小，改善使用者體驗並降低成本。一般而言，以 Lambda 函數執行的工作負載也具有需要擷取和分析的應用程式層級指標。Lambda 直接支援嵌入式指標格式，以擷取應用程式層級 CloudWatch 指標更容易。

系統層級度量

自動整合 CloudWatch 度量標準，並提供了一組您的 Lambda 函數的標準指標。Lambda 還透過這些指標為每個 Lambda 函數提供單獨的監控儀表板。您需要監視的兩個重要指標是錯誤和叫用錯誤。瞭解叫用錯誤與其他錯誤類型之間的差異，可協助您診斷並支援 Lambda 部署。

調用錯誤防止您的 Lambda 函數執行。這些錯誤發生在您的代碼運行之前，因此您無法在代碼中實現錯誤處理來識別它們。相反地，您應該為 Lambda 函數設定警示，以偵測這些錯誤並通知作業和工作

負載擁有者。這些錯誤通常與配置或權限錯誤有關，並且可能由於配置或權限的變更而發生。調用錯誤可能會啟動重試，從而導致多次調用函數。

成功叫用的 Lambda 函數會傳回 HTTP 200 回應，即使函數擲回例外狀況也是如此。您的 Lambda 函數應該實作錯誤處理並引發例外狀況，以便`Errors`指標會擷取並識別 Lambda 函數的失敗執行。您應該從 Lambda 函數叫用傳回格式化的回應，其中包含資訊，以判斷執行是完全、部分還是成功執行失敗。

CloudWatch 提供CloudWatch 拉姆達洞察您可以為個別 Lambda 函數啟用。Lambda 深入解析會收集、彙總和摘要系統層級指標 (例如 CPU 時間、記憶體、磁碟和網路使用量)。Lambda Insights 也會收集、彙總和摘要診斷資訊 (例如冷啟動和 Lambda 工作者關閉)，以協助您隔離並快速解決問題。

Lambda 深入解析會使用內嵌的指標格式，自動將效能資訊發送至`/aws/lambda-insights/`具有根據 Lambda 函數名稱的日誌串流名稱前置詞的日誌群組。這些性能日誌事件創建 CloudWatch 指標是自動的基礎 CloudWatch 儀表板。我們建議您針對效能測試和生產環境啟用 Lambda 深入解析。由 Lambda 洞察建立的其他指標包括`memory_utilization`這有助於正確調整 Lambda 函數的大小，以避免支付不必要的容量費用。

應用程式指標

您也可以在以下位置建立和擷取自己的應用程式指標 CloudWatch 使用內嵌度量格式。您可以利用AWS 提供內嵌指標格式的程式庫建立並發出嵌入式指標格式陳述式 CloudWatch。整合的拉姆達 CloudWatch 記錄功能設定為處理和擷取格式適當的內嵌指標格式陳述式。

搜尋和分析記錄 CloudWatch

將您的日誌和指標擷取為一致的格式和位置後，除了識別和疑難排解問題之外，您還可以搜尋和分析這些記錄檔和指標，以協助提高營運效率。我們建議您以格式正確的格式 (例如 JSON) 擷取記錄，以便更輕鬆地搜尋和分析記錄。大多數工作負載會使用一系列AWS資源，例如網路、運算、儲存和資料庫。在可能的情況下，您應該統一分析這些資源中的指標和日誌，並將其關聯起來，以便有效地監控和管理所有AWS工作負載。

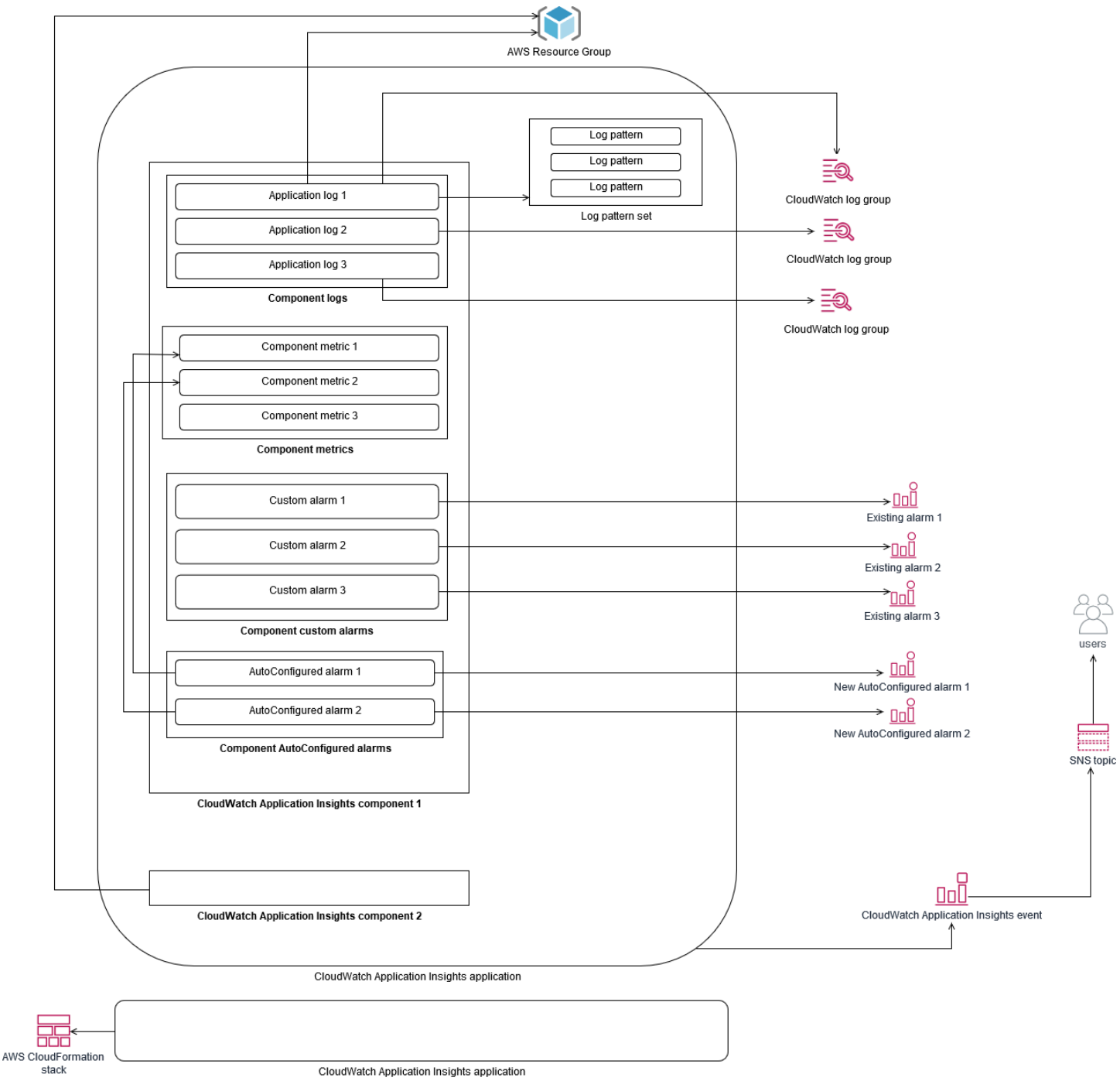
CloudWatch 提供數種功能來協助分析日誌和指標，例如CloudWatch 應用程式洞察 (Application Insights)，以統一定義和監控不同AWS資源的應用程式的指標和記錄；CloudWatch 異常偵測可顯示您的異常狀況指標和 CloudWatch Log Insights，以互動方式搜尋和分析 CloudWatch Logs 中執行的記錄資料。

