

安全支柱



安全支柱: AWS Well-Architected Framework

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

摘要和介紹	1
簡介	1
安全基礎	2
設計原則	2
定義	2
共同的責任	3
控管	4
AWS 帳戶管理和區隔	5
SEC01-BP01 使用帳戶區隔工作負載	6
SEC01-BP02 保護帳戶根使用者和屬性	9
安全操作工作負載	13
SEC01-BP03 識別和驗證控制目標	14
SEC01-BP04 隨時掌握安全威脅和建議的最新資訊	16
SEC01-BP05 縮小安全管理範圍	17
SEC01-BP06 自動部署標準安全控制	19
SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施	21
SEC01-BP08 定期評估和實作新的安全服務和功能	25
身分與存取管理	27
身管理	27
SEC02-BP01 使用強式登入機制	28
SEC02-BP02 使用臨時憑證	30
SEC02-BP03 安全地儲存和使用機密	34
SEC02-BP04 仰賴集中式身分提供者	38
SEC02-BP05 定期稽核和輪換憑證	42
SEC02-BP06 採用使用者群組和屬性	44
許可管理	46
SEC03-BP01 定義存取需求	48
SEC03-BP02 授予最低權限存取權	51
SEC03-BP03 建立緊急存取程序	54
SEC03-BP04 持續減少許可	59
SEC03-BP05 為您的組織定義許可防護機制	61
SEC03-BP06 根據生命週期管理存取	64
SEC03-BP07 分析公有和跨帳戶存取權	66
SEC03-BP08 在組織內安全地共用資源	68

SEC03-BP09 安全地與第三方共用資源	71
偵測	75
SEC04-BP01 設定服務和應用程式日誌記錄	75
實作指引	7
資源	8
SEC04-BP02 在標準化的位置擷取日誌、調查結果和指標	79
實作指引	7
實作步驟	15
資源	8
SEC04-BP03 建立安全提醒的相互關聯並增添其豐富性	82
實作指引	7
資源	8
SEC04-BP04 針對不合規資源實施補救措施	85
實作指引	7
資源	8
基礎設施保護	88
保護網路	89
SEC05-BP01 建立網路層	89
SEC05-BP02 控制網路層內的流量流程	91
SEC05-BP03 實作以檢查為基礎的保護	94
SEC05-BP04 自動化網路保護	96
保護運算	98
SEC06-BP01 執行漏洞管理	99
SEC06-BP02 從強化的影像佈建運算	101
SEC06-BP03 減少手動管理和互動式存取	103
SEC06-BP04 驗證軟體完整性	105
SEC06-BP05 自動化運算保護	107
資料保護	110
資料分類	110
SEC07-BP01 了解您的資料分類機制	110
SEC07-BP02 根據資料敏感性實施資料保護控制	112
SEC07-BP03 自動識別和分類	114
SEC07-BP04 定義可擴展的資料生命週期管理	117
保護靜態資料	119
SEC08-BP01 實作安全金鑰管理	119
SEC08-BP02 強制靜態加密	122

SEC08-BP03 自動化靜態資料保護	125
SEC08-BP04 強制存取控制	128
保護傳輸中的資料	130
SEC09-BP01 實作安全金鑰和憑證管理	131
SEC09-BP02 強制傳輸中加密	134
SEC09-BP03 驗證網路通訊	136
事件回應	140
AWS 事件回應	140
雲端回應的設計目標	141
準備	142
SEC10-BP01 識別關鍵人員和外部資源	142
SEC10-BP02 制定事件管理計畫	145
SEC10-BP03 準備鑑識功能	148
SEC10-BP04 開發和測試安全事件回應程序手冊	150
SEC10-BP05 預先佈建存取權	152
SEC10-BP06 部署前工具	155
SEC10-BP07 執行模擬	157
作業	159
事後處理	159
SEC10-BP08 建立從事件中學習的架構	160
應用程式安全	162
SEC11-BP01 應用程式安全訓練	163
實作指引	7
資源	8
SEC11-BP02 自動化在整個開發和發佈生命週期的測試	165
.....	166
.....	166
實作指引	7
資源	8
SEC11-BP03 定期進行滲透測試	168
實作指引	7
資源	8
SEC11-BP04 執程式碼檢閱	170
實作指引	7
資源	8
SEC11-BP05 集中化套件和相依性的服務	173

實作指引	7
資源	8
SEC11-BP06 以程式設計方式部署軟體	175
實作指引	7
資源	8
SEC11-BP07 定期評估管道的安全屬性	178
實作指引	7
資源	8
SEC11-BP08 打造在工作負載團隊中納入安全所有權的計畫	180
實作指引	7
資源	8
結論	182
貢獻者	183
深入閱讀	185
文件修訂	186
注意	189
AWS 詞彙表	190

安全支柱 – AWS Well Architected Framework

發布日期：2024 年 11 月 6 日 ([文件修訂](#))

本白皮書的重點是 [AWS Well-Architected Framework](#) 的安全支柱。文中提供的指引可協助您將最佳實務和目前的建議應用在安全 AWS 工作負載的設計、交付和維護中。

簡介

[AWS Well-Architected Framework](#) 可協助您了解在 AWS 上建置工作負載時所做決策的權衡取捨。透過使用該架構，您將了解關於在雲端設計和操作可靠、安全、有效率、經濟實惠且永續的工作負載的目前架構最佳實務。該架構可讓您根據最佳實務一致地量測工作負載，並找出需要改進的方面。我們相信，擁有 Well-Architected 工作負載可大幅提高企業成功的可能性。

此架構以六大支柱為基礎：

- 操作效能
- 安全
- 可靠性
- 效能效率
- 成本最佳化
- 永續性

本白皮書著重於安全支柱。本文可協助您遵循目前 AWS 建議，以符合業務和法規要求。其適用於擔任技術職務的人員，例如技術長 (CTO)、資安長 (CSO/CISO)、架構師、開發人員和營運團隊成員。

閱讀本白皮書之後，您將了解在考慮安全的情況下，設計雲端架構時應使用的 AWS 目前的建議和策略。本文並未提供實作細節或架構模式，但包含有關這些資訊的適當資源參考。透過採用本文中的實務，您可以建置保護資料和系統、控制存取並自動回應安全事件的架構。

安全基礎

安全支柱描述如何利用雲端技術來保護資料、系統和資產，從而改善您的安全狀態。本白皮書針對在 AWS 上建構安全的工作負載，提供深入的最佳實務指引。

設計原則

雲端中有一些原則能協助您強化工作負載的安全：

- **實作堅實的身分識別基礎：**實作最低權限原則，並對於每個與 AWS 資源的互動強制執行職責與適當的授權分離。集中進行身分管理，旨在消除對長期靜態憑證的倚賴。
- **維持可追蹤性：**即時監控、提醒和稽核您環境中發生的動作和變更。將日誌和指標收集與系統進行整合，以自動調查並採取動作。
- **在所有層級套用安全：**透過多個安全控制，套用深度防禦方法。套用至所有層級 (例如，網路邊緣、VPC、負載平衡、每個執行個體和運算服務、作業系統、應用程式和程式碼)。
- **將安全最佳實務自動化：**將基於軟體的安全機制自動化，以提高您安全、快速和以具成本效益的方式擴展的能力。建立安全架構 (包括實作控制) 在版本控制的範本中作為程式碼定義和管理。
- **保護傳輸中和靜態資料：**將您的資料分為不同的敏感性等級，並使用適當的機制，例如加密、記號化及存取控制。
- **讓人員遠離資料：**使用機制和工具，來降低或消除對直接存取或手動處理資料的需要。在處理敏感資料時，這降低了處理不當或修改以及人為錯誤的風險。
- **為安全事件作準備：**為事故做好萬全準備，建立與您組織的要求吻合的事件管理和調查政策與程序。執行失敗回應模擬和使用工具與自動化，以提高偵測、調查和復原的速度。

定義

雲端中的安全包括七個層面：

- [安全基礎](#)
- [身分與存取管理](#)
- [偵測](#)
- [基礎設施保護](#)
- [資料保護](#)
- [事件回應](#)

- [應用程式安全](#)

