AWS白皮书

標記 AWS 資源的最佳實務

標記 AWS 資源的最佳實務: AWS白皮书

Table of Contents

標籤 AWS 資源的最佳實務

出版日期：二零二三年三月三十日（ ）文件修訂

Amazon Web Services (AWS) 能讓您以標籤形式分配中繼AWS資料給許多資源。每個標籤都是一個簡單標籤，其中包含金鑰和選用值，用於儲存有關資源或資料保留在該資源上的資源或資料的資訊。本白皮書著重於標記使用案例、策略、技術和工具，這些使用案例、策略、技術和工具可協助您依據目的、團隊、環境或其他與業務相關的條件來分類資源。實作一致的標記策略可讓您更輕鬆地篩選和搜尋資源、監控成本和使用量，以及管理您的AWS環境。

本白皮 paper 以「使用多個帳戶組織AWS環境」白皮書中提供的做法和指引為基礎。建議您在此之前閱讀該白皮書。 AWS建議您以整體方式建立雲端基礎。如需詳細資訊，請參閱上的建立您的雲端基礎AWS。

您是否具 Well-Architected？

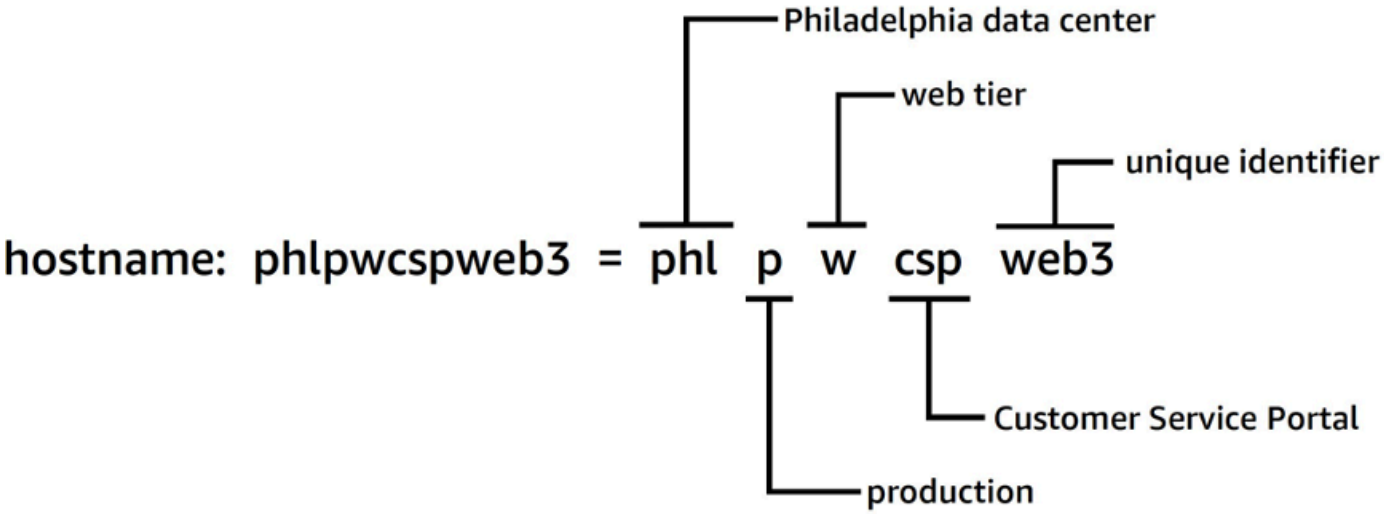
AWSWell-Architected 的架構可協助您瞭解在雲端中建置系統時所做決策的優缺點。Framework 的六大支柱可讓您學習如何設計和操作可靠、安全、高效、符合成本效益且可持續發展的系統的架構最佳實務。使用中免費提供的 AWS Well-Architected ToolAWS Management Console，您可以針對每個支柱回答一組問題，根據這些最佳實務來檢閱工作負載。

如需雲端架構的更多專家指導和最佳實務 (參考架構部署、圖表和白皮書)，請參閱架構中心。AWS

簡介

AWSAWS透過建立資源 (例如 Amazon EC2 執行個體、Amazon EBS 磁碟區、安全群組和AWS Lambda功能)，可讓您輕鬆部署工作負載。您還可以擴展託管應用程式、儲存AWS資料以及隨著時間擴展AWS基礎架構的資源集。隨著您的AWS使用量增加到跨多個應用程式的許多資源類型，您將需要一種機制來追蹤哪些資源指派給哪個應用程式。使用此機制支援您的營運活動，例如成本監控、事件管理、修補、備份和存取控制。

在內部部署環境中，這些知識通常會在知識管理系統、文件管理系統和內部 Wiki 頁面上擷取。透過組態管理資料庫 (CMDB)，您可以使用標準變更控制程序來儲存和管理相關的詳細中繼資料。這種方法提供了治理，但需要額外的努力來開發和維護。您可以採取結構化方法來命名資源，但資源名稱只能保存有限數量的資訊。



資源命名的結構化方法

例如，EC2 執行個體有一個名為 Name 的預先定義標籤，該標籤提供類似的功能，並可讓您在工作負載移至時命名工作負載AWS。

在 2010 年，AWS啟動了資源標籤，以提供可靈活且可擴展的機制，用於將中繼資料附加到您的資源。本白皮書會引導您完成整個AWS環境中開發和實作強大標記策略的程序。本指南將幫助您確保標記一致性和涵蓋範圍，以支持您的決策和運營活動

什麼是標籤？

標籤是套用至資源的索引鍵值配對，以保存有關該資源的中繼資料。每個標籤都是由鍵和可選值組成的標籤。並非所有服務和資源類型目前都支援標籤 (請參閱支援 Resource Groups 標記 API 的服務)。其他服務可能會透過自己的 API 支援標籤。標籤不會加密，不應用於儲存敏感資料 (例如個人識別資訊 (PII)。

使用者使用AWS CLI、API 或所建立並套用至AWS資源的標籤稱為使AWS Management Console用者定義的標籤。許多AWS服務 (例如 AWS CloudFormation Elastic Beanstalk 和自動 Auto Scaling) 會自動為其建立和管理的資源指派標籤。這些鍵被稱為AWS生成的標籤，並且通常以前綴aws。此前置詞不能用於使用者定義的標籤鍵中。

可新增至AWS資源的使用者定義標籤數目有使用需求和限制。如需詳細資訊，請參閱AWS一般參考指南中的標籤命名限制和需求。 AWS產生的標籤不會計入這些使用者定義的標籤限制。

表 1 — 使用者定義標籤鍵和值的範例

執行個體 ID	標籤鍵	標籤值
i-01234567abcdef89a	CostCenter	98765
	Stack	Test
我 -12345678abcdef90	CostCenter	98765
	Stack	Production

表 2 — AWS 產生的標籤範例

AWS產生的標籤鍵	理由
aws:ec2spot:fleet-request-id	識別啟動執行個體的 Amazon EC2 競價型執行個體請求
aws:cloudformation:stack-name	標識創建資源的AWS CloudFormation堆棧
lambda-console:blueprint	識別用作AWS Lambda函數範本的藍圖

AWS產生的標籤鍵	理由
elasticbeanstalk:environment-name	識別建立資源的應用程式
aws:servicecatalog: provisionedProductArn	佈建產品亞馬遜資源名稱 (ARN)
aws:servicecatalog:productArn	從中啟動佈建產品的產品 ARN

AWS產生的標籤形成命名空間。例如，在範AWS CloudFormation本中，您可以定義一組要一起部署的資源stack，其中stack-name是您指派用來識別它的描述性名稱。如果您檢查一個密鑰*aws:cloudformation:stack-name*，則可以看到用於範圍參數的命名空間使用三個元素：aws 組織，cloud form 服務以及堆棧命名參數。

使用者定義的標籤也可以使用命名空間，並建議使用組織識別碼作為前置詞。這可協助您快速識別標籤是否是受管理結構描述中的某個項目，或是您在環境中使用的服務或工具所定義的項目。

在AWS白皮書上建立您的雲端基礎中，我們建議您使用一組應該實作的標籤。不同的企業很可能會有不同的允許模式和給定標籤的不同列表。查看表 3 中的例子：

表 3-相同的標籤鍵，不同的值驗證規則

組織	標籤鍵	標籤值驗證	標籤值範例
公司 A	CostCenter	5432, 5422, 5499	5432
B 公司資訊	CostCenter	ABC*	ABC123

如果這兩個結構描述位於不同的組織中，則標籤衝突沒有問題。但是，如果這兩個環境合併，那麼命名空間可能會發生衝突，並且驗證變得更加複雜。這種情況似乎不太可能，但是企業被併購或合併，還有其他情況，例如與託管服務提供商合作的客戶，遊戲發行商或風險投資業務，其中來自不同組織的帳戶是共享AWS組織的一部分。使用商業名稱作為前綴來定義唯一命名空間，可以避免這些挑戰，如表 4 所示：