利 CloudWatch 用應用程式洞察來統一監控和分析應

應用程式擁有者可以使用 Amazon CloudWatch 應用程式洞察來設定工作負載的自動監控和分析。除了針對帳戶中所有工作負載設定的標準系統層級監控之外，還可以設定此功能。透過 CloudWatch 應用程式深入解析設定監控，也可協助應用程式團隊主動調整作業，並縮短平均復原時間 (MTTR)。CloudWatch 應用程式深入解析可協助減少建立應用程式層級記錄和監控所需的工作量。它還提供了基於組件的框架，可幫助團隊劃分日誌記錄和監控責任。

CloudWatch 應用程式深入解析會使用資源群組來識別應統一監視為應用程式的資源。資源群組中支援的資源會變成 CloudWatch 應用程式深入解析應用程式的個別定義元件。 CloudWatch 應用程式見解應用程式的每個元件都有自己的記錄、指標和警示。

對於記錄檔，您可以定義應該用於元件和應用程式深入解析 CloudWatch 應用程式中的記錄檔模式集。記錄模式集合是要根據規則運算式進行搜尋的記錄檔模式集合，以及偵測到病毒碼時的低、中或高嚴重性。對於指標，您可以從服務特定和支援的指標清單中選擇要監視每個元件的指標。針對警示，「 CloudWatch 應用程式深入解析」會針對所監控的指標自動建立和設定標準或異常偵測警示。CloudWatch 應用程式深入解析具有指標的自動組態，以及 CloudWatch 文件中 CloudWatch 應用程式深入解析支援的記錄檔和指標中概述的技術的記錄擷取。下圖顯示 CloudWatch 應用程式深入解析元件及其記錄和監視組態之間的關係。每個組件都定義了自己的日誌和指標，以使用 CloudWatch 日誌和指標進行監視。



由 CloudWatch 應用程式洞察監控的 EC2 執行個體需要 Systems Manager、 CloudWatch 代理程式和許 如需相關資訊，請參閱文件中的使用應用程式深入解析來設定 CloudWatch 應用程式的必要條 CloudWatch 件。 CloudWatch 應用程式洞察使用 Systems Manager 來安裝和更新 CloudWatch 代理程式。在「 CloudWatch應用程式深入解析」中設定的度量和記錄會建立 CloudWatch 代理程式組態檔案，該檔案會儲存在 Systems Manager 參數中，`AmazonCloudWatch-ApplicationInsights-SSMParameter`且每個「 CloudWatch應用程式深入解析」 這會導致一個單獨的 CloudWatch 代理程式組態檔案新增至 EC2 執行個體上的 CloudWatch 代理程式組態目錄。Systems Manager 命令會執

行，將此組態附加至 EC2 執行個體上的作用中組態。使用 CloudWatch 應用程式見解不會影響現有的 CloudWatch 代理程式組態設定 除了自己的系統和 CloudWatch 應用程式層級 CloudWatch 代理程式組態之外，您還可以使用應用程式見解 但是，您應該確保配置不會重疊。

使用日誌見解執行 CloudWatch 日誌分析

CloudWatch Logs Insights 可讓您使用簡單的查詢語言，輕鬆搜尋多個記錄群組。如果您的應用程式記錄是以 JSON 格式結構， CloudWatch Logs Insights 會自動探索多個記錄群組中的記錄串流中的 JSON 欄位。您可以使用 CloudWatch Logs Insights 來分析您的應用程式和系統記錄，以儲存查詢以備將 future 使用。 CloudWatch Logs Insights 的查詢語法支援函式，例如 sum ()、avg ()、count ()、min () 和 max () 的彙總等函式，可協助您疑難排解應用程式或效能分析。

如果您使用內嵌指標格式建立 CloudWatch 指標，您可以使用支援的彙總函數，查詢內嵌的指標格式記錄檔，以產生一次性量度。這樣可以根據需要捕獲生成特定指標所需的數據點，而不是主動將其捕獲為自定義指標，從而有助於降低 CloudWatch 監視成本。這對於高基數會產生大量量度的維度特別有效。 CloudWatch 容器深入解析也會採用此方法並擷取詳細的效能資料，但只會產生此資料子集的 CloudWatch 指標。

例如，下列內嵌測量結果項目只會從內嵌 CloudWatch量度格式陳述式中擷取的測量結果資料產生一組有限的測量結果：

```
{
 "AutoScalingGroupName": "eks-e0bab7f4-fa6c-64ba-dbd9-094aee6cf9ba",
 "CloudWatchMetrics": [
 {
 "Metrics": [
 {
 "Unit": "Count",
 "Name": "pod_number_of_container_restarts"
 }
 ],
 "Dimensions": [
 [
 "PodName",
 "Namespace",
 "ClusterName"
 ]
 ],
 "Namespace": "ContainerInsights"
 }
 ],
```

```
"ClusterName": "eksdemo",
"InstanceId": "i-03b21a16b854aa4ca",
"InstanceType": "t3.medium",
"Namespace": "amazon-cloudwatch",
"NodeName": "ip-172-31-10-211.ec2.internal",
"PodName": "cloudwatch-agent",
"Sources": [
"cadvisor",
"pod",
"calculated"
],
"Timestamp": "1605111338968",
"Type": "Pod",
"Version": "0",
"pod_cpu_limit": 200,
"pod_cpu_request": 200,
"pod_cpu_reserved_capacity": 10,
"pod_cpu_usage_system": 3.268605094109382,
"pod_cpu_usage_total": 8.899539221131045,
"pod_cpu_usage_user": 4.160042847048305,
"pod_cpu_utilization": 0.44497696105655227,
"pod_cpu_utilization_over_pod_limit": 4.4497696105655224,
"pod_memory_cache": 4096,
"pod_memory_failcnt": 0,
"pod_memory_hierarchical_pgfault": 0,
"pod_memory_hierarchical_pgmajfault": 0,
"pod_memory_limit": 209715200,
"pod_memory_mapped_file": 0,
"pod_memory_max_usage": 43024384,
"pod_memory_pgfault": 0,
"pod_memory_pgmajfault": 0,
"pod_memory_request": 209715200,
"pod_memory_reserved_capacity": 5.148439982463127,
"pod_memory_rss": 38481920,
"pod_memory_swap": 0,
"pod_memory_usage": 42803200,
"pod_memory_utilization": 0.6172094650851303,
"pod_memory_utilization_over_pod_limit": 11.98828125,
"pod_memory_working_set": 25141248,
"pod_network_rx_bytes": 3566.4174629544723,
"pod_network_rx_dropped": 0,
"pod_network_rx_errors": 0,
"pod_network_rx_packets": 3.3495665260575094,
"pod_network_total_bytes": 4283.442421354973,
```

```
 "pod_network_tx_bytes": 717.0249584005006,
 "pod_network_tx_dropped": 0,
 "pod_network_tx_errors": 0,
 "pod_network_tx_packets": 2.6964010534762948,
 "pod_number_of_container_restarts": 0,
 "pod_number_of_containers": 1,
 "pod_number_of_running_containers": 1,
 "pod_status": "Running"
}
```