共同的責任

安全與合規是 AWS 和客戶之間共同的責任。這種共同模式有助減輕客戶的運作負擔，因為 AWS 會深入服務運作所在設施的實體安全性，操作、管理並控制主機作業系統及虛擬化層的元件。客戶應承擔相關責任並負責管理訪客作業系統 (包括更新與安全性修補程式)，以及其他相關應用程式軟體，還有設定 AWS 提供的安全群組防火牆。客戶應審慎思考所選的服務，因為使用的服務、服務與客戶 IT 環境的整合情形，以及適用的法律與法規不同，客戶應承擔的責任也會不同。此共同責任的性質也提供允許部署的彈性和客戶控制。如下圖所示，這種責任區分通常被稱為「雲端安全性」與「雲端中的安全性」。

AWS 責任「雲端安全性」 – AWS 負責保護執行 AWS 雲端中提供的所有服務的基礎設施。此基礎設施由執行 AWS 雲端服務的硬體、軟體、聯網與設施組成。

客戶責任「雲端中的安全性」 – 客戶責任將由客戶選取的 AWS 雲端服務確定。這也確定了客戶在履行其安全性責任的過程中，必須執行的設定工作量。例如，Amazon Elastic Compute Cloud (Amazon EC2) 等服務會分類為基礎架構即服務 (IaaS)，因此會要求客戶執行所有必要的安全設定和管理任務。部署 Amazon EC2 執行個體的客戶負責管理訪客作業系統 (包括更新和安全性修補程式)、客戶在執行個體上安裝的任何應用程式軟體或公用程式，以及設定每個執行個體上由 AWS 提供的防火牆 (稱為安全群組)。對於 Amazon S3 和 Amazon DynamoDB 等抽象服務，AWS 會操作基礎設施層、作業系統和平台，而且客戶會存取端點以儲存和擷取資料。客戶負責管理其資料 (包括加密選項)、分類資產，以及使用 IAM 工具來套用適當的許可。

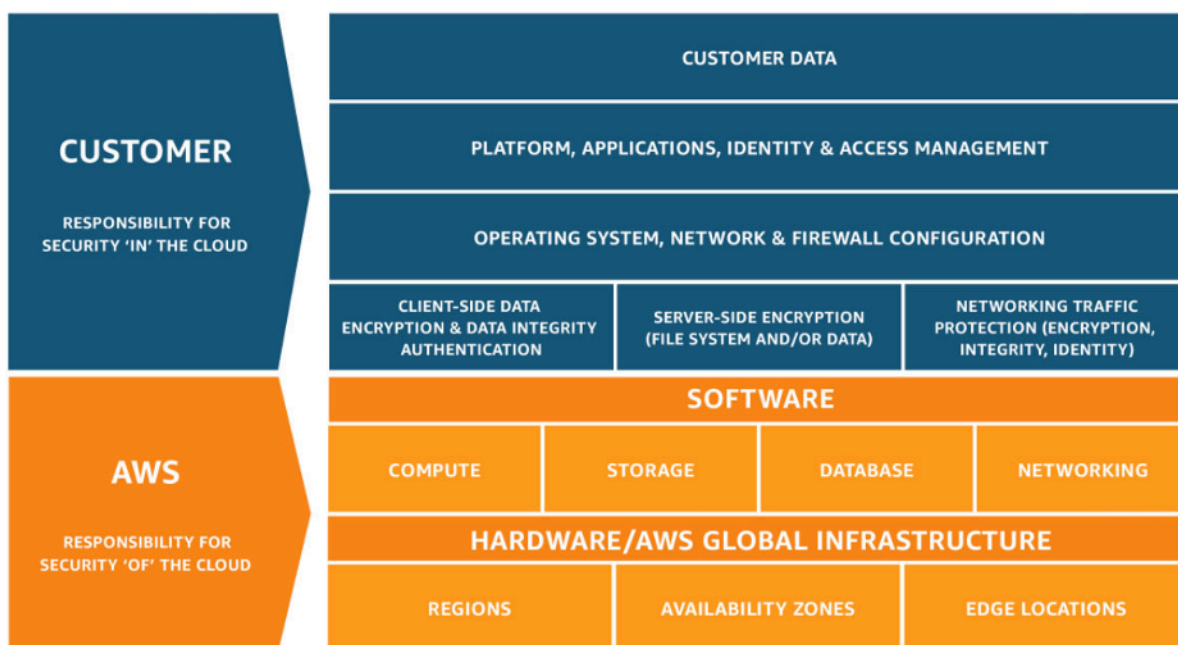


圖 1：AWS 共同責任模式。

這種客戶/AWS 共同責任模式也擴大到 IT 控制。一如 AWS 和客戶共同承擔 IT 環境的操作責任，IT 控制的管理、操作和驗證，也是由雙方共同承擔責任。AWS 可以藉由管理與 AWS 環境中所部署的實體基礎設施相關聯的那些控制，來協助減輕客戶操作控制的負擔，而上述控制先前已由客戶管理。由於每個客戶在 AWS 中的部署方式均不同，因此客戶可將特定 IT 控制的管理工作轉移給 AWS，以產生 (新的) 分散式控制環境。客戶可以使用其可取得之 AWS 控制與合規文件，視需要執行控制評估與驗證程序。以下範例控制是由 AWS、AWS 客戶或兩者共同管理。

繼承的控制 – 客戶完全繼承自 AWS 的控制。

- 實體與環境控制

共用的控制 – 同時適用於基礎設施層和客戶層的控制，但在不同的內容或觀點中。在共用的控制中，AWS 提供基礎設施的需求，而且客戶必須在他們使用的 AWS 服務內提供自己的控制實作。範例包括：

- 修補程式管理 – AWS 負責修補和修正基礎設施中的瑕疵，但客戶負責修補其客體作業系統和應用程式。
- 組態管理 – AWS 維護其基礎設施裝置的組態，但客戶負責設定其自己的客體作業系統、資料庫和應用程式。
- 認知與培訓 – AWS 負責培訓 AWS 員工，但客戶必須培訓其自己的員工。

客戶特有 – 根據客戶在 AWS 服務內部署的應用程式而定，客戶須全權負責的控制。範例包括：

- 服務和通訊保護或區域安全，客戶可能須在特定安全環境內路由或區分資料。

控管

安全管控作為整體方法的子集，旨在藉由定義政策和控制目標來協助管理風險，以支援業務目標。藉由遵循分層方法安全地控制目標來實現風險管理 - 每一層都建立在前一層之上。了解 AWS 共同責任模式是您的基礎層。這點清楚地說明了您在客戶端的責任，以及您從 AWS 繼承的責任。有益的資源是 [AWS Artifact](#)，可讓您隨需存取 AWS 安全和合規報告，以及選取線上協議。

滿足下一層的大部分控制目標。這就是全平台的能力所在。例如，這一層包括 AWS 帳戶銷售流程、與身分供應商 (例如 AWS IAM Identity Center 單一登入) 的整合，以及常見的偵測控制。這裡也會列出一些平台管控流程輸出。當您想要開始使用新的 AWS 服務時，請更新 AWS Organizations 服務中的服

務控制政策 (SCP)，為服務的初始使用提供防護機制。您可以使用其他 SCP 來實現常見的安全控制目標，通常稱為安全不變量。這些是您套用至多個帳戶、組織單位或整個 AWS 組織的控制目標或組態。典型範例是限制基礎設施執行所在的區域或防止停用偵測控制。此中間層還包含編碼政策，例如管道中的組態規則或檢查。

最上層是產品團隊符合控制目標的地方。因為這個實作是由產品團隊所控制應用程式進行完成。過程中可能是在應用程式中實作輸入驗證，或確保身分在微型服務之間正確傳遞。即使產品團隊擁有組態，他們仍可能繼承中間層的一些功能。

無論您在何處實作控制，目標都是管理風險。一系列適用於特定產業、區域或技術的風險管理架構。您的主要目標：根據可能性和後果來找出風險。這是固有風險。您可以接著定義一個控制目標，以降低可能性或後果，或同時降低兩者。然後，配合適當的控制，您就能預見可能產生的風險。這是剩餘風險。控制目標適用於一個或多個工作負載。下圖顯示典型的風險矩陣。可能性以先前發生的頻率為依據，而後果以事件的財務、信譽和時間成本為依據。

Likelihood	Risk Level				
Very Likely	Low	Medium	High	Critical	Critical
Likely	Low	Medium	Medium	High	Critical
Possible	Low	Low	Medium	Medium	High
Unlikely	Low	Low	Medium	Medium	High
Very unlikely	Low	Low	Low	Medium	High
Consequence	Minimal	Low	Medium	High	Severe