表 4 — 在標籤索引鍵中使用命名空間

組織	標籤鍵	標籤值驗證	標籤值範例
公司 A	company-a :CostCenter	5432, 5422, 5499	5432
B 公司資訊	company-b :CostCenter	ABC*	ABC123

在企業被定期收購和剝離的大型和複雜的組織中，這種情況會更頻繁地發生。隨著新收購的流程和實踐在更廣泛的團體中協調，這種情況得到解決。擁有不同的命名空間會有所幫助，因為可以報告舊標籤的使用，並與相關團隊聯繫以採用新的模式。命名空間也可以用來指示範圍，或代表與組織擁有者對齊的使用案例或責任領域。

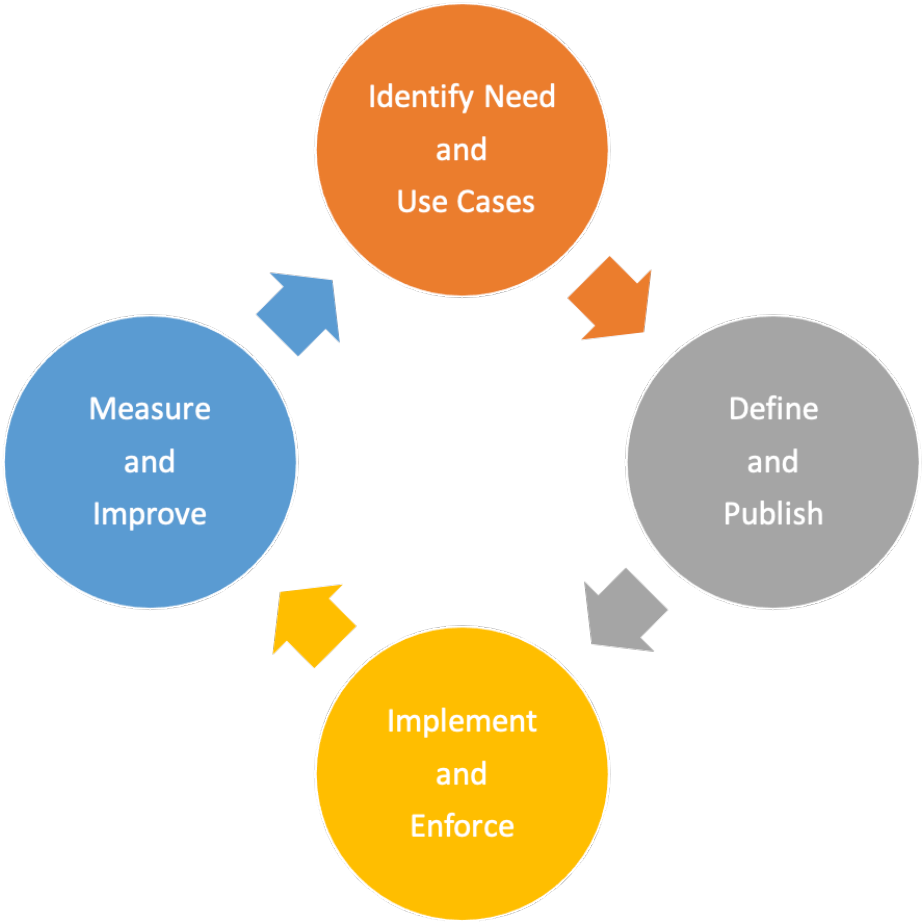
表 5 — 標籤索引鍵內的範圍或使用案例範圍的範例

使用案例	標籤鍵	理由	允許值
資料分類	example-inc:info-sec:data-classification	信息安全定義的數據分類集	sensitive ，company-confidential ，customer-identifiable
操作	example-inc:dev-ops:environment	實作測試與開發環境的排程	development ，staging, quality-assurance ，production
災難復原	example-inc:disaster-recovery:rpo	定義資源的復原點目標 (RPO)	6h, 24h
成本分配	example-inc:cost-allocation:business-unit	財務團隊需要對每個團隊的使用情況和支出進行成本報告	corporate ，recruitment ，support, engineering

標籤很簡單，易於使用。鍵和標籤的值都是可變長度字符串，並且可以支持寬字符集。如需有關長度和字元集的詳細資訊，請參閱AWS一般參考中的標記AWS資源。標籤區分大小寫，這意味著costCenter和costcenter是不同的標籤鍵。在不同國家/地區，單字的拼字可能會有所不同，因此可能會影響您的按鍵。例如，在美國，可能會將金鑰定義為costcenter，但在英國costcentre可能是首選。從資源標記的角度來看，這些是不同的鍵。將拼字、大小寫和標點符號定義為標記策略的一部分。使用這些定義做為建立或管理資源之任何人的參考。本主題將在下一節中更詳細地討論建立您的標記策略。

建立您的標記策略

與許多操作實踐一樣，實施標記策略是迭代和改進的過程。從小規模開始使用您的立即優先級，並根據需要增加標記架構。



標記策略迭代和改進週期

在整個過程中，擁有權是責任和進步的關鍵。由於標籤可用於多種用途，因此整體標籤策略可以分割為組織內的責任區域。標記允許以程式設計方式處理取決於資源特性分析的活動。可以從標記中受益的利益相關者範圍取決於組織的規模和運營實踐。較大的組織可以從明確定義參與建立和實施標記策略的團隊的責任中受益。一些利益相關者可以負責確定標記的需求（定義用例）；其他利益相關者則負責維護，實施和改進標記策略。

通過分配所有權，您可以很好地實施策略的各個方面。在適當的情況下，此所有權可以形式化為政策，並記錄在責任矩陣中 (例如，RACI：負責、可負責、諮詢和知情)，或在共同的責任模型中記錄。在較小的組織中，團隊可能會在標記策略中扮演多個角色，從需求定義到實施和執行。

定義需求和使用案例

透過與具有基本需求使用中繼資料的利益相關者互動，開始建立您的策略。這些團隊定義了資源需要標記的中繼資料，以支援其活動，例如報表、自動化和資料分類。它們概述了資源需要如何組織，以及它們需要對應到哪些策略。這些利益相關者在組織中可以擁有的角色和職能的例子包括：

- 財務和業務營運需要通過將其映射到成本來了解投資的價值，以優先處理解決不公平時需要採取的行動。瞭解產生的成本與價值有助於識別不成功的業務或產品供應項目。這會導致有關持續支持，採用替代方法（例如，使用 SaaS 產品或託管服務）或淘汰無利可圖的業務產品的明智決策。
- 控管與合規需要瞭解資料的分類 (例如，公用、敏感或機密)、特定工作負載是否在特定標準或法規進行稽核的範圍內或超出稽核範圍，以及服務的重要性 (無論服務或應用程式是否關鍵業務)，以套用適當的控制和監督，例如權限、原則和監督。
- 作業與開發需要瞭解工作負載生命週期、其支援產品的實作階段，以及發行階段的管理 (例如，開發、測試、生產分割) 及其相關的支援優先順序，以及相關業者管理需求。還需要定義和理解備份，修補，可觀察性和棄用等職責。
- 資訊安全 (InfoSec) 和安全性作業 (SecOps) 概述了必須套用哪些控制項，以及建議採用哪些控制項。 InfoSec 通常會定義控制項的實作，而且 SecOps 一般負責管理這些控制項。

視您的使用案例、優先順序、組織規模以及營運實務而定，您可能需要組織內各個團隊的代表，例如財務 (包括採購)、資訊安全性、雲端啟用和雲端作業。您還需要應用程式和處理序擁有者提供修補、備份和還原、監視、作業排程和災難復原等功能的代表。這些代表有助於推動標記策略的定義、實施和衡量有效性。他們應該向利益相關者及其用例向後工作，並進行跨職能研討會。在工作坊中，他們有機會分享自己的觀點和需求，並協助推動整體策略。本白皮書稍後將說明參與者的範例及其參與各種使用案例。

利益相關者還定義和驗證強制標籤的鍵，並可以為可選標籤建議範圍。例如，財務團隊可能需要將資源與內部成本中心、業務單位或兩者產生關聯。因此，他們可能需要強制使用某些標籤鍵`BusinessUnit`，例如`CostCenter`和。個別開發團隊可能會決定將其他標籤用於自動化目的`EnvironmentName`，例如`OptIn`、或`OptOut`。

主要利益相關者需要就標記策略方法達成一致，並記錄與合規和治理相關問題的答案，例如：

- 需要解決哪些用例？
- 誰負責標記資源（實施）？
- 標籤如何強制執行，以及將使用哪些方法和自動化（主動或被動）？
- 如何衡量標記的有效性和目標？

- 標記策略應多久檢討一次？
- 誰推動改進？ 這是如何完成的？

如雲端啟用、雲端業務 Oce 和雲端平台工程等業務功能，就可以在建立標記策略的過程中扮演協助者的角色，協助推動其採用，並透過測量進度、消除障礙和減少重複的工作量來確保應用程式的一致性。