但是，您可以查詢捕獲的指標以獲得進一步的見解。例如，您可以執行下列查詢，以查看有記憶體分頁錯誤的最新 20 個網繭：

```
fields @timestamp, @message
| filter (pod_memory_pgfault > 0)
| sort @timestamp desc
| limit 20
```

使用亞馬遜 OpenSearch 服務執行日誌分析

CloudWatch 透過訂閱篩選器，您可以透過訂閱篩選器將日 CloudWatch 誌資料從日誌群組串流到您選擇的 Amazon OpenSearch 服務叢集，與 Amazon 服務整合。 OpenSearch 您可以用 CloudWatch 於主要日誌和指標擷取和分析，然後針對下列使用案例使用 Amazon Ser OpenSearch vice 進行擴充：

- 精細的資料存取控制 — Amazon Ser OpenSearch vice 可讓您將資料存取限制在欄位層級，並根據使用者許可協助匿名化欄位中的資料。如果您想要支援疑難排解而不暴露敏感資料，此功能非常有用。
- 跨多個帳戶、區域和基礎設施彙總和搜尋日誌 — 您可以將多個帳戶和區域的日誌串流到一個通用的 Amazon Ser OpenSearch vice 叢集。您的集中式營運團隊可以分析趨勢、問題，並執行跨帳戶和區域的分析。將 CloudWatch 日誌串流至 Amazon Ser OpenSearch vice 也可協助您在中央位置搜尋和分析多區域應用程式。
- 使用 ElasticSearch代理程式直接將日誌寄送和豐富到 Amazon Ser OpenSearch vice — 您的應用程式和技術堆疊元件可以使用 CloudWatch 代理程式不支援的作業系統。您也可能想要在將記錄檔資料傳送至記錄解決方案之前，先進行豐富和轉換。Amazon OpenSearch 服務支援標準的彈性搜尋用戶端，例如彈性 Beats 系列資料寄件者和 Logstash，在將日誌資料傳送至亞馬遜 OpenSearch 服務之前支援日誌擴充和轉換。

- 現有的營運管理解決方案使用 Logstash、Kibana (ELK) 堆疊進行記錄和監控 — 您可能已經在 Amazon OpenSearch 服務或開放原始碼 Elasticsearch 上投入了大量投資，而且已經設定了許多工作負載。ElasticSearch您可能還擁有在 Kibana 中建立的操作儀表板，以供您繼續使用。

如果您不打算使用 CloudWatch 日誌，則可以使用 Amazon OpenSearch 服務支援的代理程式、日誌驅動程式和程式庫 (例如，流利位元、流暢、記錄檔和 ElasticSearch API 的開放式發行版) 將您的日誌直接傳送到 Amazon OpenSearch 服務並略過 CloudWatch。不過，您也應該實作解決方案來擷取AWS服務所產生的記錄檔。 CloudWatch 日誌是許多AWS服務的主要日誌捕獲解決方案，多個服務會自動在中創建新的日誌組 CloudWatch。例如，Lambda 會為每個 Lambda 函數建立新的記錄群組。您可以為日誌群組設定訂閱篩選器，以將其日誌串流到 Amazon OpenSearch 服務。您可以針對要串流至 Amazon OpenSearch 服務的每個個別日誌群組手動設定訂閱篩選器。或者，您可以部署解決方案，自動將新的記錄群組訂閱至 ElasticSearch 叢集。您可以將記錄串流至相同帳戶或集中式帳戶中的 ElasticSearch 叢集。將記錄串流至相同帳戶中的 ElasticSearch 叢集，可協助工作負載擁有者更好地分析和支援其工作負載。

您應該考慮在集中式或共用帳戶中設定 ElasticSearch 叢集，以便跨帳戶、區域和應用程式彙總記錄。例如，AWS Control Tower設定用於集中記錄的記錄封存帳戶。在中建立新帳戶時AWS Control Tower，其AWS CloudTrail和AWS Config日誌會傳送到此集中式帳戶中的 S3 儲存貯體。所檢測的記錄用AWS Control Tower於配置、變更和稽核記錄。

若要使用 Amazon Ser OpenSearch vice 建立集中式應用程式日誌分析解決方案，您可以將一或多個集中式 Amazon Ser OpenSearch vice 叢集部署到集中式記錄帳戶，並在其他帳戶中設定日誌群組，以將日誌串流到集中式的 Amazon OpenSearch 服務叢集。

您可以建立個別的 Amazon Ser OpenSearch vice 叢集來處理可能分散在各個帳戶之間的不同應用程式或雲端架構層。使用個別的 Amazon Ser OpenSearch vice 叢集可協助您降低安全性和可用性風險，而且擁有通用的 Amazon Ser OpenSearch vice 叢集可讓您更輕鬆地在同一叢集中搜尋和關聯資料。

CloudWatch 警示選項

對重要指標執行一次性自動分析可幫助您在問題影響您的工作負載之前檢測和解決問題。 CloudWatch 可以通過在特定時間段內使用多個統計信息來輕鬆繪製和比較多個指標。您可以使用 CloudWatch 以搜索具有所需維度值的所有量度，以查找分析所需的指標。

我們建議您通過包含一組初始衡量指標和維度來開始衡量指標捕獲方法，以便用作監視工作負載的基線。隨着時間的推移，工作負載逐漸成熟，您可以添加其他指標和維度來幫助您進一步分析和支持它。應用程序或工作負載可能會使用多個AWS資源並擁有自己的自定義指標，則應將這些資源分組在命名空間下，以便更易於識別這些資源。

您還應該考慮日誌記錄和監控數據的關聯方式，以便您可以快速識別相關的日誌記錄和監控數據，以診斷特定問題。您可以使用CloudWatch ServiceLens以關聯跟蹤、指標、日誌和警報以診斷問題。您還應考慮在工作負載日誌中的指標和標識符中包含其他維度，以幫助您快速搜索和識別跨系統和服務的問題。

使用 CloudWatch 警示來監控和警示

您可以使用CloudWatch 警示以減少工作負載或應用程序中的手動監控。首先，您應該查看為每個工作負載組件捕獲的度量，並確定每個度量的適當閾值。確保您確定在違反閾值時必須通知哪些團隊成員。您應該建立和定位通訊組，而不是單個團隊成員。

CloudWatch 警報可以與您的服務管理解決方案集成，以自動創建新票證並運行操作工作流程。例如：AWS提供AWSService Management ConnectorServiceNow和Jira ServiceLens，以協助您快速設定集成。此方法對於確保發出的警報得到確認並與可能已在這些產品中定義的現有操作工作流程保持一致至關重要。