圖 2：風險等級可能性矩陣

AWS 帳戶管理和區隔

我們建議您根據功能、合規需求或一組常用的控制項，將個別帳戶和群組帳戶中的工作負載分門別類，而不要複製組織的報告結構。在 AWS 中，帳戶是硬性界限。例如，強烈建議進行帳戶層級的區隔，以便將生產工作負載與開發和測試工作負載隔離。

集中管理帳戶： AWS Organizations 可[自動建立和管理 AWS 帳戶](#)，並在建立帳戶之後控制這些帳戶。當您透過 AWS Organizations 建立帳戶時，請務必考量所使用的電子郵件地址，因為這會允許重設密碼的根使用者。Organizations 可讓您將帳戶分組為[組織單位 \(OU\)](#)，這些單位可根據工作負載的需求和用途代表不同的環境。

集中設定控制： 僅允許適當層級的特定服務、區域和服務動作，以控制您的 AWS 帳戶可以執行的操作。AWS Organizations 可讓您使用服務控制政策 (SCP) 在組織、組織單位或帳戶層級套用許可防護機制，這適用於所有[AWS Identity and Access Management \(IAM\)](#) 使用者和角色。例如，您可以套用 SCP，限制使用者在您未明確允許的區域中啟動資源。AWS Control Tower 提供一種簡化的方式來設定和管控多個帳戶。它會自動設定 AWS Organization 中的帳戶、自動化佈建、套用[防護機制](#) (包括預防和偵測)，以及為您提供可見性的儀表板。

集中設定服務和資源： AWS Organizations 可協助您設定適用於所有帳戶的[AWS 服務](#)。例如，您可以使用[AWS CloudTrail](#)，為組織內執行的所有動作設定集中日誌記錄，並阻止成員帳戶停用日誌記錄。此外，您還可以集中彙總使用[AWS Config](#) 定義的規則資料，讓您能夠稽核工作負載的合規性，並快速對變更做出反應。AWS CloudFormation[StackSets](#) 可讓您跨帳戶和組織單位集中管理組織中的 AWS CloudFormation 堆疊。這讓您能夠自動佈建新帳戶以符合您的安全需求。

使用安全服務的委託管理功能將用於管理的帳戶與組織計費 (管理) 帳戶分開。數個 AWS 服務 (例如 GuardDuty、Security Hub 和 AWS Config) 支援與 AWS Organizations 的整合，包括為管理功能指定特定帳戶。

最佳實務

- [SEC01-BP01 使用帳戶區隔工作負載](#)
- [SEC01-BP02 保護帳戶根使用者和屬性](#)

SEC01-BP01 使用帳戶區隔工作負載

透過多帳戶策略在環境 (例如生產、開發和測試) 與工作負載之間建立共通的防護機制和隔離。強烈建議帳戶層級的區隔，因為這在安全性、帳單和存取方面提供了有力的隔離界限。

預期成果： 一種帳戶結構，可將雲端作業、不相關的工作負載和環境隔離為單獨的帳戶，從而提高整個雲端基礎設施的安全性。

常見的反模式：

- 將多個具有不同資料敏感度等級且不相關的工作負載置於相同的帳戶中。
- 定義不良的組織單位 (OU) 結構。

建立此最佳實務的優勢：

- 若工作負載遭到意外存取，縮小影響範圍。
- 集中管控對 AWS 服務、資源和區域的存取。
- 利用政策以及集中管理安全服務，維護雲端基礎設施的安全性。
- 自動化帳戶建立和維護流程。
- 集中稽核您的基礎設施以滿足合規性和法規需求。

未建立此最佳實務時的風險暴露等級：高

實作指引

AWS 帳戶 在以不同的敏感度等級操作的工作負載或資源之間提供安全隔離界限。AWS 提供工具透過多帳戶策略大規模管理您的雲端工作負載，以利用此隔離界限。如需有關 AWS 中的多帳戶策略的概念、模式和實作指引，請參閱 [Organizing Your AWS Environment Using Multiple Accounts](#)。

當您集中管理多個 AWS 帳戶 時，應該將您的帳戶組織成由組織單位 (OU) 層定義的階層。接著可以組織安全控制並套用至 OU 和成員帳戶，在組織內的成員帳戶上建立一致的預防性控制。安全控制是繼承的，讓您能夠篩選位於 OU 階層較低層級的成員帳戶可用的許可。良好的設計可利用此繼承關係來降低必要的安全政策數目和複雜度，達成每個成員帳戶預期的安全控制。

[AWS Organizations](#) 和 [AWS Control Tower](#) 是可以用來在 AWS 環境中實作和管理此多帳戶結構的兩個服務。AWS Organizations 可讓您將帳戶組織到由一個或多個 OU 層定義的階層中，每個 OU 都包含許多成員帳戶。[服務控制政策](#) (SCP) 可讓組織管理員在成員帳戶上建立精細的預防性控制，並且 [AWS Config](#) 可用於在成員帳戶上建立主動性和偵測性控制。許多 AWS 服務與 [AWS Organizations](#) [整合](#)，以提供委派的管理控制，並在組織的所有成員帳戶中執行服務特定任務。

[AWS Control Tower](#) 位於 AWS Organizations 之上，為具有[登陸區域](#)的多帳戶 AWS 環境提供一鍵式最佳實務設定。該登陸區域是通往由 Control Tower 所建立之多帳戶環境的進入點。與 AWS Organizations 相比，Control Tower 具有數個[好處](#)。提供改進的帳戶管控的三個優點是：

- 整合式強制性安全控制，會自動套用至獲准加入組織的帳戶。
- 選擇性控制，可針對指定 OU 集合開啟或關閉。
- [AWS Control Tower Account Factory](#) 提供自動化帳戶部署，其中包含組織內部預先核准的基準和組態選項。

實作步驟

1. 設計組織單位結構：設計合理的組織單位結構可減少建立及維護服務控制政策及其他安全性控制所需的管理負擔。您的組織單位結構應與業務需求、資料敏感度和工作負載結構保持一致。
2. 為多帳戶環境建立登陸區域：登陸區域可提供一致的安全性和基礎設施基礎，您的組織可以從中快速開發、啟動和部署工作負載。可以使用自訂建置的登陸區域或 [AWS Control Tower](#) 來協調您的環境。
3. 建立防護機制：透過登陸區域為您的環境實作一致的安全防護機制。AWS Control Tower 提供可部署的[強制性](#)控制與[選擇性](#)控制清單。實作 Control Tower 時會自動部署強制性控制。審核強烈建議的控制和選擇性控制清單，並實作符合您需求的控制。
4. 限制對新增區域的存取：對於新 AWS 區域，IAM 資源 (例如使用者和角色) 只會傳播到您指定的區域。可以在[使用 Control Tower 時透過主控台](#)執行此動作，或在 [AWS Organizations 中調整 IAM 權限政策](#)。
5. 考慮 AWS [CloudFormation StackSets](#)：StackSets 可協助您將資源 (包括 IAM 政策、角色和群組) 從核准的範本中部署到不同的 AWS 帳戶 帳戶和區域。

資源

相關的最佳實務：

- [SEC02-BP04 仰賴集中式身分提供者](#)

相關文件：

- [AWS Control Tower](#)
- [《AWS 安全性稽核指南》](#)
- [IAM 最佳實務](#)
- [使用 CloudFormation StackSets 在多個 AWS 帳戶 和區域中佈建資源](#)
- [Organizations 常見問答集](#)
- [AWS Organizations 術語與概念](#)
- [AWS Organizations 多帳戶環境中服務控制政策的最佳實務](#)
- [AWS 帳戶管理參考指南](#)
- [使用多個帳戶組織您的 AWS 環境](#)

相關影片：

- [透過自動化和管控大規模採用 AWS](#)
- [以 Well-Architected 方式提供安全最佳實務](#)
- [使用 AWS Control Tower 建置和管控多個帳戶](#)
- [為現有組織啟用 Control Tower](#)

相關研討會：

- [Control Tower Immersion Day](#)

SEC01-BP02 保護帳戶根使用者和屬性

根使用者是 AWS 帳戶中最具特權的使用者，對帳戶內的所有資源具備完整的管理存取權，並且在某些情況下，不受安全政策的限制。停用對根使用者的程式設計存取，為根使用者建立適當的控制，以及避免例行使用根使用者，可降低意外暴露根憑證及後續危及雲端環境的風險。