定義和發佈標籤資料架構

採用一致的方法來標記您的AWS資源，包括強制性和可選標籤。全面的標記結構描述可協助您達成此一致性。以下範例可幫助您開始：

- 同意強制性標籤鍵
- 定義可接受的值和標籤命名慣例 (大寫或小寫、破折號或底線、階層等)
- 確認值不會構成個人識別資訊 (PII)
- 決定誰可以定義和建立新的標籤關鍵字
- 同意如何新增強制標籤值以及如何管理選用標籤

請檢閱下列標記類別表格，該表格可用作標記結構描述中可能包含內容的基準線。您仍然需要確定您將用於標籤鍵的慣例以及每個標籤鍵允許哪些值。標籤結構描述是您為環境定義此結構描述的文件。

表 6 — 確定標籤結構描述的範例 (第 1 部分)

使用案例	標籤鍵	理由	允許的值 (列出或值首碼/次 x)	用於成本分配	資源類型	範圍	必要
成本分配	example-inc:cost-allocation: ApplicationId	跟踪每個業務線產生的成本與價值	DataLakeX ，RetailSiteX	Y	全部	全部	強制性
成本分配	example-inc:cost-allocation: BusinessUnitId	按業務單位監控成本	Architecture ，DevOps, Finance	Y	全部	全部	強制性
成本分配	example-inc:cost-allocation:CostCenter	按成本中心監控成本	123-*	Y	全部	全部	強制性
成本分配	example-inc:cost-allocation:Owner	哪個預算持有人負責此工作負載	Marketing ，RetailSupport	Y	全部	全部	強制性
存取控制	example-inc:access-control:LayerId	識別 SubComponent /分層以根據角色授予對資源的訪問權限	DB_Layer, Web_Layer ，App_Layer	否	全部	全部	選用
自動化	example-i	實施測試和開發	Prod, Dev,	否	EC2,	全部	強制性

表 6 — 確定標籤結構描述的範例 (第 2 部分)

使用案例	標籤鍵	理由	允許的值 (列出或值首碼/次 x)	用於成本分配	資源類型	範圍	必要
DevOps	example-inc:operations: Owner	哪個團隊/小隊負責資源的創建和維護	Squad01	否	全部	全部	強制性
災難復原	example-inc:disaster-recovery:rpo	定義資源的復原點目標 (RPO)	6h, 24h	否	S3 , EBS	生產	強制性
資料分類	example-inc:data:classification	對資料進行分類以達到合規性	Public, Private, Confidential , Restricted	否	S3 , EBS	全部	強制性
合規	example-inc:compliance:framework	識別工作負載受限制的合規性架構	PCI-DSS, HIPAA	否	全部	生產	強制性

定義標記結構描述後，請在版本控制的儲存庫中管理結構描述，該儲存庫可供所有相關利益相關者存取，以便於參考和追蹤更新。這種方法提高了效率並允許敏捷性。

AWS Organizations— 標籤政策

中的原則AWS Organizations允許您應用其他類型的管理。AWS 帳戶標籤原則是您如何以 JSON 格式表示標記結構描述，以便平台可以在您的AWS環境中報告並選擇性地強制執行結構描述。標籤策略定義了特定資源類型上標籤鍵可接受的值。此原則可以是值清單的形式，也可以是前置字元後跟萬用字元(*)。簡單的前綴方法比離散的值列表不太嚴格，但需要較少的維護。

下列範例顯示如何定義標記原則，以驗證指定索引鍵可接受的值。從結構描述的人性化表格定義中工作，您可以將此資訊轉錄為一或多個標籤原則。您可以使用個別原則來支援委派的擁有權，或者某些原則可能只適用於特定案例。

ExampleInc-CostAllocation. JSON

以下是報告成本配置標籤的標籤原則範例：

```
{
  "tags": {
    "example-inc:cost-allocation:ApplicationId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:ApplicationId"
      },
      "tag_value": {
        "@@assign": [
          "DataLakeX",
          "RetailSiteX"
        ]
      }
    },
    "example-inc:cost-allocation:BusinessUnitId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:BusinessUnitId"
      },
      "tag_value": {
        "@@assign": [
          "Architecture",
          "DevOps",
          "FinanceDataLakeX"
        ]
```

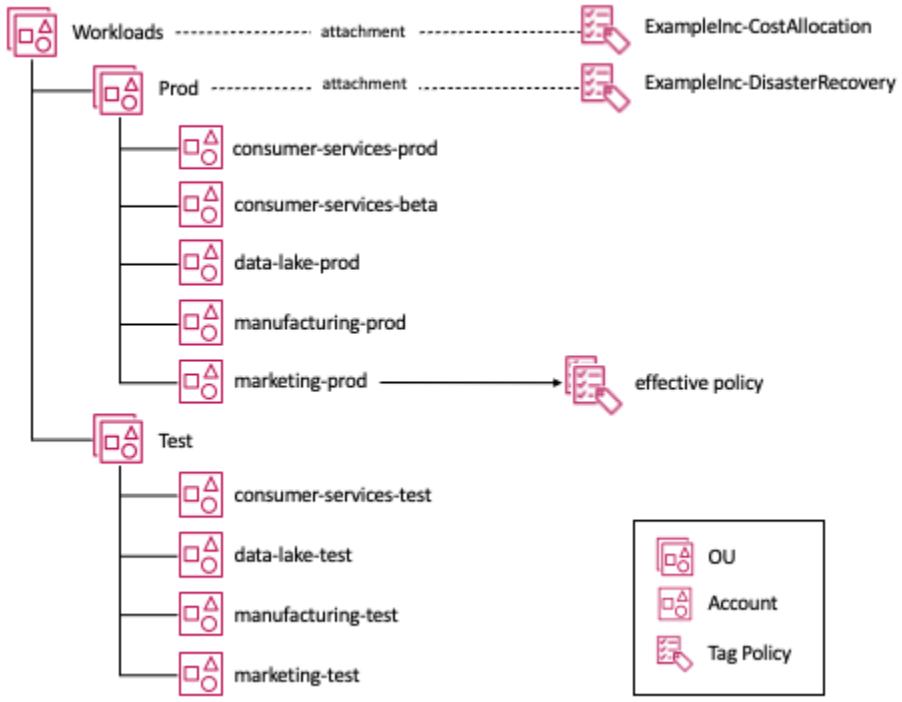
```
      }
    },
    "example-inc:cost-allocation:CostCenter": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:CostCenter"
      },
      "tag_value": {
        "@@assign": [
          "123-*"
        ]
      }
    }
  }
}
```

ExampleInc-DisasterRecovery. JSON

以下是報告嚴重損壞修復標籤的標籤原則範例：

```
{
    "tags": {
        "example-inc:disaster-recovery:rpo": {
            "tag_key": {
                "@@assign": "example-inc:disaster-recovery:rpo"
            },
            "tag_value": {
                "@@assign": [
                    "6h",
                    "24h"
                ]
            }
        }
    }
}
```

在此範例中，ExampleInc-CostAllocation標籤原則會附加至 Workloads OU，因此適用於Prod和Test子 OU 中的所有帳戶。同樣地，標ExampleInc-DisasterRecovery籤原則會附加至 Prod OU，因此僅適用於此 OU 以下的帳戶。使用多個帳戶組織環境白皮書更詳細地探討建議的 OU 結構。



將標籤原則附加至 OU 結構

查看圖表中的marketing-prod帳戶，這兩個標籤策略都適用於此帳戶，因此我們有一個有效策略的概念，這是指定類型的策略適用於帳戶的卷積。如果您主要是手動管理資源，則可以造訪主控台中的Resource Groups 和標籤編輯器:標籤原則，以檢閱有效的政策。如果您使用基礎結構即程式碼 (IaC) 或指令碼來管理資源，您可以使用 AWS::Organizations::DescribeEffectivePolicyAPI 呼叫。

實施和強制標記

在本節中，我們將向您介紹下列資源管理策略可用的工具：手動、基礎結構即程式碼 (IaC) 和持續整合/持續傳遞 (CI/CD)。這些方法的關鍵維度是部署的頻率越來越頻繁。

手動管理資源

這些工作負載通常屬於採用的基礎或移轉階段。這些工作負載通常是簡單的大部分靜態工作負載，是使用傳統的書面程序建立的，或是使用內部部署環境等工具進行移轉的工作 CloudEndure 負載。遷移工具 (例如 Migration Hub 和 CloudEndure) 可以套用標籤做為移轉程序的一部分。但是，如果在原始移轉期間未套用標籤，或從那時起標籤結構描述發生變更，則標籤編輯器 (的功能AWS Management

Console) 可讓您使用各種搜尋條件搜尋資源，並大量新增、修改或刪除標籤。搜尋條件可以包括具有或不存在特定標籤或值的資源。資AWS源標記 API 允許您以編程方式執行這些功能。

由於這些工作負載已現代化，因此會引入 Auto Scaling 群組等資源類型。這些資源類型允許更大的彈性和改善彈性。auto 動擴展群組會代表您管理 Amazon EC2 執行個體，但是，您可能仍希望 EC2 執行個體與手動建立的資源一致地標記。Amazon EC2 啟動範本提供指定 Auto Scaling 應套用至其建立執行個體的標籤的方法。