您還可以為具有不同閾值和評估週期的同一指標創建多個警報，這有助於建立升級過程。例如，如果您有`OrderQueueDepth`衡量指標，您可以在短時間內定義一個較低的閾值，通過電子郵件或Slack。您還可以在相同閾值的 15 分鐘內為同一指標定義另一個警報，並通知應用團隊和應用程序團隊的負責人。最後，您可以為 30 分鐘內的硬平均閾值定義第三個警報，通知上層管理層並通知所有團隊成員之前通知。創建多個警報可幫助您針對不同條件採取不同的操作。您可以從一個簡單的通知流程開始，然後根據需要進行調整和改進。

使用 CloudWatch 用於監控和報警的異常檢測

您可以使用CloudWatch 異常偵測如果您不確定要應用於特定指標的閾值，或者您希望警報根據觀察到的歷史值自動調整閾值。 CloudWatch 異常檢測對於可能具有定期、可預測的活動變化的指標尤其有

用，例如，當天交貨的每日採購訂單在截止時間之前增加。異常檢測可實現自動調整的閾值，並有助於減少誤報。您可以為每個指標和統計信息啟用異常檢測，並配置 CloudWatch 以基於異常值進行警報。

例如，您可以啟用異常檢測`CPUUtilization`指標和`AVG`EC2 執行個體的統計資料。然後，異常檢測使用長達 14 天的歷史數據來創建機器學習 (ML) 模型。您可以創建具有不同異常檢測頻段的多個警報，以建立警報升級過程，類似於創建具有不同閾值的多個標準警報。

如需此章節的詳細資訊，請參建立以異常偵測為基礎的 CloudWatch 警示中的 CloudWatch 文件中)。

跨多個區域和帳户執行個體警示

應用程序和工作負載所有者應為跨多個區域的工作負載創建應用程序級別的警報。我們建議在部署工作負載的每個帳户和區域中創建單獨的警報。您可以通過使用帳户和地區不可知的方式簡化和自動化此過程AWS CloudFormation StackSets 和模板來部署具有所需警報的應用程序資源。模板您可以將警報操作配置為針對常見的 Amazon SSimple Notification Service (Amazon SNS) 主題，這意味着使用相同的通知或修正操作，而不管賬户或區域如何。

在多賬户和多區域環境中，我們建議您為您的帳户和地區創建聚合警報，以便通過使用AWS CloudFormation StackSets 和聚合衡量指標，如平均`CPUUtilization`跨所有 EC2 執行個體。

您還應該考慮為每個配置為標準 CloudWatch 指標和日誌。例如，您可以為每個 EC2 實例創建一個單獨的警報，以監控 CPU 利用率指標，並在每天平均 CPU 使用率超過 80% 時通知中央運營團隊。您還可以創建標準警報，每天監控平均 CPU 使用率低於 10%。這些警報可幫助中央運營團隊與特定的工作負載所有者合作，以便在需要時更改 EC2 實例的大小。

使用 EC2 實例標籤自動創建警報

為 EC2 實例創建一組標準警報可能非常耗時、不一致且容易出錯。您可以通過使用亞馬遜雲監視自動報警解決方案為 EC2 實例自動創建一組標準的 CloudWatch 警報，並根據 EC2 實例標籤創建自定義警報。該解決方案無需手動創建標準警報，並且在使用 CloudEndure 等工具的 EC2 實例的大規模遷移期間非常有用。您也可以將此解決方案與AWS CloudFormation StackSets 以支持多個區域和帳號。如需詳細資訊，請參閱「」使用標籤創建和維護亞馬遜 CloudWatch Amazon EC2 執行個體的警示在AWS 部落格。

監控應用程序和服務可用性

CloudWatch 可幫助您監控和分析應用程序和工作負載的性能和運行時方面。您還應監控應用程序和工作負載的可用性和可訪問性方面。您可以使用主動監控方法Amazon Route 53 運作狀態檢查和CloudWatch Synthetics。

如果要監視通過 HTTP 或 HTTPS 到網頁的連接，或通過 TCP 到公有域名系統 (DNS) 名稱或 IP 地址的網絡連接，可以使用 Route 53 運行狀況檢查。Route 53 運行狀況檢查從您以 10 秒或 30 秒間隔指定的區域啟動連接。您可以選擇運行狀況檢查的多個區域，每個運行狀況檢查獨立運行，並且必須至少選擇三個區域。如果 HTTP 或 HTTPS 請求的響應正文出現在運行狀況檢查評估返回的前 5,120 字節的數據中，則可以搜索特定子字符串。如果返回 2xx 或 3xx 個回應，HTTP 或 HTTPS 請求會被視為運作狀態良好。Route 53 運行狀況檢查可用於通過檢查其他運行狀況檢查的運行狀況來創建複合運行狀況檢查。如果您有多個服務終端節點，並且希望在其中一個終端節點運行狀況不佳時執行相同的通知，則可以執行此操作。如果您對 DNS 使用路由 53，則可以將路由 53 配置為故障轉移到另一個 DNS 條目如果運行狀況檢查變得不正常。對於每個關鍵工作負載，應考慮為對正常操作至關重要的外部終端節點設置 Route 53 運行狀況檢查。Route 53 運行狀況檢查可幫助您避免將故障轉移邏輯寫入應用程序。

CloudWatch 合成器允許您將金絲雀定義為腳本，以評估工作負載的運行狀況和可用性。Canaries 是以 Node.js 或 Python 撰寫的指令碼，可以使用 HTTP 或 HTTPS 協議進行操作。Canary 會使用 Node.js 或 Python 作為架構，在您的帳戶中建立 Lambda 函數。您定義的每個金絲雀都可以對不同的終端執行多個 HTTP 或 HTTPS 調用。這意味着您可以監視一系列步驟的運行狀況，例如使用案例或具有下遊依賴關係的終端節點。Canaries 創建 CloudWatch 指標，這些指標包括已運行的每個步驟，以便您可以獨立提醒和測量不同步驟。雖然金絲雀比 Route 53 運行狀況檢查需要更多的規劃和努力來開發，但它們為您提供了高度可定製的監控和評估方法。Canary 還支持在虛擬私有雲 (VPC) 中運行的私有資源，這使得它們非常適合在您沒有端點的公有 IP 地址時進行可用性監控。您也可以使用金絲雀監控本地工作負載，只要您在 VPC 內部連接到終端節點。當您的工作負載包括本地存在的終端節點時，這一點尤其重要。

使用跟蹤應用程序AWS X-Ray

通過應用程序發出的請求可能包括對本地服務器、Amazon EC2、容器或 Lambda 中運行的數據庫、應用程序和 Web 服務的調用。通過實施應用程序跟蹤，您可以快速識別使用分佈式組件和服務的應用程序中出現問題的根本原因。您可以使用AWS X-Ray以跨多個組件跟蹤應用程序請求。X-Ray 樣本和可視化請求服務圖表當它們通過應用程序組件並且每個組件都表示為一個段時。X-Ray 會生成跟蹤標識符，以便您可以在請求流經多個組件時關聯該請求，從而幫助您從端到端查看請求。您可以通過包含註釋和元數據來幫助唯一搜索和識別請求的特徵，從而進一步增強這一點。