預期成果：保護根使用者有助於減少因濫用根使用者憑證而造成意外或蓄意損壞的機會。建立偵測控制也能在當使用根使用者採取動作時警告適當的人員。

常見的反模式：

- 將根使用者用於需要根使用者憑證以外的工作。
- 疏於定期測試緊急應變計劃以確認重大基礎設施、程序和人員在緊急情況下的運作情形。
- 僅考慮一般帳戶登入流程而疏於考慮或測試替代帳戶復原方法。
- 未將 DNS、電子郵件伺服器 and 電話提供者作為重要安全周邊的一部分來處理，因為其會用於帳戶復原流程。

建立此最佳實務的優勢：保護對根使用者的存取，可建立對帳戶中的動作加以控制和稽核的信心。

未建立此最佳實務時的風險暴露等級：高

實作指引

AWS 提供眾多工具來協助保護您的帳戶。然而，由於預設情況下不會開啟其中一些措施，因此您必須採取直接行動加以實作。考慮將這些建議作為保護 AWS 帳戶的基本步驟。實作這些步驟時，務必建立程序以持續評估和監視安全控制。

首次建立 AWS 帳戶時，您是從一個對帳戶中所有 AWS 服務和資源具有完全存取權的身分開始。此身分稱為 AWS 帳戶根使用者。您可以使用您用來建立帳戶的電子郵件地址和密碼，以根使用者的身分

登入。由於授予給 AWS 根使用者更高的存取權限，您必須限制 AWS 根使用者的使用，才能執行[特別需要它](#)的任務。根使用者登入憑證必須受嚴密防護，並且多重要素驗證 (MFA) 應始終用於 AWS 帳戶根使用者。

除了一般驗證流程 (使用使用者名稱、密碼和多重要素驗證 (MFA) 裝置登入根使用者) 之外，還有帳戶復原流程會登入 AWS 帳戶根使用者，而其能夠存取與您的帳戶相關聯的電子郵件地址和電話號碼。因此，保護傳送復原電子郵件的根使用者電子郵件帳戶以及與帳戶相關聯的電話號碼同樣也很重要。另外，對於與根使用者相關聯的電子郵件地址託管在相同 AWS 帳戶的電子郵件伺服器或網域名稱服務 (DNS) 資源上的情況，也要考慮可能的循環相依性。

使用 AWS Organizations 時，會有多個 AWS 帳戶，各自都有根使用者。將一個帳戶指定為管理帳戶，接著可以在該管理帳戶之下新增數層成員帳戶。優先保護您的管理帳戶根使用者後，再來處理成員帳戶根使用者。保護管理帳戶根使用者的策略可不同於成員帳戶根使用者，而且您可以對成員帳戶根使用者設立預防性安全控制。

實作步驟

以下是為根使用者建立控制的建議實作步驟。如果適用，建議將交叉引用到 [CIS AWS Foundations 基準版本 1.4.0](#)。除了這些步驟之外，請參閱[保護您的 AWS 帳戶及其資源的 AWS 最佳實務指引](#)。

預防性控制

1. 為帳戶設定準確的[聯絡資訊](#)。
 - a. 此資訊會用於遺失密碼復原流程、遺失 MFA 裝置帳戶復原流程，以及與您的團隊進行重大安全相關通訊。
 - b. 使用由您的企業網域所託管的電子郵件地址 (最好是使用分發清單) 作為根使用者的電子郵件地址。使用分發清單而不是個人的電子郵件帳戶可對長期存取根帳戶提供額外的備援和持續性。
 - c. 聯絡資訊上所列的電話號碼應該是針對此用途的專用安全電話。不應公布或與他人共用電話號碼。
2. 請勿為根使用者建立存取金鑰。若存在存取金鑰，請將其移除 (CIS 1.4)。
 - a. 去除根使用者任何長期存留的程式設計憑證 (存取和秘密金鑰)。
 - b. 如果根使用者存取金鑰已存在，您應該使用這些金鑰來轉換程序，以使用來自 AWS Identity and Access Management (IAM) 角色的臨時存取金鑰，然後[刪除根使用者存取金鑰](#)。
3. 確定您是否需要儲存根使用者的憑證。
 - a. 如果您使用 AWS Organizations 建立新成員帳戶，則成員帳戶上的根使用者的初始密碼會設為隨機值，並且不會向您公開。如有需要，請考慮使用來自 AWS 組織管理帳戶的密碼重設流程，[以取得成員帳戶的存取權](#)。

- b. 對於獨立 AWS 帳戶 或管理 AWS 組織帳戶，請考慮建立根使用者的憑證並安全存放。為根使用者使用 MFA。
4. 在 AWS 多帳戶環境中為成員帳戶根使用者使用預防性控制。
 - a. 考慮針對成員帳戶使用 [不允許為根使用者建立根存取金鑰](#) 預防性防護機制。
 - b. 考慮針對成員帳戶使用 [不允許以根使用者身分執行動作](#) 預防性防護機制。
 5. 如果您需要根使用者的憑證：
 - a. 使用複雜密碼。
 - b. 為根使用者開啟多重要素驗證 (MFA)，尤其是 AWS Organizations 管理 (付款人) 帳戶 (CIS 1.5)。
 - c. 考慮硬體 MFA 裝置以獲得彈性和安全性，因為一次性裝置可減少包含 MFA 代碼的裝置重複用於其他用途的可能性。確認定期更換使用電池的硬體 MFA 裝置。(CIS 1.6)
 - 若要為根使用者設定 MFA，請遵循建立 [虛擬 MFA](#) 或 [硬體 MFA 裝置](#) 的指示。
 - d. 考慮註冊多個 MFA 裝置進行備份。 [每個帳戶最多允許 8 個 MFA 裝置](#)。
 - 請注意， [如果 MFA 裝置遺失](#)，為根使用者註冊多個 MFA 裝置會自動關閉復原帳戶的流程。
 - e. 請將密碼妥善保管，如果以電子方式儲存密碼，請考慮循環相依性。儲存密碼時，請勿以需要存取相同的 AWS 帳戶 來取得密碼的方式儲存。
 6. 選擇性：考慮為根使用者建立定期密碼輪流排程。
 - 憑證管理最佳實務取決於您法規和政策需求。受 MFA 保護的根使用者不依賴把密碼當作單一驗證要素。
 - 定期 [變更根使用者密碼](#) 可降低不慎公開密碼遭到濫用的風險。

偵測性控制

- 建立警示以偵測根憑證的使用 (CIS 1.7)。 [Amazon GuardDuty](#) 可以透過 [RootCredentialUsage](#) 調查結果來監控根使用者 API 憑證使用情況，並發出提醒。
- 評估並實作 [適用於 AWS Config 的《AWS Well-Architected 安全支柱》一致性套件](#) 中包含的偵測控制，或者如果使用 AWS Control Tower，則評估並實作 Control Tower 內可用的 [強烈建議的控制](#)。

操作指引

- 確定組織內誰應該存取根使用者憑證。
 - 使用雙人規則，如此沒有單獨一人可以存取所有必要的憑證和 MFA 來取得根使用者存取權。

- 確認組織而不是單一個人持有對與帳戶相關聯的電話號碼和電子郵件別名 (用於密碼重設和 MFA 重設程序) 的控制權。
- 只在特殊情況下使用根使用者 (CIS 1.7)。
 - AWS 根使用者不可用於日常任務，即使管理任務也一樣。只有以根使用者身分登入，才能執行需要根使用者的 AWS 任務。所有其他動作都應該由其他擔任適當角色的使用者執行。
- 定期檢查根使用者的存取權操作正常，以便在發生需要使用根使用者憑證的緊急情況之前，測試相關程序。
- 定期檢查與帳戶相關聯的電子郵件地址，以及替代連絡人下列出的電子郵件地址。監控這些電子郵件收件匣，查看您可能接收的來自 <abuse@amazon.com> 的安全通知。另外確保與帳戶相關聯的任何電話號碼都有效。
- 準備事件回應程序以回應根帳戶誤用的情況。請參閱 [AWS Security Incident Response Guide](#) 和 [安全支柱白皮書中「事件回應」一節](#)中的最佳實務，了解如何為 AWS 帳戶建立事件回應策略的詳細資訊。

資源

相關的最佳實務：

- [SEC01-BP01 使用帳戶區隔工作負載](#)
- [SEC02-BP01 使用強式登入機制](#)
- [SEC03-BP02 授予最低權限存取權](#)
- [SEC03-BP03 建立緊急存取程序](#)
- [SEC10-BP05 預先佈建存取權](#)