手動管理工作負載的資源時，自動化資源標記會很有幫助。有各種解決方案可用。一種方法是使用 AWS Config 規則，它可以檢查`required_tags`並啟動 Lambda 函數以應用它們。 AWS Config 規則本白皮書稍後會詳細說明。

基礎架構即程式碼 (IaC) 受管資源

AWS CloudFormation提供一種通用語言來佈建您AWS環境中的所有基礎結構資源。 CloudFormation範本是以自動方式建立AWS資源的 JSON 或 YAML 檔案。使用 CloudFormation範本建立AWS資源時，您可以使用 CloudFormation Resource Tags 屬性在建立時將標籤套用至支援的資源類型。使用 IaC 管理標籤和資源有助於確保一致性。

當資源由建立時AWS CloudFormation，服務會自動將一組AWS已定義的標籤套用至AWS CloudFormation範本所建立的資源。這些是：

```
aws:cloudformation:stack-name
aws:cloudformation:stack-id
aws:cloudformation:logical-id
```

您可以根據AWS CloudFormation堆疊輕鬆定義資源群組。這些AWS定義的標籤會由堆疊建立的資源繼承。但是，對於 Auto Scaling 群組中的 Amazon EC2 執行個體，則`AWS::AutoScaling::AutoScalingGroup TagProperty`需要在AWS CloudFormation範本中的「Auto Scaling」群組的定義中進行設定。或者，如果您將 EC2 啟動範本與 Auto Scaling 群組搭配使用，則可以在其定義中定義標籤。建議將 EC2 啟動範本與 Auto Scaling 群組搭配使用，或搭配AWS容器服務使用。這些服務可協助確保 Amazon EC2 執行個體標記的一致性，並支援跨多個執行個體類型的 Auto Scaling 和購買選項，進而提升彈性並最佳化運算成本。

AWS CloudFormationHook 為開發人員提供了一種方法，讓其應用程式的關鍵層面與其組織的標準保持一致。勾點可以設定為提供警告或防止部署。此功能最適合檢查範本中的關鍵組態元素，例如 Auto Scaling 群組是否設定為將啟動的所有 Amazon EC2 執行個體套用客戶定義的標籤，或確保所有 Amazon S3 儲存貯體都使用必要的加密設定建立。在這兩種情況下，在部署之前，都會使用AWS CloudFormation掛接將此相容性的評估推送至較早的部署程序。

AWS CloudFormation提供了檢測從模板佈建的資源（請參閱支持漂移檢測的資源）何時被修改，並且資源不再匹配其預期的模板配置的功能。這就是所謂的漂移。如果您使用自動化將標籤應用於通過 IaC 管理的資源，那麼您正在修改它們，引入漂移。當使用 IaC，目前建議管理任何標記要求作為 IaC 模板的一部分，實現AWS CloudFormation掛鉤，並發布可以通過自動化使用的AWS CloudFormation保護規則集。

CI/CD 管道的管理資源

隨著工作負載的成熟度增加，很可能會採用持續整合和持續部署 (CI/CD) 等技術。這些技術有助於降低部署風險，藉由提高測試的自動化功能，讓您更容易更頻繁地部署小型變更。偵測部署引入的非預期行為的可觀察性策略可以自動回復部署，而且對使用者的影響最小。當您進入每天部署數十次的階段時，以追溯方式套用標籤已經不再實際了。一切都需要表示為代碼或配置，版本控制，並在可能的情況下，在部署到生產環境之前進行測試和評估。在合併的開發和操作 (DevOps) 模型中，許多實務將操作考量作為代碼進行解決，並在部署生命週期的早期驗證它們。

理想情況下，您希望盡可能早地在程序中推送這些檢查 (如AWS CloudFormation鉤子所示)，這樣您就可以確信您的AWS CloudFormation範本在離開開發人員的電腦之前符合您的政策。

AWS CloudFormationGuard 2.0 提供了為您可以定義的任何內容編寫預防性合規規則的 CloudFormation方法。範本會根據開發環境中的規則進行驗證。顯然，此功能具有一系列應用程序，但在本白皮書中，我們將只看一些可以確保始終使用的示例。`AWS::AutoScaling::AutoScalingGroup TagProperty`

以下是 CloudFormation 守門員規則的範例：

```
let all_asgs = Resources.*[ Type == 'AWS::AutoScaling::AutoScalingGroup' ]

rule tags_asg_automation_EnvironmentId when %all_asgs !empty {
    let required_tags = %all_asgs.Properties.Tags.*[
        Key == 'example-inc:automation:EnvironmentId' ]
    %required_tags[*] {
        PropagateAtLaunch == 'true'
        Value IN ['Prod', 'Dev', 'Test', 'Sandbox']
        <<Tag must have a permitted value
          Tag must have PropagateAtLaunch set to 'true'>>
    }
}

rule tags_asg_costAllocation_CostCenter when %all_asgs !empty {
    let required_tags = %all_asgs.Properties.Tags.*[
        Key == 'example-inc:cost-allocation:CostCenter' ]
```

```
    %required_tags[*] {
        PropagateAtLaunch == 'true'
        Value == /^123-/
        <<Tag must have a permitted value
          Tag must have PropagateAtLaunch set to 'true'>>
    }
}
```

在程式碼範例中，我們篩選該類型的所有資源的範本AutoScalingGroup，然後有兩個規則：

- **tags_asg_automation_EnvironmentId**-檢查具有此索引鍵的標籤是否存在、在允許的值清單中具有值，PropagateAtLaunch且設定為 true
- **tags_asg_costAllocation_CostCenter**-檢查此索引鍵PropagateAtLaunch是否存在標籤、具有以定義前置字元值開頭的值，且設定為 true

執法

如前所述，Resource Groups 與標籤編輯器提供了識別資源無法符合套用至組織 OU 之標籤原則中定義的標籤需求的方法。從組織成員帳號內存取 Resource Groups 與標籤編輯器主控台工具，會顯示套用至該帳號的策略，以及無法符合標籤策略需求的帳號內的資源。如果從管理帳戶存取 (且如果標籤策略在下的服務中啟用了存取權AWS Organizations)，則可以檢視組織中所有連結帳號的標籤策略符合性。

在標籤原則本身中，您可以針對特定資源類型啟用強制執行功能。在下列原則範例中，我們新增了強制執行功能，使所有類型ec2:instance的資源ec2:volume都必須符合原則。有一些已知的限制，例如資源上必須有標籤，才能由標籤策略評估。如需清單，請參閱支援使用標籤政策強制執行的資源。

ExampleInc-成本分配

以下是報告和/或強制執行成本配置標籤的標籤策略範例：

```
{
  "tags": {
    "example-inc:cost-allocation:ApplicationId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:ApplicationId"
      },
      "tag_value": {
        "@@assign": [
          "DataLakeX",
          "RetailSiteX"
```

```
        ]
      },
      "enforced_for": {
        "@@assign": [
          "ec2:instance",
          "ec2:volume"
        ]
      }
    },
    "example-inc:cost-allocation:BusinessUnitId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:BusinessUnitId"
      },
      "tag_value": {
        "@@assign": [
          "Architecture",
          "DevOps",
          "FinanceDataLakeX"
        ]
      },
      "enforced_for": {
        "@@assign": [
          "ec2:instance",
          "ec2:volume"
        ]
      }
    },
    "example-inc:cost-allocation:CostCenter": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:CostCenter"
      },
      "tag_value": {
        "@@assign": [
          "123-*"
        ]
      },
      "enforced_for": {
        "@@assign": [
          "ec2:instance",
          "ec2:volume"
        ]
      }
    }
  }
```

```
}
```

AWS Config (**`required_tag`**)

AWS Config是允許您存取、稽核和評估AWS資源的組態 (請參閱支援的資源類型 AWS Config)。在進行標記的情況下，我們可以使用該規則來使用該`required_tags`規則來識別缺少具有特定鍵標籤的資源（請參閱 required_tags 支持的資源類型）。從前面的範例中，我們可能會測試金鑰是否存在於所有 Amazon EC2 執行個體上。在金鑰不存在的情況下，執行個體將會註冊為不相容。此AWS CloudFormation範本說明用於測試表格、Amazon S3 儲存貯體、Amazon EC2 執行個體和 Amazon EBS 磁碟區是否存在強制金鑰的AWS Config規則。

```
Resources:
 MandatoryTags:
   Type: AWS::Config::ConfigRule
   Properties:
     ConfigRuleName: ExampleIncMandatoryTags
     Description: These tags should be in place
     InputParameters:
       tag1Key: example-inc:cost-allocation:ApplicationId
       tag2Key: example-inc:cost-allocation:BusinessUnitId
       tag3Key: example-inc:cost-allocation:CostCenter
       tag4Key: example-inc:automation:EnvironmentId
     Scope:
       ComplianceResourceTypes:
       - "AWS::S3::Bucket"
       - "AWS::EC2::Instance"
       - "AWS::EC2::Volume"
     Source:
       Owner: AWS
       SourceIdentifier: REQUIRED_TAGS
```