我們建議您使用 X-Ray 配置和測量應用程序中的每個服務器或端點。X-Ray 是通過調用 X-Ray 服務在您的應用程序代碼中實現的。X-Ray 還提供AWS適用於多種語言的 SDK，包括自動將數據發送到 X-Ray 的分析客户端。X-Ray 軟件開發工具包為用於調用其他服務（例如 HTTP、MySQL、PostgreSQL 或 MongoDB）的常見庫提供修補程序。

X-Ray 提供了一個 X-Ray 守護程序，您可以在 Amazon EC2 和亞馬遜雲服務器上安裝和運行，以便將數據中繼到 X-Ray。X-Ray 為您的應用程序創建跟蹤，從運行服務請求的 X-Ray 守護程序的服務器和容器中捕獲性能數據。X-Ray 自動對您的呼叫進行調用AWS服務（例如 Amazon DynamoDB），通過修補AWS開發套件。X-Ray 還可以自動與 Lambda 功能集成。

如果您的應用程序組件調用無法配置和安裝 X-Ray 守護程序或儀器代碼的外部服務，則可以創建子段來包裝對外部服務的調用。X-Ray 相關 CloudWatch 日誌和指標與應用程序跟蹤（如果您使用適用於 JAVA 的 AWS X-Ray SDK，這意味着您可以快速分析請求的相關指標和日誌。

部署 X-Ray 常駐程式，追蹤 Amazon EC2 上的應用程式和服務

您需要在您的應用程式組件或微服務上運行 X-Ray 常駐程式。您可以使用使用者資料指令碼在預配 EC2 實例時部署 X-Ray 守護程序，或者如果您創建自己的 AMI，則可以將其包含在 AMI 構建過程中。當 EC2 實例是短暫的時候，這可能特別有用。

您應該使用狀態管理器來確保 X-Ray 守護程序一致地安裝在 EC2 實例上。適用於 Amazon EC2視窗實例，您可以使用 Systems ManagerAWS-正在執行功能外殼指令碼執行視窗腳本，下載並安裝 X-Ray 代理。對於 Linux 上的 EC2 實例，您可以使用AWS-運行外殼腳本文檔來運行下載並將代理作為服務。

您可以使用 Systems ManagerAWS-執行遠程腳本以在多帳户環境中運行腳本。您必須創建可從您的所有賬户訪問的 S3 存儲桶，我們建議使用基於組織的存儲桶策略創建 S3 存儲桶如果您使用AWS Organizations。然後，您將腳本上傳到 S3 存儲桶，但確保 EC2 實例的 IAM 角色具有訪問存儲桶和腳本的權限。

您還可以將狀態管理器配置為將腳本與安裝了 X-Ray 代理的 EC2 實例相關聯。由於您的所有 EC2 實例可能不需要或使用 X-Ray，因此您可以定位與實例標籤的關聯。例如，您可以根據`InstallAWSXRayDaemonWindows`或者`InstallAWSXRayDaemonLinux`標籤。

部署 X-Ray 常駐程式，追蹤 Amazon ECS 或 Amazon EKS 上的應用程式和服務

您可以部署X-Ray 常駐程式作為基於容器的工作負載（如亞馬遜雲服務器或 Amazon EKS）的旁邊容器。然後，如果您使用亞馬遜彈性雲服務器，則應用程序容器可以通過容器鏈接連接到您的側車容器，或者如果您使用AWSvpc 網路模式。

對於 Amazon EKS，您可以在應用程序的容器定義中定義 X-Ray 守護程序，然後您的應用程序可以通過您指定的容器端口上的 localhost 連接到守護進程。

將 Lambda 配置為將請求跟蹤到 X-Ray

您的應用程序可能包括對 Lambda 函數的調用。您無需安裝 Lambda X-Ray 常駐程式，因為常駐程式程式由 Lambda 完全管理，且無法由使用者設定。您可以通過使用AWS Management Console並檢查主動追蹤選項。

如需進一步檢測，您可以將 Lambda 函數與 X-Ray 開發套件組合，以記錄傳出呼叫並添加註釋或元數據。

設定適用於 X-Ray 的應用

您應該評估與應用程序編程語言一致的 X-Ray SDK，並對應用程序對其他系統發出的所有調用進行分類。查看您選擇的庫提供的客户端，並查看 SDK 是否可以自動測試應用程序請求或響應的跟蹤。確定 SDK 提供的客户端是否可用於其他下遊系統。對於應用程序調用且無法使用 X-Ray 進行測試的外部系統，應創建一個自定義子段，以在跟蹤信息中捕獲和標識它們。

在測試應用程序時，請確保創建註釋以幫助您識別和搜索請求。例如，您的應用程序可能會為客户使用標識符，例如`customer id`，或根據不同用户在應用程序中的角色對不同用户進行細分。

您最多可以為每條跟蹤創建 50 個註釋，但只要段文檔不超過 64 千字節，就可以創建包含一個或多個字段的元數據對象。您應有選擇地使用註釋來查找信息，並使用元數據對象提供更多上下文，以幫助在找到請求後對其進行故障排除。

設定 X-Ray 取樣規則

By自訂抽樣規則，您可以控制記錄的數據量並修改取樣行為，而無需修改或重新部署代碼。抽樣規則會告知 X-Ray 開發套件針對一組條件要記錄多少請求。默認情況下，X-Ray 開發套件會記錄第一個請求每秒和 5% 的任何額外請求。每秒一個請求是儲槽。這可確保只要服務持續提供請求，每秒都會記錄至少一個追蹤。5% 是超過儲層大小的額外請求抽樣速率。

您應查看並更新默認配置，以確定適合您的帳户的值。在開發、測試、性能測試和生產環境中，您的要求可能會有所不同。您的應用程序可能需要根據其接收的流量或其嚴重程度級別自己的採樣規則。您應該從基準開始，並定期重新評估基準是否符合您的要求。

使用雲手錶的儀錶板和可視化

儀錶板可幫助您快速關注應用程序和工作負載的關注領域。CloudWatch 提供自動儀錶板，您還可以輕鬆創建使用 CloudWatch 指標。 CloudWatch 儀錶板提供的見解比單獨查看指標更多，因為它們可幫助您關聯多個指標並識別趨勢。例如，包含已收到的訂單、內存、CPU 利用率和數據庫連接的儀錶板可幫助您將工作負載度量的更改與多個AWS資源，而您的訂單計數正在增加或減少。

您應該在客戶和應用程序級別創建儀錶板，以監控工作負載和應用程序。您可以透過使用 CloudWatch 自動儀錶板，它們是AWS使用特定於服務的指標預配置的服務級別儀錶板。自動服務儀錶板顯示所有標準 CloudWatch 指標。自動儀錶板可以繪製用於每個服務指標的所有資源的圖形，並幫助您快速識別您的賬戶中的異常資源。這可以幫助您識別利用率高和低利用率的資源，從而幫助您優化成本。