相關文件：

- [AWS Control Tower](#)
- [《AWS 安全性稽核指南》](#)
- [IAM 最佳實務](#)
- [Amazon GuardDuty – 根憑證使用情況警示](#)
- [透過 CloudTrail 監控根憑證使用情況的逐步指引](#)
- [已核准可與 AWS 搭配使用的 MFA 權杖](#)
- [在 AWS 上實作緊急存取](#)

- [在 AWS 帳戶 中要改善的十大安全項目](#)
- [如果發現我的 AWS 帳戶 中有未經授權的活動，該怎麼辦？](#)

相關影片：

- [透過自動化和管控大規模採用 AWS](#)
- [以 Well-Architected 方式提供安全最佳實務](#)
- [限制使用 AWS 根憑證](#)，來自 AWS re:inforce 2022 – AWS IAM 安全最佳實務

相關範例和實驗室：

- [實驗室：AWS 帳戶設定和根使用者](#)

安全操作工作負載

操作工作負載安全地涵蓋了工作負載的整個生命週期，從設計、建置、執行到持續改善。改善在雲端中安全操作能力的方法之一，就是採用組織方法進行管控。管控是一致地指引決策的方式，而不是僅依賴相關人員的良好判斷。您的管控模型和流程是您回答「我如何知道是否達到給定工作負載的控制目標，並且這些目標是否適合於該工作負載？」問題的方式。採用一致方法做出決策，可以加快工作負載的部署，並有助於提高組織中安全能力的標準。

若要安全地操作工作負載，您必須將整體的最佳實務套用到每個安全區域。採用您在組織和工作負載層級於卓越營運中定義的要求和程序，並將其應用於所有區域。掌握最新的 AWS 和產業建議以及威脅情報，可協助您發展威脅模型並控制目標。自動化安全流程、測試和驗證可協助您擴展安全操作。

自動化讓流程得以維持一致性和可重複性。人員或許擅長做很多事情，但卻不包括一直重複同一件事而能不出錯。即使執行手冊撰寫得再完整，仍難免發生人員未能始終如一地執行重複任務的風險。尤其當人員承擔不同的責任，而必須回應不熟悉的提醒時，更是如此。然而，自動化每次都會以相同的方式回應。部署應用程式的最佳方式就是透過自動化。執行部署的程式碼可以進行測試，然後用來執行部署。這可在變更過程中增加信心，並降低變更失敗的風險。

若要驗證組態是否達到您的控制目標，請先在非生產環境中測試自動化和部署的應用程式。如此一來，您可以測試自動化以證明它已正確地執行所有步驟。您也可以在開發和部署週期中獲得早期意見回饋，從而減少返工。若要減少部署錯誤的機會，請透過程式碼而不是人員來進行組態變更。如果您需要重新部署應用程式，自動化會使這個動作容易得多。當定義其他控制目標時，您可以輕鬆地將它們新增到所有工作負載的自動化。

與其讓個別工作負載擁有者投資於其工作負載特有的安全性，不如透過使用常用功能和共用元件來節省時間。多個團隊可以取用的一些服務範例包括 AWS 帳戶建立流程、集中式人員身分、常用的日誌記錄組態，以及 AMI 和容器基礎映像建立。這種方法可以協助建立者縮短工作負載週期時間，並始終如一地達到安全控制目標。當團隊更加一致時，您可以驗證控制目標，並向利害關係人更好地報告您的控制態勢和風險位置。

最佳實務

- [SEC01-BP03 識別和驗證控制目標](#)
- [SEC01-BP04 隨時掌握安全威脅和建議的最新資訊](#)
- [SEC01-BP05 縮小安全管理範圍](#)
- [SEC01-BP06 自動部署標準安全控制](#)
- [SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施](#)
- [SEC01-BP08 定期評估和實作新的安全服務和功能](#)

SEC01-BP03 識別和驗證控制目標

根據合規需求以及從威脅模型識別的風險，衍生並驗證您需要套用到工作負載的控制目標和控制。對控制目標與控制持續進行驗證，可協助您測量風險降低的有效性。

預期成果：企業的安全控制目標是明確定義的，並符合您的合規要求。控制項是透過自動化和政策來實作和強制執行的，並持續評估其在達成目標方面的有效性。在一個時間點和一段時間內的有效性證據可以很容易地向稽核人員報告。

常見的反模式：

- 您的企業對可保證安全的法規要求、市場預期和業界標準並未充分了解
- 網路安全架構和控制目標不符合業務需求
- 控制措施的實作並未以可衡量的方式與您的控制目標保持一致
- 您不使用自動化來報告控制措施的有效性

未建立此最佳實務時的曝險等級：高

實作指引

有許多常見的網路安全框架可以構成安全控制目標的基礎。考慮企業的法規要求、市場期望和業界標準，以決定哪些架構最能支援您的需求。範例包括 [AICPA SOC 2](#)、[HITRUST](#)、[PCI-DSS](#)、[ISO 27001](#) 和 [NIST SP 800-53](#)。

針對您確定的控制目標，了解您使用的 AWS 服務如何協助您達成這些目標。使用 [AWS Artifact](#) 尋找符合目標架構的文件和報告，這些文件和報告可說明 AWS 涵蓋的責任範圍，以及您負責的剩餘範圍的指引。如需符合各種架構控制聲明的詳細服務特定指引，請參閱 [AWS Customer Compliance Guides](#)。

當您定義實現目標的控制措施時，使用預防性控制措施對執行進行整理，並使用偵測性控制來自動化緩解措施。使用 [服務控制政策 \(SCP\)](#)，協助防止您的 AWS Organizations 中的不合規資源組態和動作。在 [AWS Config](#) 中實作規則以監控和報告不合規的資源，然後在對其行為有信心後，將規則切換為強制執行模型。若要部署符合網路安全架構的預定義和受控規則集，請首先評估 [AWS Security Hub 標準](#) 的使用。AWS Foundational Service Best Practices (FSBP) 標準和 CIS AWS Foundations Benchmark 是良好的起點，具有與多個標準框架共享的許多目標保持一致的控制措施。如果 Security Hub 本質上沒有所需的控制偵測，則可以使用 [AWS Config 一致性套件](#) 來補充它。

使用 AWS 全球安全與合規加速 (GSCA) 團隊建議的 [APN 合作夥伴套件](#)，在需要時從安全顧問、諮詢機構、證據收集和報告系統、稽核人員和其他輔助服務那裡取得協助。

實作步驟

1. 評估常見的網路安全架構，並使您的控制目標與選擇的目標保持一致。
2. 使用 AWS Artifact 取得有關架構指引和責任的相關文件。了解哪些法規遵循部分屬於共同責任模式的 AWS 方面，以及哪些部分是您的責任。
3. 使用 SCP、資源政策、角色信任政策及其他防護機制來防止不合規的資源組態和動作。
4. 評估部署符合您控制目標的 Security Hub 標準和 AWS Config 一致性套件。

資源

相關的最佳實務：

- [SEC03-BP01 定義存取需求](#)
- [SEC04-BP01 設定服務和應用程式日誌記錄](#)
- [SEC07-BP01 了解您的資料分類方案](#)
- [OPS01-BP03 評估管控要求](#)
- [OPS01-BP04 評估合規要求](#)
- [PERF01-BP05 使用政策和參考架構](#)
- [COST02-BP01 根據貴組織的需求制定政策](#)

相關文件：

- [AWS 客戶合規指南](#)

相關工具：

- [AWS Artifact](#)

SEC01-BP04 隨時掌握安全威脅和建議的最新資訊

透過監控業界威脅情報刊物和資料摘要以取得更新，以便隨時掌握最新的威脅和緩解措施。評估根據最新威脅資料自動更新的受管服務產品。

預期成果：隨著產業刊物更新，隨時掌握最新威脅和建議的資訊。在識別新威脅時，使用自動化技術偵測潛在的漏洞和暴露。您可以採取緩解措施對抗這些威脅。可採用 AWS 服務，以自動掌握最新威脅情報的資訊。

常見的反模式：

- 沒有可靠且可反覆執行的機制，因而無法隨時掌握新威脅情報。
- 手動維護技術產品組合、工作負載和相依項的庫存清單，而這些都需要人員審查才能得知是否有潛在的漏洞和暴露。
- 沒有既定的機制能夠將工作負載和相依項更新到可用的最新版本，因而無法取得已知的威脅緩解措施。