對於手動管理資源的環境，可以增強AWS Config規則，使用自動補救功能透過功能自動補救AWS Lambda功能，將遺失的標籤金鑰自動新增至資源。雖然這適用於靜態工作負載，但是當您開始透過 IAC 和部署管道管理資源時，效率逐漸降低。

AWS Organizations— 服務控制原則 (SCP) 是一種組織原則，以便您管理組織權限。SCP 可讓您集中控制組織或組織單位 (OU) 所有帳戶的最大可用權限。SCP 只會影響由組織的帳戶所管理的使用者和角色。雖然它們不會直接影響資源，但它們會限制使用者和角色的權限，其中包括標記動作的權限。關於標記，除了標籤原則可提供的標籤原則之外，SCP 還可以提供額外的標籤強制執行細微性。

在下列範例中，政策將拒絕example-inc:cost-allocation:CostCenter標籤不存在的ec2:RunInstances要求。

以下是拒絕 SCP：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyRunInstanceWithNoCostCenterTag",
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:*:*:instance/"
      ],
      "Condition": {
        "Null": {
          "aws:RequestTag/example-inc:cost-allocation:CostCenter": "true"
        }
      }
    }
  ]
}
```

無法根據設計擷取套用至連結帳戶的有效服務控制政策。在您使用 SCP 強制標記的情況下，開發人員必須提供文件，以確保他們的資源符合已套用至其帳戶的政策。提供帳戶內 CloudTrail 事件的唯讀存取權，可支援開發人員在資源無法遵守時執行除錯工作。

衡量標記有效性和推動改進

實作標記策略之後，請務必根據目標使用案例來衡量其有效性。有效性的衡量將因使用案例而異。例如：

- 成本歸因-您可以使用諸如「成本和用量報告」等工具，根據支出來衡量資源的標記涵蓋範圍。AWS Cost ExplorerAWS例如，您可以追蹤產生費用的已標記或未標記資源百分比，尤其是監控特定標籤金鑰。
- 自動化-您可能想要稽核是否已達到所需的結果。例如，非生產 Amazon EC2 執行個體是否在營業時間以外暫停，稽核執行個體的開始和停止時間。

管理帳號中的 Resource Groups 與標籤編輯器提供額外功能，可分析組織中所有連結帳號的標籤原則符合性。

根據標記有效性的測量結果，識別是否需要在任何步驟 (例如使用案例定義、標記結構描述實作或強制執行) 中進行任何改進或變更。進行必要的更改並重複週期，直到達到所需的效果。在具有成本歸因的範例中，您可以查看百分比改善情況。

由於開發人員和運營商需要執行資源的實際標記，因此讓他們擁有所有權至關重要。這不是團隊在AWS採用過程中通常承擔的唯一額外責任。對於開發和操作其應用程序的安全性和成本的增加責任也很重要。Organizations 通常使用目標和目標作為激勵採用新做法的手段，因此這也可以在這裡適用。

標籤使用案例

主題

成本分配和財務管理標籤

組織經常遇到的第一個標記使用案例之一是成本和使用的可見性和管理。這種情況通常有幾個原因：

- 這通常是一個很好理解的場景和要求是眾所周知的。例如，財務團隊希望查看跨多項服務、功能、帳戶或團隊的工作負載和基礎架構的總成本。實現此成本可見性的一種方法是通過對資源進行一致標記。
- 標籤及其值得清楚定義。通常，成本配置機制已存在於組織的財務系統中，例如，依成本中心、業務單位、小組或組織職能進行追蹤。
- 快速、明顯的投資回報率。當資源一致地標記時，可以追蹤一段時間內的成本最佳化趨勢，例如，針對大小適中、自動調整規模或按排程進行的資源。

了解如何產生成本，您AWS可以做出明智的財務決策。瞭解您在資源、工作負載、專案團隊或組織層級產生成本的位置，可讓您瞭解相較於達成的業務成果，在適用層級提供的價值。

工程團隊可能沒有資源財務管理的經驗。聯繫一個具有AWS財務管理專業技能的人員，他們可以對工程和開發團隊進行AWS財務管理基礎知識培訓，並在財務和工程之間建立關係，以促進 FinOps 將有助於實現可衡量的業務成果，並鼓勵團隊考慮到成本。Well-Architected 的框架的成本優化支柱深入介紹了建立良好的財務實踐，但我們將討論本白皮書中的一些基本原則。

成本分配標籤

成本配置是指在定義的處理之後，將發生的成本指定或分配給這些成本的使用者或受益者。在本白皮書中，我們將成本分配分為兩種類型：反對和退款。

回顧工具和機制有助於提高成本意識。退款有助於回收成本，並推動成本感知的啟用。「回溯」是關於特定實體 (例如業務單位、應用程式、使用者或成本中心) 所產生的費用的簡報、計算及報告。例如：「基礎架構工程團隊負責上個月 X 美元的AWS支出」。借項沖回是指透過組織的內部會計處理，例如

財務系統或分錄傳票，對這些實體產生的成本實際收取費用。例如：「從基礎設施工程團隊的AWS預算中扣除了 X 美元。」 在這兩種情況下，適當地標記資源可以幫助將成本分配給實體，唯一的區別在於是否有人需要付款。

貴組織的財務治理可能需要透明地核算應用程式、業務單位、成本中心和團隊層級產生的成本。執行成本配置標籤支援的成本歸因可為您提供必要的資料，以準確歸因實體從適當標記的資源產生的成本。

- 責任 — 確保將成本分配給負責資源使用的人員。單一服務點或群組可負責支出審查和報告。
- 財務透明度 — 透過為領導層建立有效的儀表板和有意義的成本分析，顯示對 IT 現金分配的清晰視野。
- 明智的 IT 投資 — 根據專案、應用程式或業務線追蹤投資報酬率，並讓團隊做出更好的業務決策，例如為創造收入的應用程式分配更多資金。

總而言之，成本分配標籤有助於告訴您：

- 誰擁有支出，並負責優化它？
- 哪些工作負載、應用程式或產品會產生花費？ 哪個環境或階段？
- 哪些消費領域增長最快？
- 根據過去的趨勢，可以從AWS預算中扣除多少支出？
- 在特定工作負載、應用程式或產品中，成本最佳化工作會產生什麼影響？

啟用成本分配的資源標籤有助於定義組織內的測量實務，這些實務可用於提供使用AWS情況的可見性，從而提高支出責任的透明度。它還著重於在成本和用量可見性方面創建合適的粒度級別，並通過成本分配報告和 KPI 跟踪來研究雲消費行為。

建立成本分配策略

定義與導入成本配置模型

為中部署的資源建立帳戶和成本結構AWS。建立AWS支出成本、如何產生這些成本，以及發生這些成本的人或原因之間的關係。常見的成本結構是以組織內的、、環境和實體為基礎，例如業務單位或工作負載。AWS Organizations AWS 帳戶成本結構可以基於多個屬性，以允許以不同的方式或在不同的粒度層次檢查成本，例如將個別工作負載的成本累計到其所服務的業務範圍。

當選擇符合所需結果的成本結構時，請評估成本分配機制，以便於實施與所需的準確性。這可能包括有關責任、工具可用性和文化變更的考量。AWS客戶通常從三種常見的成本分配模式開始：

- 以帳戶為基礎 — 此模型需要最少的精力，而且對於回撥和退款提供高準確度，且適用於具有已定義帳戶結構 (且與「使用多個帳戶組織AWS環境」白皮書中的建議一致) 的組織。這可以根據每個帳戶提供明確的成本可見性。對於成本可見性和配置，您可以使用 AWS Cost Explorer「成本和用量報告」以及「AWS預算」進行成本監控和追蹤。這些工具提供篩選和分組選項的依據AWS 帳戶。從成本分配的角度來看，此模型不需要依賴個別資源的準確標記。
- 業務單位或團隊型 — 可分配給企業內團隊、業務單位或組織的成本。此模型需要適量的努力，提供反對和借項沖回的高準確性，並且適用於具有已定義帳戶結構 (通常使用AWS Organizations) 且在不同團隊、應用程式和工作負載類型之間有區隔的組織。這提供了跨團隊和應用程序的清晰成本可見性，並作為額外的好處降低單一AWS服務配額的風險AWS 帳戶。例如，每個團隊可能有五個帳戶 (`prod`、、`staging`、`sandbox`) `testdev`，而且沒有兩個團隊和應用程式會共用同一個帳戶。然後，使用這種結構，「AWSCost Categories」將提供將帳戶或其他標籤 (「中繼標記」) 分組到類別中的功能，這些類別可以在上一個範例中提到的工具中進行追蹤。請務必注意，AWS Organizations允許標記帳戶和組織單位 (OU)，不過這些標記不適用於成本分配和帳單報告 (也就是說，您無法AWS Cost Explorer依 OU 分組或篩選成本)。 AWSCost Categories 應該用於此目的。