建立跨服務儀表板

您可以通過查看AWS服務並使用添加到儀表板選項從動作選單。然後，您可以將其他自動儀錶板中的衡量指標添加到新的儀錶板，並刪除指標以縮小儀錶板的焦點範圍。您還應該添加自己的自定義指標來跟蹤關鍵觀測值（例如，收到的訂單或每秒交易量）。創建自己的自定義跨服務儀錶板可幫助您專注於與工作負載最相關的指標。我們建議您創建涵蓋關鍵指標並顯示賬戶中所有工作負載的賬戶級別跨服務儀錶板。

如果您的雲運營團隊擁有中央辦公空間或公共區域，則可以顯示 CloudWatch 儀錶板上的大型電視顯示器上的全屏模式，具有自動刷新功能。

創建特定於應用程序或工作負載的儀錶板

我們建議您創建特定於應用程序和工作負載的儀錶板，重點關注生產環境中每個關鍵應用程序或工作負載的關鍵指標和資源。特定於應用程序和工作負載的儀錶板側重於自定義應用程序或工作負載指標以及重要的AWS影響其性能的資源衡量指標。

您應定期評估和定製您的 CloudWatch 應用程序或工作負載儀錶板，以便在事件發生後跟蹤關鍵指標。引入或停用功能時，還應更新特定於應用程序或工作負載的儀錶板。對工作負載和特定於應用程序的儀錶板的更新應該是不斷提高質量的必要活動，以及記錄和監控。

建立跨帳户或跨區域儀表板

AWS資源主要是「區域」，並且衡量指標、警報和儀錶板特定於部署資源的區域。這可能要求您更改區域以查看跨區域工作負載和應用程序的指標、儀錶板和警報。如果您將應用程序和工作負載分為多個

帳戶，您可能還需要重新進行身份驗證並登錄到每個帳戶。但是, CloudWatch 支持從單個賬戶查看跨賬戶和跨區域數據，這意味着您可以在單個帳戶和區域中查看指標、警報、儀錶板和日誌小組件。如果您有一個集中的日誌記錄和監視帳戶，這將非常有用。

客户所有者和應用程序團隊所有者應為特定於賬户的跨區域應用程序創建儀錶板，以便在集中位置有效監控關鍵指標。CloudWatch 儀錶板自動支持跨區域小組件，這意味着您可以創建包含來自多個區域的指標的儀錶板，而無需進行進一步配置。

一個重要的例外是 CloudWatch 日誌見解小組件，因為只能為您當前登錄的帳户和區域顯示日誌數據。您可以使用指標篩選器從日誌中創建特定於區域的指標，這些指標可以顯示在跨區域儀錶板上。然後，您可以在需要進一步分析這些日誌時切換到特定區域。

運營團隊應創建集中式儀錶板，以監控重要的跨賬户和跨區域指標。例如，您可以創建跨賬户儀錶板，其中包括每個帳户和區域中的總 CPU 利用率。您也可以使用指標數學，以彙整多個帳户和區域的資料。

使用度量數學微調可觀察性和警報

您可以使用量度數學來幫助計算與您的工作負載相關的格式和表達式的指標。計算的指標可以保存並在儀錶板上查看，以便進行跟蹤。例如，標準的 Amazon EBS 卷指標提供了讀取次數（`VolumeReadOps`）和寫入（`VolumeWriteOps`) 在特定時段內執行的操作。

但是,AWS提供了關於亞馬遜 EBS 卷在 IOPS 中性能的指南。您可以在度量數學中繪製和計算 Amazon EBS 體積的 IOPS，方法是添加`VolumeReadOps`和`VolumeWriteOps`，然後除以為這些指標選擇的時間段。

在此示例中，我們將期間內的 IOPS 進行總和，然後除以週期長度以獲取 IOPS。然後，您可以根據此指標數學表達式設置警報，以便在卷的 IOPS 接近其卷類型的最大容量時提醒您。有關使用指標數學監控 Amazon Elastic File System (Amazon EFS) 檔案系統的更多信息和示例 CloudWatch 指標，請參閱亞馬遜 CloudWatch 指標數學簡化了對 Amazon EFS 文件系統的近乎實時的監控等在AWS部落格。

使用亞馬遜雲服務器、亞馬遜 EKS 和 Lambda 的自動儀錶板 CloudWatchContainer 深入分析和 CloudWatch Lambda Insights

CloudWatch 容器洞察為在亞馬遜雲服務器和 Amazon EKS 上運行的容器工作負載創建動態、自動的儀錶板。您應該啟用容器見解，使 CPU、內存、磁盤、網絡和診斷信息（如容器重新啟動失敗）具有可觀察性。容器見解生成動態儀錶板，您可以在集羣、容器實例或節點、服務、任務、容器級別和單個容器級別快速篩選這些儀錶板。容器洞見在羣集和節點或容器實例級別配置取決於AWS服務。

與容器見解類似， CloudWatch Lambda 見解為您的 Lambda 函數創建動態的自動儀錶板。此解決方案會收集、彙總和摘要系統層級的指標，包括 CPU 時間、記憶體、磁碟和網路。它也會收集、彙總和摘要診斷資訊，例如冷啟動和 Lambda 工作人員關閉，協助您隔離和快速解決 Lambda 函數問題。Lambda 在功能級別啟用，不需要任何代理。

容器見解和 Lambda 見解還可幫助您快速切換到應用程序或性能日誌、X-Ray 跟蹤和服務地圖，以便可視化容器工作負載。他們都使用 CloudWatch 要捕獲的嵌入指標格式 CloudWatch 指標與性能日誌。

您可以創建共享 CloudWatch 儀錶板，該儀錶板使用容器見解和 Lambda 見解捕獲的指標。您可以透過篩選和查看 CloudWatch 容器見解，然後選擇添加到儀錶板選項，該選項允許您將顯示的指標添加到標準 CloudWatch 儀錶板。然後，您可以刪除或自定義衡量指標，並添加其他指標以正確表示您的工作負載。

與 CloudWatch 整合AWS服務

AWS提供了許多服務，包括用於日誌記錄和指標的其他配置選項。這些服務通常使您能夠配置 CloudWatch 日誌輸出的日誌和 CloudWatch 指標輸出的指標。用於提供這些服務的底層基礎架構由AWS和無法訪問，但您可以使用預配置服務的日誌記錄和指標選項來獲取更多見解並排除問題。例如，您可以VPC 流程日誌至 CloudWatch，或者您也可以將 Amazon Relational Database Service (Amazon RDS) 實例配置為將日誌發佈至 CloudWatch。

最AWS使用記錄 API 呼叫整合至AWS CloudTrail。 CloudTrail 也現在支援與 整合 CloudWatch 日誌，這意味着您可以搜索和分析AWS服務。您也可以使用 Amazon Patabase CloudWatch Events 或 Amazon Events EventBridge 創建和配置自動化和通知 CloudWatch 在中執行的特定操作的事件事件規則AWS服務。某些服務直接整合取代為 CloudWatch 活動和 EventBridge. 您也可以創建通過 CloudTrail 交付的事件。