建立此最佳實務的優勢：使用威脅情報來源隨時掌握新資訊，即可盡量避免錯過可能影響業務的威脅態勢中發生的重大變化。採用自動化的方式取代手動，以掃描、偵測和修復工作負載及其相依項中存在的潛在漏洞或暴露，如此就能協助您事先預測並快速降低風險。這有助於控制與漏洞緩解相關的時間和成本。

未建立此最佳實務時的曝險等級：高

實作指引

審核值得信賴的威脅情報刊物，以掌握威脅態勢。請參閱 [MITRE ATT&CK](#) 知識庫中的文件，了解已知的對手策略、技巧和程序 (TTP)。審核 MITRE 的 [常見漏洞和暴露 \(CVE\)](#) 清單，隨時了解您仰賴的產品中的已知漏洞。透過開放式全球應用程式安全專案 (OWASP) 熱門的 [OWASP 十大專案](#)，了解 Web 應用程式的重大風險。

透過 CVE 的 AWS [安全公告](#)，隨時掌握 AWS 安全事件和建議的修復步驟。

為了減輕您隨時掌握新知的整體投入與負擔，請考慮使用 AWS 服務，這些服務會隨著時間自動納入新的威脅情報。例如，[Amazon GuardDuty](#) 會隨時提供業界威脅情報的最新資訊，可用來偵測您帳戶內的異常行為和威脅特徵。[Amazon Inspector](#) 會自動將其用於連續掃描功能的 CVE 資料庫保持在最新狀態。[AWS WAF](#) 和 [AWS Shield Advanced](#) 兩者都提供受管規則群組，這些群組會隨著新的威脅出現時自動更新。

審核 [Well-Architected 卓越營運支柱](#)，了解自動機群管理和修補的資訊。

實作步驟

- 訂閱與您的業務和產業相關的威脅情報刊物更新。訂閱 AWS 安全公告。
- 考慮採用會自動納入新威脅情報的服務，例如 Amazon GuardDuty 和 Amazon Inspector。
- 部署符合 Well-Architected 卓越營運支柱最佳實務的機群管理和修補策略。

資源

相關的最佳實務：

- [SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施](#)
- [OPS01-BP05 評估威脅環境](#)
- [OPS11-BP01 建立持續改進程序](#)

SEC01-BP05 縮小安全管理範圍

確定您是否可以使用 AWS 服務將某些控制的管理轉移至 AWS (受管服務)，藉此縮小安全範圍。這些服務有助於減少安全維護任務，例如基礎設施佈建、軟體設定、修補或備份。

預期成果：您為工作負載選取 AWS 服務時，會考慮安全管理的範圍。除了其他 Well-Architected 考量外，還會根據您所選取服務的成本來權衡管理開銷和維護任務的成本 (總體擁有成本，亦即 TCO)。您會將 AWS 控制和合規文件納入您的控制評估和驗證程序中。

常見的反模式：

- 部署工作負載時，並未徹底了解您所選取服務的共用責任模式。
- 在虛擬機器上託管資料庫和其他技術時，未先行評估對等的受管服務。

- 比較受管服務選項時，未將安全管理任務納入虛擬機器上託管技術的總體擁有成本中。

建立此最佳實務的優勢：使用受管服務可以減輕您管理營運安全控制的整體負擔，進而降低安全風險和總體擁有成本。否則，時間可能會花費在某些安全任務上，而無法轉投入其他為企業創造更多價值的任務。受管服務也可以透過將某些控制要求轉移到 AWS，以縮小合規要求的範圍。

未建立此最佳實務時的曝險等級：中

實作指引

您可以透過多種方式在 AWS 上整合工作負載的元件。您在 Amazon EC2 執行個體上安裝和執行技術時，通常需要承擔最大部分的整體安全責任。為了協助您減輕操作特定控制的負擔，請識別可縮小您的共同責任模式範圍的 AWS 受管服務，並了解如何在現有架構中使用這些服務。範例包括使用 [Amazon Relational Database Service \(Amazon RDS\)](#) 來部署資料庫，使用 [Amazon Elastic Kubernetes Service \(Amazon EKS\)](#) 或 [Amazon Elastic Container Service \(Amazon ECS\)](#) 來協調容器，或使用[無伺服器選項](#)。建置新的應用程式時，請仔細考量哪些服務有助於減少實作和管理安全控制方面的時間和成本。

合規要求也是選取服務時的考量因素。受管服務可以將某些合規要求轉移至 AWS。請與合規團隊討論，了解他們在稽核您操作和管理的服務方面，以及接受相關 AWS 稽核報告中控制聲明時面對的難度。您可以將 [AWS Artifact](#) 中找到的稽核成品提供給稽核員或監管機構，作為 AWS 安全控制的證據。您也可以使用某些 AWS 稽核成品所提供的責任指引來設計您的架構，以及參考 [AWS 客戶合規指南](#)。本指引可協助您確定應採取哪些額外的安全控制，以便支援系統的特定使用案例。

使用受管服務時，請熟悉將其資源更新為較新版本的程序 (例如，更新 Amazon RDS 所管理資料庫的版本，或 AWS Lambda 函數的程式設計語言執行時期)。雖然受管服務可能會自動為您執行此操作，但設定更新的時間並了解對操作的影響仍然是您的責任。[AWS Health](#) 等工具可以協助您在整個環境中追蹤和管理這些更新。

實作步驟

1. 評估可取代為受管服務的工作負載元件。
 - a. 如果您要將工作負載遷移至 AWS，在評估是否應主機轉換、重構、平台轉換、重新建置或取代您的工作負載時，考慮減少管理 (時間和費用) 並降低風險。有時候，在一開始遷移時的額外投資，長遠來看可能帶來大幅的節省。
2. 考慮實作 Amazon RDS 等受管服務，而非安裝和管理您自己的技術部署。
3. 使用 AWS Artifact 中的責任指引來協助您確定應針對工作負載採取的安全控制。
4. 將使用中的資源記錄起來，並隨時掌握新的服務和方法，以識別縮小範圍的新機會。

資源

相關的最佳實務：

- [PERF02-BP01 選取最適合您工作負載的運算選項](#)
- [PERF03-BP01 使用最能滿足資料存取和儲存需求的專用資料存放區](#)
- [SUS05-BP03 使用受管服務](#)

相關文件：

- [AWS Health 的計劃生命週期事件](#)

相關工具：

- [AWS Health](#)
- [AWS Artifact](#)
- [AWS 客戶合規指南](#)

相關影片：

- [如何使用 AWS DMS 遷移至 Amazon RDS 或 Aurora MySQL 資料庫執行個體？](#)
- [AWS re:Invent 2023 - 使用 AWS Health 大規模管理資源生命週期事件](#)

SEC01-BP06 自動部署標準安全控制

在開發和部署整個 AWS 環境的標準安全控制時，採用現代 DevOps 實務。使用基礎設施即程式碼 (IaC) 範本定義標準安全控制和組態、擷取版本控制系統中的變更、在 CI/CD 管道中測試變更，並將變更自動部署至您的 AWS 環境。

預期成果：IaC 範本會擷取標準化的安全控制，並將其遞交至版本控制系統。CI/CD 管道準備好偵測變更，並自動測試和部署您的 AWS 環境。防護機制準備好在進行部署之前，先偵測範本中的組態錯誤並發出提醒。工作負載會部署到已採取標準控制的環境中。團隊有權透過自助服務機制部署經核准的服務組態。已制定安全的備份和復原策略來控制組態、指令碼和相關資料。

常見的反模式：

- 透過 Web 主控台或命令列介面，手動變更標準安全控制。

- 仰賴個別工作負載團隊手動實作中央團隊定義的控制。
- 仰賴中央安全團隊應工作負載團隊的要求部署工作負載層級的控制。
- 允許相同的個人或團隊開發、測試和部署安全控制自動化指令碼，而未能妥善區分職責，或適當地對其加以制衡。

建立此最佳實務的優勢：使用範本定義標準安全控制，可讓您使用版本控制系統追蹤和比較一段時間的變更。使用自動化方式測試和部署變更可建立標準化和可預測性，從而提高成功部署的機會，並減少手動重複任務。為工作負載團隊提供自助服務機制來部署核准的服務和組態，可降低組態錯誤和濫用的風險。這也有助於讓團隊及早在開發過程中納入控制。