- 基於標籤-與前兩種模式相比，該模型需要更多的努力，並且將根據要求和最終目標為反對和退款提供高準確性。雖然我們強烈建議您採用使用多個帳戶組織AWS環境白皮書中概述的做法，但實際上，客戶往往會發現自己擁有混合且複雜的帳戶結構，需要花費一些時間才能遷移。在此案例中，實施嚴格有效的標記策略是關鍵，然後在「帳單與成本管理」主控台中啟用成本分配的相關標籤 (在中 AWS Organizations，只能從 Management Partners 帳戶啟動標籤以進行成本配置)。在針對成本分配啟動標籤之後，先前方法中提及的成本可見度與配置工具可用於回覆和借項沖回。請注意，成本分攤標記並非追溯性，而且只會在針對成本分攤啟動後，才會出現在帳單報表與成本追蹤工具中。

總而言之，如果您需要依業務單位追蹤成本，您可以使用「AWSCost Categories」，對「組AWS織」內的連結帳戶進行分組，並在帳單報表中檢視此群組。當您針對生產環境和非生產環境建立不同的帳戶時，您也可以在工具 (例如) 中篩選與環境相關的成本 AWS Cost Explorer，或使用「AWS預算」追蹤這些成本。最後，如果您的使用案例需要更精細的成本追蹤 (例如依個別工作負載或應用程式)，您可以相應地標記這些帳戶內的資源，在管理帳戶上啟用這些標籤金鑰，以便在管理帳戶上分配成本，然後在帳單報告工具中依標籤金鑰篩選該成本。

建立成本報告和監控流程

從確定對內部利益相關者而言很重要的成本類型開始 (例如，每日支出、按帳戶分類的成本、按 X 計算的成本、攤銷成本)。這樣，您可以比等待最終AWS發票更快地減輕與意外或異常支出相關的預算風險。標籤提供啟用這些報告案例的歸因。從報告中獲得的見解可以為您的行動提供信息，以減輕異常和意外支出對財務預算的影響。當成本出現意外激增時，重要的是評估交付的價值是否出現意外激增，以便您可以確定是否需要採取什麼措施。

在開發標籤政策來支援成本分配時，請記住下列要素：

- AWS Organizations-多個帳戶中的成本分配可以按帳戶，帳戶組或為這些帳戶上的資源創建的標籤組執行。針對位於中個別帳戶中的資源建立的標籤，只AWS Organizations能用於管理帳戶的成本分配。
- AWS帳戶-一個帳戶中的成本分配AWS 帳戶可以由其他維度（例如服務或區域）執行。您可以進一步標記帳號內的資源，並使用此類資源標籤的群組。
- 成本配置標籤-如有需要，可啟動使用者建立的標籤和AWS產生的標籤以進行成本配置。在帳單主控台中啟用成本分配標籤 (在中的管理帳戶AWS Organizations) 有助於進行反對和退款。
- Cost Categories-「AWSCost Categories」允許在「組AWS織」中將帳戶與群組標記（「中繼標記」）分組，此功能進一步提供了透過工具 (例如 AWS Cost Explorer「AWS預算」與「成本與使用量報表」) 來分析與這些分類相關的AWS成本。

為企業內的業務單位、團隊或組織執行回收和借項沖回

使用成本結構與成本配置標籤所支援的成本分配程序來屬性成本。標籤可用於向團隊提供回報，這些團隊不直接負責支付費用，但負責產生這些成本。這種方法提供了他們對支出的貢獻以及如何產生這些成本的認識。對直接負責成本的團隊執行借項沖回，以收回他們消耗的資源費用，並讓他們了解這些成本以及如何產生這些成本。

測量和流通性或價值 KPI

同意一組單位成本或 KPI 指標，以衡量雲端財務管理投資的影響。這個練習創造了跨越技術和業務利益相關者的共同語言，並講述了一個以電子學為基礎的故事，而不是一個專注於絕對、總支出的故事。有關其他信息，請查看此博客，該博客討論單位指標如何幫助在業務功能之間建立一致。

分配不可分配的支出

根據組織的會計慣例，不同的收費類型可能需要不同的處理方式。識別無法標記的資源或成本類別。根據所使用的服務和計劃使用的服務，就如何處理和衡量此類不可分配開支出的機制達成一致。例如，檢查AWS Resource Groups和「標籤使用指南」中的「標籤編輯器」支援的資源清單，以AWS Resource Groups及「標籤編輯器」。

無法標記成本類別的常見範例是承諾折扣的一些費用，例如預留執行個體 (RI) 和 Savings Plans (SP)。雖然訂閱費用和未使用的 SP 和 RI 費用無法在帳單報告工具中顯示之前加上標記，但您可以在事AWS Organizations後追蹤 RI 和 SP 折扣如何套用至帳戶、資源及其標籤。例如，可AWS Cost Explorer以查看攤銷成本，按相關標籤鍵支出的分組並應用與您的使用案例相關的篩選器。在「AWS成本與用量報告」(CUR) 中，您可以篩選出與 RI 和 SP 折扣涵蓋之使用情況相對應的明細行 (請參閱

CUR 文件的使用案例一節中的詳細資訊)，並選取僅與您相關的欄。每個針對成本分配啟動的標籤金鑰，都會顯示在 CUR 報告末尾的個別欄中，與其他舊版帳單報告 (例如每月成本分配報表) 中的呈現方式類似。如需其他參考資料，請參閱 AWSWell-Architected 的實驗室，以取得從 CUR 資料取得成本和使用情況深入分析的範例。

報告

除了可用於協助回撥和退款的AWS工具之外，還有一系列其他AWS建立的第三方解決方案，可協助監控已標記資源的成本，並衡量標記策略的有效性。根據組織的需求和最終目標，可以投入時間和資源來建立自訂的解決方案，或購買AWS 雲端管理工具能力合作夥伴所提供的工具。如果您決定使用與業務相關的受控參數來建立自己的單一事實成本分配工具，AWS成本和用量報告 (CUR) 會提供最詳細的成本和使用情況資料，並可建立自訂的最佳化儀表板，允許依帳戶、服務、成本類別、成本分配標籤和多個其他維度進行篩選和分組。在由開發的基於 CURO 的解決方案中AWS，可以用作這些工具之一，請查看 AWS Well-Architected 的實驗室網站上的雲端智慧儀表板。

操作和支持標籤

一個AWS環境將具有多個帳戶、資源和工作負載，具有不同的操作需求。標籤可用於提供背景資訊和指引，以支援營運團隊，以加強對服務的管理。標籤也可用於提供受管資源的營運治理透明度。

驅動操作標籤一致定義的一些主要因素包括：

- 在自動化基礎架構活動期間篩選資源。例如，部署、更新或刪除資源時。另一個是擴展資源以進行成本優化和非工作時間的使用減少。如需工作範例，請參閱AWS執行個體排程器解決
- 識別隔離或棄用的資源。應適當標記已超過其定義壽命的資源，或已被內部機制標記為隔離的資源，以協助支援人員進行調查。在隔離、封存和刪除之前，應標記棄用資源。
- 一組資源的 Support 需求。資源通常具有不同的支援需求，例如，這些需求可以在團隊之間進行協商或設定為應用程式關鍵性的一部分。有關營運模式的進一步指引，請參閱卓越營運支柱。
- 加強事件管理流程。透過標籤來標記資源，以提高事件管理流程的透明度，支援團隊和工程師以及重大事件管理 (MIM) 團隊可以更有效地管理事件。
- 備份。標籤也可用於識別資源需要備份的頻率，以及備份副本需要去的位置或還原備份的位置。上AWS的 Backup 和復原方法的規範性指引。
- 修補。修補在中AWS執行的可變執行個體對於整體修補策略和零時差弱點的修補至關重要。您可以在規範性指引中找到更深入的修補策略指引。此部落格討論零時差弱點的修補程式。
- 操作可觀察性。將營運重要績效指標策略轉換為資源標籤，將有助於營運團隊更好地追蹤是否符合目標，以提高業務需求。制定 KPI 策略是一個單獨的主題，但往往集中在穩定狀態下運營的業務或

衡量變化的影響和結果的地方。KPI 儀表板 (W AWS ell-Architected 的實驗室) 和營運 KPI 研討會 (AWS企業 Support 主動式服務) 都在穩定狀態下解決衡量效能的問題。AWS企業策略部落格文章「衡量您轉型的成功」，探索轉型計畫的 KPI 測量，例如 IT 現代化或從內部部署移轉至。AWS

自動化基礎架構

管理基礎結構時，標籤可用於廣泛的自動化活動。例如，使用 AWSSystems Manager 可讓您針對您建立的已定義索引鍵值組所指定的資源管理自動化和執行手冊。對於受管節點，您可以定義一組標籤，以依作業系統和環境來追蹤節點或鎖定節點。然後，您可以針對群組中的所有節點執行更新指令碼，或檢閱這些節點的狀態。Systems Manager 資源也可以加上標籤，以進一步調整和追蹤您的自動化活動。

自動化環境資源的開始和停止生命週期，可大幅降低任何組織的成本。執行個體排程器AWS是一個解決方案範例，可在不需要 Amazon EC2 和 Amazon RDS 執行個體時啟動和停止這些執行個體。例如，使用 Amazon EC2 或 Amazon RDS 執行個體的開發人員環境不需要在週末執行，並不會利用關閉這些執行個體所能提供的成本節省潛力。透過分析團隊及其環境的需求，並適當標記這些資源以自動化其管理，您可以有效地利用預算。

Amazon EC2 執行個體上執行個體排程器使用的排程標籤範例：