用於儀表板和可視化的亞馬遜託管 Grafana

Amazon Managed Grafana可用於觀察和可視化您的AWS工作負載。Amazon 受管 Grafana 可協助您大規模視覺化和分析您的營運資料。Grafana是一個開放原始碼分析平台，可協助您查詢、視覺化、提醒和瞭解指標，不論其儲存在何處。如果您的組織已經使用 Grafana 來視覺化現有工作負載，而且您想要將涵蓋範圍擴展至AWS工作負載。您可以使用亞馬遜託管 Grafana 與 CloudWatch 由將其新增為資料來源，這表示您可以使用建立視覺效果 CloudWatch指標。Amazon Managed GrafanaAWS Organizations並且您可以使用以下方式集中控制面板 CloudWatch 來自多個帳戶和區域的指標。

下表提供使用亞馬遜受管的 Grafana 而非使用的優點和考量 CloudWatch 用於儀表板。根據最終使用者、工作負載和應用程式的不同需求，混合式方法可能適合。

建立視覺效果和儀表板，並與 Amazon 受管理的 Grafana 和開放原始碼 Grafana 支援的資料來源整合	Amazon 受管的 Grafana 可協助您從許多不同的資料來源建立視覺效果和儀表板，包括 CloudWatch 指標。亞馬遜受管的 Grafana 包含許多跨越的內建資料來源AWS服務、開放原始碼軟體和 COTS 軟體。如需此項目的詳細資訊，請參閱內建資料來源在亞馬遜託管 Grafana 文檔中。您也可以透過將工作區升級至新增對更多資料來源的支援Grafana 企業。Grafana 也支持資料來源外掛程式允許您與不同的外部系統進行通信。 CloudWatch儀表板需要 CloudWatch 指標或 CloudWatch 日誌見解查詢要顯示在 CloudWatch 儀表板。
分別管理儀表板解決方案的存取AWS帳戶存取	亞馬遜託管 Grafana 要求使用AWS IAM Identity Center(IAM 身分識別中心) 和AWS Organizations進行身份驗證和授權。這可讓您使用可能已與 IAM 身分中心搭配使用的聯合身分識別，向 Grafana 驗證使用者。AWS Organizations。但是，如果您不使用 IAM 身分中心或 AWS Organizations，然後將其設置為亞馬遜託管 Grafana 設置過程的一部分。如果您的組織限制了 IAM 身分中心的使用，或者，這可能會成為一個問題AWS Organizations。

透過以下方式擷取和存取多個帳戶和區域的資料AWS Organizations整合	Amazon Managed GrafanaAWS Organizations使您能夠從中讀取數據AWS來源，如 CloudWatch 和 Amazon OpenSearch 為您的所有帳戶提供服務。這樣就可以建立儀表板，使用跨帳戶的資料顯示視覺效果。若要自動啟用資料存取AWS Organizations，您需要設定您的 Amazon Managed GrafanaAWS Organizations 管理帳戶。這是不建議基於AWS Organizations 管理帳戶的最佳實務。相比之下， CloudWatch 也支援跨帳戶、跨帳戶、跨帳戶、跨帳戶、跨 CloudWatch 指標。
使用開放原始碼社群中提供的進階視覺化 Widget 和 Grafana 定義	Grafana 提供大量的視覺效果集合，您可以在建立儀表板時使用這些視覺效果。此外，還有一個大型的社群貢獻儀表板庫，您可以根據自己的需求編輯和重複使用這些儀表板。
使用儀表板搭配全新和現有的 Grafana 部署	如果您已經使用 Grafana，您可以從 Grafana 部署匯入和匯出儀表板，並自訂這些儀表板，以便在亞馬遜受管 Grafana 中使用。亞馬遜託管 Grafana 允許您在 Grafana 上進行標準化，作為您的儀表板解決方案。
工作區、權限和資料來源的進階設定和設定	Amazon 受管的 Grafana 可讓您建立多個 Grafana 工作區，這些工作區擁有自己的一組已設定資料來源、使用者和政策。這可協助您滿足更進階的使用案例需求，以及進階的安全性設定。如果您的團隊還沒有這些技能，進階功能可能需要他們提升在 Grafana 的經驗。

設計和實施日誌記錄和監控 CloudWatch 常見問答集

本節提供了關於使用 CloudWatch 設計和實施日誌記錄和監控解決方案的常見問題的答案。

在何處存儲 CloudWatch 組態檔案？

所以此 CloudWatch 代理可以應用多個配置文件，這些文件存儲在 CloudWatch 組態檔案。理想情況下，您應將 CloudWatch 配置存儲為一組文件，因為您可以在多個帳户和環境中進行版本控制並再次使用它們。如需此項目的詳細資訊，請參管理 CloudWatch組態章節。或者，您也可以將配置文件存儲在 GitHub ，並在配置新 EC2 實例時自動檢索配置文件。

當警報發生時，如何在服務管理解決方案中創建票證？

您將服務管理系統與 Amazon Simple Notification Service (Amazon SNS) 主題集成，並將 CloudWatch 警報以在發出警報時通知 SNS 主題。您的集成系統接收 SNS 消息，並可以使用服務管理系統 API 或 SDK 創建票證。

如何使用 CloudWatch 捕獲我的容器中的日誌文件？

亞馬遜雲服務器任務和亞馬遜 EKS 容器可以配置為自動發送標準輸出和 STDERR 輸出到雲手錶。記錄容器化應用程序的建議方法是讓容器將其輸出發送到 STDOUT 和 STDERR。這也包含在十二因素應用宣言。

但是，如果要將特定的日誌文件發送到 CloudWatch 然後，您可以在 Amazon EKS 容器或亞馬遜彈性雲服務器任務定義中安裝卷到您的應用程序寫入批次文件的位置，並使用 Fluentd 或 Fluent Bit 的旁邊容器將日誌發送到 CloudWatch。您應該考慮將容器中的特定日誌文件符號鏈接到`/dev/stdout`和`/dev/stderr`。如需此項目的詳細資訊，請參查看容器或服務的日誌。

如何監控健康問題AWS服務？

您可以使用AWS Health Dashboard要監控AWS運作狀態事件。您也可以參AWS 健康工具 GitHub 存儲庫，瞭解與AWS運作狀態事件。

如何建立自定義 CloudWatch 指標是否存在代理支持？

您可以使用內嵌指標格式將指標引入 CloudWatch。您也可以使用AWSSDK（例如，put_metric_data),AWS CLI(例如，put-metric-data），或AWSAPI（例如，PutMetricData) 來建立

自定義指標。您應該考慮如何長期維護任何自定義邏輯。一種方法是將 Lambda 與集成的嵌入式指標格式支持結合使用來創建您的指標，以及 CloudWatch 活動事件計劃規則以建立指標的週期。

如何將現有的日誌記錄和監控工具與AWS?