未建立此最佳實務時的風險暴露等級：中

實作指引

遵循 [SEC01-BP01 使用帳戶區隔工作負載](#) 中所述的實務，您最終會有多個 AWS 帳戶，可用於您使用 AWS Organizations 管理的不同環境。雖然其中每個環境和工作負載可能需要不同的安全控制，但您可以將整個組織的某些安全控制標準化。範例包括整合集中式身分提供者、定義網路和防火牆，以及設定用於儲存和分析日誌的標準位置。同樣地，您可以使用基礎設施即程式碼 (IaC) 將同樣嚴謹的應用程式程式碼開發程序套用至基礎設施佈建，也可以使用 IaC 來定義和部署標準安全控制。

請盡可能以宣告的方式定義安全控制 (例如在 [AWS CloudFormation](#) 中)，並將其儲存在來源控制系統中。使用 DevOps 實務來自動部署控制，以便更容易預測發行版本、使用 [AWS CloudFormation Guard](#) 等工具進行自動測試，以及偵測已部署控制與所需組態之間的偏差。您可以使用 [AWS CodePipeline](#)、[AWS CodeBuild](#) 和 [AWS CodeDeploy](#) 等服務來建構 CI/CD 管道。請考慮 [使用多個帳戶整理您的 AWS 環境](#) 中的指引，以在自己的帳戶中設定這些服務，將其與其他部署管道分開來。

您也可以定義範本來標準化定義和部署 AWS 帳戶、服務和組態。此技術可讓中央安全團隊管理這些定義，並透過自助服務方法將其提供給工作負載團隊。實現這一點的方法之一是使用 [Service Catalog](#)，您可以在其中將範本發佈為產品，讓工作負載團隊可以將這些產品納入自己的管道部署中。如果您使用的是 [AWS Control Tower](#)，某些範本和控制可以作為起點。Control Tower 還提供 [Account Factory](#) 功能，可讓工作負載團隊使用您定義的標準建立新的 AWS 帳戶。此功能有助於消除對中央團隊的依賴，以在工作負載團隊視需要識別新帳戶時核准和建立新帳戶。您可能需要這些帳戶，以便根據諸如提供的功能、所處理資料的敏感性或其行為等原因，區隔不同的工作負載元件。

實作步驟

1. 確定如何在版本控制系統中儲存和維護範本。

2. 建立 CI/CD 管道以測試和部署您的範本。定義測試以檢查是否有組態錯誤，以及範本是否符合您公司的標準。
3. 建置標準化範本目錄，供工作負載團隊根據您的需求部署 AWS 帳戶和服務。
4. 針對控制組態、指令碼和相關資料實作安全的備份和復原策略。

資源

相關的最佳實務：

- [OPS05-BP01 使用版本控制](#)
- [OPS05-BP04 使用建置和部署管理系統](#)
- [REL08-BP05 使用自動化部署變更](#)
- [SUS06-BP01 採用可以快速引入永續性改進的方法](#)

相關文件：

- [使用多個帳戶組織您的 AWS 環境](#)

相關範例：

- [使用 Service Catalog、AWS Organizations 和 AWS Lambda 自動建立帳戶和佈建資源](#)
- [使用 AWS Secrets Manager、AWS KMS 和 AWS Certificate Manager 加強 DevOps 管道並保護資料](#)

相關工具：

- [AWS CloudFormation Guard](#)
- [AWS 登陸區域加速器](#)

SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施

執行威脅建模，為您的工作負載識別並保有潛在威脅及關聯緩解措施的最新記錄。排定威脅的優先順序並調整安全控制緩解措施，以防止、偵測和回應威脅。就您的工作負載的情況，以及不斷演變的安全態勢，重新檢視和維護此工作。

未建立此最佳實務時的曝險等級：高

實作指引

什麼是威脅建模？

「威脅建模以保護有價值物為目標，識別、溝通和了解威脅及緩解措施。」 – [開放式 Web 應用程式安全專案 \(OWASP\) 應用卡程式威脅建模](#)

為何使用威脅模型？

系統本身錯綜複雜，並且隨時間變得更複雜且更具能力，從而實現更大的商業價值及更高的客戶滿意度和參與度。這意味著 IT 設計決策需要考慮不斷增加的使用案例數量。這種複雜性和使用案例數量的排列通常使得非結構化方法無法有效尋找和緩解威脅。反之，您需要一套系統化方法來列舉對系統的潛在威脅，以及制定緩解措施，並以這些緩解措施為優先來確保組織的有限資源能在改善系統整體安全狀態上發揮最大的影響力。

威脅建模旨在提供這套系統化方法，目的是要在設計過程中及早尋找和解決問題，此時進行緩解的成本和精力與生命週期稍後相比要來得低。此方法與[往前移安全性的](#)業界原則相一致。威脅建模最終會與組織的風險管理程序整合，透過使用威脅驅動的方法，協助推動要實作哪些控制決策。

何時應執行威脅建模？

在工作負載的生命週期中及早開始威脅建模，可給予您更大的彈性來決定要如何處理所識別的威脅。就跟軟體錯誤一樣，越早識別威脅，就能以越具成本效益的方式加以解決。威脅模型是不斷更新的文件，並且應該持續隨著工作負載的變更而演進。隨時間重新檢視您的威脅模型，包括當有重大變更、威脅形勢有變化，或是採用新功能或服務時。

實作步驟

我們如何執行威脅建模？

有許多不同的方式來執行威脅建模。就跟程式設計語言一樣，各有優缺點，而您應選擇最適合您的方式。其中一個方法是從 [Shostack 針對威脅建模的 4 個問題框架](#) 開始著手，當中提出自由回答的問題會為您的威脅建模練習提供結構：

1. 我們目前在做什麼？

此問題的目的是協助您了解正在建置的系統並對之取得一致的意見，以及該系統與安全相關的詳細資訊。建立模型或圖表是回答此問題最受歡迎的方法，因為這可協助您將正在建置的項目視覺化，例如使用[資料流程圖](#)。寫下關於您的系統的假設和重要詳細資訊也有助於您定義涵蓋的範圍。這可讓對參與威脅模型的每個人專注於相同的事情，並避免耗時地繞入不在範圍內的主題 (包括系統的過

時版本)。舉例來說，如果您正在建置 Web 應用程式，可能不值得花時間為瀏覽器用戶端建立作業系統信任開機順序的模型，因為您無法透過您的設計對此產生影響。

2. 什麼可能出錯？

這是您識別對系統的威脅之處。威脅是意外或故意的動作或事件，會帶來不必要的影響，並且可能會影響系統安全。如果對可能出錯之處沒有清楚的了解，則無法對症下藥。

對於什麼可能出錯，您並沒有標準的清單可循。建立此清單需要團隊內的每個人與[涉及的相關角色](#)在威脅建模練習中集思廣益和共同協作。您可以使用識別威脅的模型來協助集思廣益，例如[STRIDE](#)，這會建議不同的類別以進行評估：詐騙、竄改、否認、資訊洩露、拒絕服務和提升權限。此外，您可能會想要審核現有的清單並研究以獲得靈感來協助集思廣益，包括[OWASP 前十大](#)、[HiTrust 威脅型錄](#)，以及您組織本身的威脅型錄。

3. 我們要做何處理？

就跟上一個問題一樣，對於所有可能的緩解措施並沒有標準的清單可循。此步驟的輸入是上一步識別的威脅、動作和改進之處。

安全與合規是[您和 AWS 之間共同責任](#)。了解當您提出「我們要做何處理？」時，也是在問「誰要對其負責？」，這一點很重要。了解您與 AWS 之間的責任制衡有助於您將威脅建模練習的範圍定在您控制之下的緩解措施，這通常是 AWS 服務組態選項與您自身的系統特定緩解措施的組合。

對於共同責任的 AWS 部分，您將發現[AWS 服務在許多合規計畫的範圍之內](#)。這些計畫可協助您了解 AWS 在維護雲端安全和合規方面設立的強大控制。來自這些計畫的稽核報告可供 AWS 客戶從[AWS Artifact](#) 下載。