```
{
    "Tags": [
        {
            "Key": "Schedule",
            "ResourceId": "i-1234567890abcdef8",
            "ResourceType": "instance",
            "Value": "mon-9am-fri-5pm"
        }
    ]
}
```

工作負載生

審查支持操作數據的準確性。請確定定期檢閱與您的工作負載生命週期相關聯的標籤，而且這些檢閱中涉及適當的利益相關者。

表 7 — 檢閱作業標籤，做為工作負載生命週期的一部分

使用案例	標籤鍵	理由	範例數值
帳戶擁有者	example-inc:account-owner:owner	該帳戶的所有者和它包含的資源。	ops-center ，dev-ops, app-team
帳戶擁有者審查	example-inc:account-owner:review	審查帳戶所有權詳細信息是最新的和正確的。	<review date in the correct format defined in your tagging library>
資料擁有者	example-inc:data-owner:owner	存放資料之帳戶的資料擁有者。	bi-team, logistics ，security
資料擁有者檢閱	example-inc:data-owner:review	審查數據所有權詳細信息是最新的和正確的。	<review date in the correct format defined in your tagging library>

在移轉至暫停的 OU 之前，指派標記以暫停帳戶

在暫停帳戶並移入暫停的 OU (如使用多個帳戶組織您的AWS環境) 白皮書中所述，應該將標籤新增至帳戶，以協助您在內部追蹤和稽核帳戶的生命週期。例如，組織 ITSM 票務系統上的相對 URL 或票證參考，顯示暫停應用程式的稽核記錄。

表 8-當工作負載生命週期進入新階段時，新增操作標籤

使用案例	標籤鍵	理由	範例數值
帳戶擁有者	example-inc:account-owner:owner	該帳戶的所有者和它包含的資源。	ops-center ，dev-ops, app-team
資料擁有者	example-inc:data-owner:owner	存放資料之帳戶的資料擁有者。	bi-team, logistics ，security

使用案例	標籤鍵	理由	範例數值
暫停日期	example-inc:suspension:date	帳戶被暫停的日期	<suspended date in the correct format defined in your tagging library>
暫停批准	example-inc:suspension:approval	鏈接到帳戶暫停批准	workload/deprecation

事件管理

從事件記錄、排定優先順序、調查、溝通、解決到關閉，標籤在事件管理的所有階段都扮演著至關重要的角色。

標籤可以詳細說明應記錄事件的位置、應該通知事件的小組或團隊，以及定義的上報優先順序。重要的是要記住標籤沒有加密，因此請考慮您在其中存儲的信息。此外，在組織、團隊和報告行中，職責會發生變化，因此請考慮儲存一個安全入口網站的連結，以便更有效地管理這些資訊。這些標籤不需要是排他性的。例如，應用程式識別碼可用來查詢 IT 服務管理入口網站中的呈報路徑。確保在操作定義中清楚地表明此標籤用於多種用途。

操作需求標籤也可以詳細說明，以幫助事件經理和運營人員進一步完善其目標以響應事件或事件。

手冊和教戰手冊的相對鏈接（指向知識系統庫 URL）可以作為標籤包含，以幫助響應團隊識別相應的過程，程序和文檔。

表 9-使用操作標籤來通知事件管理

使用案例	標籤鍵	理由	範例數值
事件管理	example-inc:incident-management:escalationlog	支持團隊正在使用的系統記錄事件	jira, servicenow ，zendesk
事件管理	example-inc:incident-	升級的路徑	ops-center ，dev-ops, app-team

使用案例	標籤鍵	理由	範例數值
	management:escalationpath		
成本分配與事件管理	example-inc:cost-allocation:CostCenter	按成本中心監控成本。這是雙重用途標籤的範例，其中成本中心用作事件記錄的應用程式程式碼	123-*
Backup 排程	example-inc:backup:schedule	資源的 Backup 排程	Daily
教戰手冊/事件管理	example-inc:incident-management:playbook	記錄的劇本	webapp/incident/playbook

修補

Organizations 可以使用 AWS Systems Manager 修補程式管理員和.，自動執行可變計算環境的修補策略，並將可變執行個體與該應用程式環境的定義修補程式基準保持一致AWS Lambda。這些環境中可變動執行個體的標記策略可透過將上述執行個體指派給修補程式群組和維護時段來管理。有關開發 → 測試 → 生產分割，請參閱以下示例。 AWS規範指引適用於可變動執行個體的修補程式管理。

表 10-操作標籤可以是環境特定

開發	安裝	生產
{ "Tags": [{ "Key": "Maintenance Window", "ResourceId": "i-012345678ab9ab111",	{ "Tags": [{ "Key": "Maintenance Window", "ResourceId": "i-012345678ab9ab444",	{ "Tags": [{ "Key": "Maintenance Window", "ResourceId": "i-012345678ab9ab777",

開發	安裝	生產
"ResourceType": "instance", "Value": "cron(30 23 ? * TUE#1 *)" }, { "Key": "Name", "ResourceId": "i-012345678ab9ab222", "ResourceType": "instance", "Value": "WEBAPP" }, { "Key": "Patch Group", "ResourceId": "i-012345678ab9ab333", "ResourceType": "instance", "Value": "WEBAPP-DEV-AL2" }] }	"ResourceType": "instance", "Value": "cron(30 23 ? * TUE#2 *)" }, { "Key": "Name", "ResourceId": "i-012345678ab9ab555", "ResourceType": "instance", "Value": "WEBAPP" }, { "Key": "Patch Group", "ResourceId": "i-012345678ab9ab666", "ResourceType": "instance", "Value": "WEBAPP-TEST-AL2" }] }	"ResourceType": "instance", "Value": "cron(30 23 ? * TUE#3 *)" }, { "Key": "Name", "ResourceId": "i-012345678ab9ab888", "ResourceType": "instance", "Value": "WEBAPP" }, { "Key": "Patch Group", "ResourceId": "i-012345678ab9ab999", "ResourceType": "instance", "Value": "WEBAPP-PROD-AL2" }] }

零時差漏洞也可以通過定義標籤來補充您的修補策略來管理。如需詳細指引，請參閱使用 AWS Systems Manager 進行同日安全性修補來避免零時差弱點。

操作可觀察性

您必須具備觀察力，才能取得環境效能的可行洞察，並協助您偵測和調查問題。它還具有次要目的，允許您定義和衡量關鍵績效指標（KPI）和服務水平目標（SLO），例如正常運行時間。對於大多數組織而言，重要的作業 KPI 是偵測平均時間 (MTTD)，以及從事件復原的平均時間 (MTTR)。

在觀察性中，上下文很重要，因為收集數據，然後收集關聯的標籤。無論您關注的服務、應用程式或應用程式層為何，都可以篩選和分析該特定資料集。標籤可用於自動化「CloudWatch 警示」的上線，以便在違反特定測量結果臨界值時收到警示適當的團隊。例如，標籤鍵`example-inc:ops:alarm-`

tag及其上的值可能表示已建立「 CloudWatch 警示」。在使用標籤為 Amazon EC2 執行個體建立和維護 Amazon CloudWatch 警示中描述了一個解決方案。

配置過多的警報可以輕鬆地產生警報風暴-當大量警報或通知迅速壓倒操作員並降低他們的整體效率，同時操作員手動分類和優先級單獨的警報。警報的其他內容可以以標籤的形式提供，這意味著可以在 Amazon 中定義規則， EventBridge 以幫助確保將焦點放在上游問題上，而不是下游依賴關係。

同時營運的角色往往 DevOps 被忽略，但對於許多組織而言，中央營運團隊仍然會在正常工作時間以外提供關鍵的第一回應。(有關此模型的更多詳細資訊，請參閱卓越營運白皮書。) 與擁有工作負載的 DevOps 團隊不同，它們通常沒有相同的知識深度，因此標籤在儀表板和警示中提供的內容，可以將它們導向到正確的 Runbook 以解決問題，或啟動自動執行手冊（請參閱使用自動化 Amazon CloudWatch Alerps 的部落格文章）。AWS Systems Manager

用於資料安全性、風險管理和存取控制的標籤