您應參考軟件或服務供應商提供的指導，以便與AWS。您可以使用代理軟件、SDK 或提供的 API 將日誌和指標發送到其解決方案。您也可以使用開源解決方案，如 Fluentd 或 Fluent Bit，根據供應商的規格進行配置。您也可以使用AWS軟體開發套件和 CloudWatch 使用 Lambda 和 Kinesis Data Streams 記錄訂閱篩選器，以創建自定義日誌處理器和發貨人。最後，如果您使用多個帳户和地區，您還應該考慮如何集成軟件。

資源

簡介

- AWSWell-Architected

目標的業務成果

- logging-monitoring-apg-guide-例子
- 雲計算的六大優勢

規劃您的 CloudWatch 部署

- AWS Organizations 術語與概念
- AWS Systems Manager快速設定
- 使用 CloudWatch 代理程式從 Amazon EC2 執行個體和內部部署伺服器收集指標和日誌
- cloudwatch-config-s3-桶. 羊
- 使用精靈建立 CloudWatch 代理程式組態檔案
- 企業 DevOps：為什麼您應該運行您構建的內容
- 將日誌資料匯出至 Amazon S3
- Amazon Ser OpenSearch vice 中的精細存取控制
- Lambda 配額
- 手動建立或編輯 CloudWatch 代理程式組態檔案
- 使用訂閱即時處理日誌資料
- 建立在基礎上的工具AWS

設定 EC2 執行個體和內部部署伺服器的 CloudWatch 代理

- Amazon EC2 指標維度
- 爆量效能執行個體

- CloudWatch 代理程式預先定義的
- 使用 procstat 外掛程式收集程序指標
- 設定監測器的 CloudWatch 代理程式
- 啟用或關閉執行個體的詳細監控
- 使用 CloudWatch 內嵌指標格式擷取高基數日誌和產生指標
- 使用日誌群組和日誌串流
- 列出您的執行個體可用的 CloudWatch 指標
- PutLogEvents
- 使用 collectd 擷取自訂指標
- 使用 StatsD 擷取自訂指標

CloudWatch Amazon EC2 和現場部署伺服器的代理程式安裝方法

- 建立混合環境的 IAM 服務角色
- 為混合環境建立受管執行個體啟用
- 建立 IAM 角色和使用者以使用 CloudWatch 代理程式
- 使用命令列下載及設定 CloudWatch 代理程式
- 如何將使用 Systems Manager 代理程式和整合 CloudWatch代理程式的內部部署伺服器設定為僅使用臨時認證？
- 堆疊集操作的先決條件
- 使用競價型例項

在 Amazon ECS 上記錄和監控

- amazon-cloudwatch-logs-for-流體位
- 亞馬遜 ECS CloudWatch 指標
- Amazon ECS 容器見解指標
- Amazon ECS 容器代理程式
- Amazon ECS 啟動類型
- 部署 CloudWatch 代理程式以收集 Amazon ECS 上的 EC2 執行個體層級指標

- 集群與 _ 雲觀察 _ Linux.YAML
- 例子-中文 (繁體)
- 電子火焰 _ 範例
- ecs-task-nginx-firelense.json
- 擷取亞馬遜 ECS 優化 AMI 中繼資料
- 使用 awslogs 日誌驅動程式
- 使用戶端程式庫產生內嵌指標格式日誌

在 Amazon EKS 上記錄和監控

- Amazon EKS 控制平面記錄
- 亞馬遜管理節點組啟動配置
- Amazon EKS 節點
- amazon-eks-nodegroup. 山羊
- 亞馬遜 EKS 服務級別協議
- Container Insights Prometheus 指標監控
- 使用 Prometheus 的控制平面度量
- 部署 Kubernetes 儀表板 (Web UI)
- Fargate 記錄
- Fargate 上亞馬遜 EKS 的流利位
- 如何在 Fargate 上使用 Amazon EKS 時擷取應用程式日誌
- 安裝 CloudWatch 代理程式以收集 Prometheus 指標
- 安裝 Kubernetes 測量結果伺服器
- 管理儀表板/儀表板
- Kubernetes 水平網繭自動配置器
- 庫氏控制平面元件
- 庫伯尼特豆莢
- 啟動範本支援
- 受管節點群組
- 受管節點更新行為

- 指標服務器
- 使用 Prometheus 和 Grafana 監控 Fargate 上的亞馬遜 EKS
- 普羅米修斯 _jmx
- 普羅米修斯/Jmx_ 出口商
- 湊集其他 Prometheus 來源並匯入這些指標
- 自我管理的節點
- 將記錄檔傳送至 CloudWatch 記錄
- 將 FluentD 設定為 DaemonSet 將日誌傳送至 CloudWatch 日誌
- 設定適用於 Amazon EKS 和 Kubernetes 的 Java/JMX 範例工作負載
- 新增 Prometheus 湊集目標的教學課程：Prometheus API 伺服器指標
- 垂直莢型自動分配器

記錄和指標AWS Lambda

- Lambda 調用錯誤
- 日誌記錄-Python 的日誌記錄工具
- 使用戶端程式庫產生內嵌指標格式日誌
- 使用 Lambda 函數指標

搜尋和分析記錄 CloudWatch

- 節拍家族
- 彈性斜角
- 彈性堆疊
- 將 CloudWatch 日誌資料串流至亞馬遜 OpenSearch 服務

警示選項 CloudWatch

- amazon-cloudwatch-auto-alarms
- AWS用於 Jira 服務管理的服務管理連接器
- AWS服務管理連接器 ServiceNow

監控應用程式和服務可用

- 設定 DNS 容錯移轉

追蹤應用程式AWS X-Ray

- Amazon ECS 任務聯網
- 在 X-Ray 主控台中設定取樣規則
- 執行視窗 PowerShell 命令或指令碼
- 在 Amazon EC2 上運行 X-Ray 常駐程式
- 將追蹤資料傳送至 X-Ray
- 服務圖表, 在, X-Ray

儀表板和視覺化 CloudWatch

- 亞馬遜 CloudWatch 指標數學簡化了對 Amazon EFS 檔案系統的近乎即時監控
- 設定 CloudWatch 容器深入解析
- 使用指標數學

CloudWatch 與AWS服務整合

- AWS CloudTrail 支援的服務和整合
- CloudWatch 來自支援服務的事件事件範例
- 活動透過 CloudTrail
- 使用記 CloudTrail 錄監控記 CloudWatch 錄檔
- 將資料庫引擎日誌發佈至 CloudWatch Logs
- 將流程記錄發佈至 CloudWatch Logs

用於儀表板和可視化的亞馬遜託管 Grafana

- 中的管理帳戶的最佳實務AWS Organizations
- 適用於亞馬遜受管 Grafana 的內建資料來源

- 跨帳戶和跨區域儀表板 CloudWatch
- Grafana 插件

文件歷史紀錄

下表說明本指南的重大變更。如果您想收到有關 future 更新的通知，您可以訂閱 RSS 摘要。

變更	描述	日期
更新的記錄資訊	更新了關於的記錄一節AWS Lambda。	2023 年 4 月 17 日
更新的組態資訊	更新並重新命名關於建立和儲存 CloudWatch 組態的區段。	2023 年 2 月 9 日
更新的量度資訊	更新 Amazon ECS 指標區段中的自訂應用程式指標資訊。	2023 年 1 月 31 日
刪除預覽通知	亞馬遜託管 Grafana 已全面推出。	2022 年 5 月 25 日
刪除部分	CloudWatch 不再支援軟體開發套件。	2022 年 1 月 7 日
初始出版	—	2021 年 4 月 30 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。