Organizations 在適當處理資料儲存和處理方面有不同的需求和義務。資料分類是數個使用案例的重要前兆，例如存取控制、資料保留、資料分析和合規性。

資料安全與風險管理

在AWS環境中，您可能會擁有不同合規性和安全性需求的帳戶。例如，您可能有一個開發人員沙箱，以及一個託管生產環境的帳戶，用於高度管制的工作負載，例如處理付款。透過將它們隔離到不同的帳戶中，您可以套用不同的安全性控制、限制對敏感資料的存取，並減少受管制工作負載的稽核範圍。

對所有工作負載採用單一標準可能會導致挑戰。雖然許多控制項同樣適用於整個環境，但對於不需要符合特定法規架構的帳戶，以及不存在個人可識別資料的帳戶 (例如，開發人員沙箱或工作負載開發帳戶)，某些控制項過多或無關。這通常會導致誤判的安全發現項目，這些發現必須進行分類和關閉，而不採取任何動作，這會消除應該調查的發現項目的努力。

表 11 — 範例資料安全性和風險管理標籤

使用案例	標籤鍵	理由	範例數值
事件管理	`example-inc:incident-management:escalationlog`	支持團隊正在使用的系統記錄事件	`jira`, `servicenow` , `zendesk`

使用案例	標籤鍵	理由	範例數值
事件管理	example-inc:incident-management:escalationpath	升級的路徑	ops-center ，dev-ops, app-team
資料分類	example-inc:data:classification	對資料進行分類以達到合規性	Public, Private, Confidential ，Restricted
合規	example-inc:compliance:framework	識別工作負載受限制的合規性架構	PCI-DSS, HIPAA

跨AWS環境手動管理不同的控制項既耗時又容易出錯。下一個步驟是自動化適當的安全控制的部署，並根據該帳戶的分類配置資源檢查。透過將標記套用至帳戶及其中的資源，即可針對工作負載自動化並適當地設定控制項的部署。

範例：

工作負載包括具有該值的標籤example-inc:data:classification的 Amazon S3 儲存貯體Private。安全工具自動化會部署AWS Config規則s3-bucket-public-read-prohibited，以檢查 Amazon S3 儲存貯體的區塊公用存取設定、儲存貯體政策和儲存貯體存取控制清單 (ACL)，確認儲存貯體的組態適合其資料分類。為確保儲存貯體的內容與分類一致，Amazon Macie 可設定為檢查個人可識別資訊 (PII)。使用 Amazon Macie 驗證 S3 儲存貯體資料分類的部落格深入探索此模式。

某些法規環境（例如保險和醫療保健）可能會受到強制性資料保留政策的約束。使用標籤的資料保留結合 Amazon S3 生命週期政策，是一種有效且簡單的方法，將物件轉換範圍設為不同的儲存層。Amazon S3 生命週期規則也可用於在強制保留期到期後使物件過期，以便刪除資料。如需此程序的深入指南，請參閱透過將物件標籤與 Amazon S3 生命週期搭配使用來簡化資料生命週期。

此外，在分類或解決安全性發現時，標籤可以為調查人員提供重要的上下文，以幫助限定風險，並有助於吸引適當的團隊來調查或減輕發現。

身分識別管理和存取控制的標籤

使用跨AWS環境管理存取控制時AWS IAM Identity Center，標籤可以啟用多種擴展模式。有幾種委派模式可以應用，有些是基於標記。我們將單獨解決這些問題，並提供鏈接以進一步閱讀每個問題。

適用於個人資源的 ABAC

IAM 身分中心使用者和 IAM 角色支援以屬性為基礎的存取控制 (ABAC)，可讓您根據標籤來定義操作和資源的存取權限。ABAC 有助於減少更新權限原則的需求，並協助您從公司目錄中存取員工屬性。如果您已經在使用多帳戶策略，除了角色型存取控制 (RBAC) 之外，還可以使用 ABAC，為在同一帳戶中操作的多個團隊提供對不同資源的精細存取權。例如，IAM 身分中心使用者或 IAM 角色可以包含限制對特定 Amazon EC2 執行個體的存取限制條件，否則必須在每個政策中明確列出這些執行個體才能存取這些執行個體。

由於 ABAC 授權模型依賴於對操作和資源的訪問的標籤，因此提供護欄以防止意外訪問非常重要。SCP 只允許在特定條件下修改標籤，藉此保護整個組織的標籤。部落格中的使用服務控制政策來保護用於授權的資源標籤AWS Organizations和 IAM 實體的許可邊界提供如何實作此功能的資訊。

如果使用長壽命的 Amazon EC2 執行個體支援更傳統的操作實務，則可以使用此方法，部落格為 Amazon EC2 執行個體設定 IAM 身分中心 ABAC 和系統管理員工作階段管理器會話管理員會詳細討論這種形式的屬性型存取控制。如前所述，並非所有資源類型都支援標記，而且並非所有資源類型都支援使用標籤政策強制執行，因此最好在AWS 帳戶.

如需了解支援 ABAC 的服務，請參閱與 IAM 搭配使用的AWS服務。

結論

AWS您可以針對各種用途加上標記資源，從實施成本分配策略到支援自動化或授權AWS資源存取權。由於涉及的利益相關者群組數量以及資料來源和標籤治理等考量，因此對於某些組織來說，實施標記策略可能具有挑戰性。

在本白皮書中，我們概述了有關根據營運實踐、定義的使用案例、參與流程的利益相關者，以及提供的工具和服務在組織中設計和實施標記策略的AWS建議。當涉及到標記策略時，這是一個迭代和改進的過程，您可以從立即的優先級開始，識別整個組織中的相關使用案例，然後根據需要實施和擴展標記結構描述，同時不斷測量和提高效率。我們已經指出，組織內一組明確定義的標籤可讓您將AWS使用量和消耗與負責其所存在資源和業務目的的團隊建立關聯，以便與組織策略和價值保持一致。

貢獻者

本文件的貢獻者包括：

- 克里斯·派斯，高級專家技術客戶經理，Amazon Web Services
- 維傑·謝卡饒，企業 Support 負責人，Amazon Web Services
- 納塔莉婭·戈杜諾克, 高級專家技術客戶經理, Amazon Web Services
- 拜縣日出, 高級解決方案建築師, 亞馬遜互聯網服務私人有限公司
- 傑米·伊布斯，高級專家技術客戶經理，Amazon Web Services

深入閱讀

如需詳細資訊，請參閱

- AWSRe：發明 2020：倒退工作：亞馬遜的創新方法
- AWS規範指引：使用 AWS Systems Manager 在混合雲中自動修補可變執行個體
- AWS建築中心

AWSWell-Architected

- AWSWell-Architected 的框架
- 卓越營運支柱-AWS Well-Architected 的架構
- 災難復原規劃 (DR)-AWS Well-Architected 的可靠性支柱
- 成本優化支柱-AWS Well-Architected 的框架
- AWSWell-Architected 的實驗室：啟用AWS產生的成本分配 Tags
- AWSWell-Architected 的實驗室：標籤政策
- AWSWell-Architected 的實驗室：AWSCUR 查詢庫

AWS部落格

- AWS Health感知 — 自訂組織和個人AWS帳戶的AWS Health警示
- 如何自動標記 Amazon EC2 資源以回應 API 事件
- AWS產生與使用者定義的成本分攤
- 成本標記和報告 AWS Organizations
- 使用修補程AWS Systems Manager式管理員修補 Windows EC2 執行個體
- 使用即日安全性修補程式，避免零時差漏洞 AWS Systems Manager

AWS 文件

- 使用成本配置標籤-AWS Billing and Cost Management 以及成本管理與成本管理
- 什麼是AWS成本與用量報告
- AWS Resource Groups API 參考
- 如何使用 IAM 政策標籤來限制 EC2 執行個體或 EBS 磁碟區的建立方式？

- 可變與不可變的更新模型

Other (其他)

- 布賴爾, C. 和卡爾, B. (2021). 倒退工作：來自亞馬遜內部的見解，故事和秘密。倫敦麥克米倫。
- AWS CloudFormation警衛 (GitHub)

文件修訂

如要接收此白皮書更新的通知，請訂閱 RSS 摘要。

變更	描述	日期
次要更新	身分識別管理的更新	2023 年 3 月 30 日
次要修訂	更新 ABAC 中個別資源的參考資料。	2023 年 2 月 24 日
次要修訂	更新了指南以符合 IAM 最佳實務。如需詳細資訊，請參閱 IAM 中的安全最佳實務。	2023 年 2 月 6 日
主要修訂	已針對AWS Config規則支援的資源類型新增更具體的參照`required_tags` 。	2023 年 1 月 18 日
主要修訂	已更新，包括最新的做法和服務功能，特別是在身份領域。	2022 年 9 月 29 日
次要更新	修正 PDF 版本中的表格格式。	2022 年 4 月 25 日
主要修訂	更新了文件結構和擴充的標籤策略和使用案例區段。根據最新的工具、技術和可用資源，新增更多規範性指引。	2022 年 4 月 22 日
初始出版	白皮書首次出版。	2018 年 12 月 1 日

> **Note**
>
> 若要訂閱 RSS 更新，您必須為所使用的瀏覽器啟用 RSS 外掛程式。

注意

客戶有責任對本文件中的資訊進行自行獨立評估。本文件：(a) 僅供參考，(b) 代表目前的AWS產品供應項目和做法，如有變更，恕不另行通知，且 (c) 不會向其關聯公司、供應商或授權人建立任何承諾或保證。AWS AWS產品或服務係依「原狀」提供，不含任何明示或暗示之擔保、陳述或條件。客戶的責任和責任由AWS協議控制，本文件不屬於與客戶之間AWS的任何協議的一部分，也不會修改。AWS

AWS 詞彙表

如需最新的 AWS 術語，請參閱《AWS 詞彙表 參考》中的 AWS 詞彙表。

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。