無論您使用何種 AWS 服務，其始終涉及客戶責任，而您的威脅模型中應包含與這些責任一致的緩解措施。對於 AWS 服務本身的安全控制緩解措施，您要考慮跨領域實作安全控制，包括身分和存取管理 (驗證和授權)、資料保護 (靜態和傳輸中)、基礎設施安全、日誌記錄和監控等領域。每個 AWS 服務的文件都有[專屬的安全章節](#)，提供將安全控制視為緩解措施的指引。重要的是，考慮您正在編寫的程式碼及其程式碼相依性，並思考您可以設立以解決這些威脅的控制。這些控制可以是[輸入驗證](#)、[工作階段處理](#)和[界限處理](#)等事項。大多數漏洞通常是在自訂程式碼中引入，因此請專注於此區域。

4. 我們處理得當嗎？

目標是讓您的團隊和組織改進威脅模型的品質以及隨時間執行威脅建模的速度。這些改進出自練習、學習、教導和審查的組合。若要更加深入並實作，建議您與您的團隊完成[建置人員建立威脅模型的正確方式訓練課程](#)或[研討會](#)。此外，如果您正在尋找有關如何將威脅建模整合至您組織的應用程式開發生命週期，請參閱 AWS 安全部落格上的[如何進行威脅建模](#)文章。

威脅編寫器

為了協助並指導您執行威脅建模，請考慮使用[威脅編寫器](#)工具，該工具旨在縮短威脅建模實現價值的時間。此工具可協助您執行下列操作：

- 撰寫與[威脅文法](#)相符、可在自然非線性工作流程中使用的有用威脅陳述式
- 產生人類可讀的威脅模型
- 產生機器可讀的威脅模型，以讓您將威脅模型視為程式碼
- 使用洞察儀表板協助您快速識別品質和涵蓋範圍有所改進的領域

如需進一步的參考，請造訪「[威脅編寫器](#)」，並切換到系統定義的範例工作區。

資源

相關的最佳實務：

- [SEC01-BP03 識別和驗證控制目標](#)
- [SEC01-BP04 隨時掌握安全威脅和建議的最新資訊](#)
- [SEC01-BP05 縮小安全管理範圍](#)
- [SEC01-BP08 定期評估和實作新的安全服務和功能](#)

相關文件：

- [如何進行威脅建模 \(AWS 安全部落格\)](#)
- [NIST：以資料為中心的系統威脅建模指南](#)

相關影片：

- [AWS Summit ANZ 2021 - 如何進行威脅建模](#)
- [AWS Summit ANZ 2022 - 擴展安全性 – 針對快速和安全交付進行最佳化](#)

相關訓練：

- [建置人員建立威脅模型的正確方式 – AWS Skill Builder 虛擬自定進度培訓](#)
- [建置人員建立威脅模型的正確方式 – AWS 研討會](#)

相關工具：

- [威脅編寫器](#)

SEC01-BP08 定期評估和實作新的安全服務和功能

評估和實作 AWS 和 AWS 合作夥伴提供的安全服務和功能，讓您發展工作負載的安全狀態。

預期成果：您擁有一套標準實務，可通知您 AWS 和 AWS 合作夥伴發行的新功能和服務。您會評估這些新功能對於環境和工作負載目前和新控制的設計有何影響。

常見的反模式：

- 您未訂閱 AWS 部落格和 RSS 摘要，因此未能迅速得知相關的新功能和服務
- 您仰賴第二手來源得知安全服務和功能的最新消息和更新
- 您沒有鼓勵組織中的 AWS 使用者隨時掌握最新更新

建立此最佳實務的優勢：如果您能隨時掌握新的安全服務和功能，就可以在雲端環境和工作負載中實作控制方面做出明智的決策。這些來源有助於提高對於不斷變化的安全態勢的意識，以及說明如何使用 AWS 服務來防範新出現的威脅。

未建立此最佳實務時的曝險等級：低

實作指引

AWS 會透過幾種通道通知客戶新的安全服務和功能：

- [AWS 最新消息](#)
- [AWS 新聞部落格](#)
- [AWS 安全部落格](#)
- [AWS 安全公告](#)
- [AWS 文件概觀](#)

您可以使用 Amazon Simple Notification Service (Amazon SNS) 訂閱 [AWS 每日功能更新](#) 主題，以獲得完整的每日更新摘要。某些安全服務 (例如 [Amazon GuardDuty](#) 和 [AWS Security Hub](#)) 會提供自己的 SNS 主題，以持續掌握與這些特定服務有關的新標準、調查結果及其他更新。

每年全球各地也會舉行多場[會議、活動和網路研討會](#)，於會中宣佈並詳細描述新服務和功能。其中特別值得關注的是年度 [AWS re:Inforce](#) 安全會議，以及較為常態的 [AWS re:Invent](#) 會議。先前所述的 AWS 新聞頻道會分享這些有關安全和其他服務的會議公告，您可以在 YouTube 上的 [AWS 活動頻道](#) 線上觀看更深入的教育性分組會議。

您也可以向您的 [AWS 帳戶 團隊](#) 詢問有關最新安全服務更新和建議的資訊。如果沒有直接聯絡資訊，您可以透過 [銷售人員支援表單](#) 聯繫您的團隊。同樣地，如果您訂閱 [AWS 企業支援](#)，將會收到來自技術客戶經理 (TAM) 的每週更新，您可以與他們安排定期審查會議。

實作步驟

1. 使用您偏好的 RSS 閱讀器訂閱各種不同的部落格和公告，以及訂閱每日功能更新 SNS 主題。
2. 評估要參加哪些 AWS 活動，以獲得有關新功能和服務的第一手資訊。
3. 與您的 AWS 帳戶 團隊排定會議，以討論有關更新安全服務和功能的任何問題。
4. 考慮訂閱企業支援，即可定期向技術客戶經理 (TAM) 諮詢。

資源

相關的最佳實務：

- [PERF01-BP01 了解並熟悉可用的雲端服務和功能](#)
- [COST01-BP07 及時了解新的服務版本](#)

身分與存取管理

若要使用 AWS 服務，您必須授權您的使用者和應用程式存取您 AWS 帳戶中的資源。當您在 AWS 上執行更多工作負載時，需要穩固的身分管理和許可，以確保相應人員在適當條件下存取正確的資源。AWS 提供多種功能選項，協助您管理人類和機器身分及其許可。這些功能的最佳實務分為兩個主要領域。

主題

- [身分管理](#)
- [許可管理](#)

身分管理

處理安全的 AWS 工作負載時，您需要管理兩種身分類型。

- **真人身分：**需要存取您的 AWS 環境和應用程式的真人身分可分為三個群組：人力資源、第三方和使用者。

人力資源群組包括管理員、開發人員，以及身為您組織成員的操作人員。他們需要存取權來管理、建置和操作您的 AWS 資源。

第三方是外部協作者，例如承包商、廠商或合作夥伴。他們與您組織互動的過程中，會需要與 AWS 資源互動。

使用者是您應用程式的取用者。他們透過 Web 瀏覽器、用戶端應用程式、行動應用程式或互動式命令列工具存取您的 AWS 資源。

- **機器身分：**您的工作負載應用程式、操作工具和元件需要身分才能向 AWS 服務發出請求，例如讀取資料。這些身分也包括您 AWS 環境 (如 Amazon EC2 執行個體或 AWS Lambda 函數) 內執行的機器。您也可以管理需要存取您 AWS 環境之外部對象的機器身分，或 AWS 外部的機器。

最佳實務

- [SEC02-BP01 使用強式登入機制](#)
- [SEC02-BP02 使用臨時憑證](#)
- [SEC02-BP03 安全地儲存和使用機密](#)
- [SEC02-BP04 仰賴集中式身分提供者](#)

- [SEC02-BP05 定期稽核和輪換憑證](#)
- [SEC02-BP06 採用使用者群組和屬性](#)

SEC02-BP01 使用強式登入機制

登入 (使用登入憑證進行驗證) 可預防當沒有使用多重要素驗證 (MFA) 等機制時的風險，尤其是在登入憑證遭到意外暴露或輕易被猜出的情況下。使用強式登入機制透過要求 MFA 和強式密碼政策來降低這些風險。

預期成果：透過對 [AWS Identity and Access Management \(IAM\)](#) 使用者、[AWS 帳戶 根使用者](#)、[AWS IAM Identity Center](#) 和協力廠商身分提供者使用強式登入機制，來降低 AWS 中意外存取憑證的風險。這意味著要求使用 MFA，強制強式密碼政策，以及偵測異常的登入行為。

常見的反模式：

- 沒有為您的身分強制強式密碼政策，包括複雜密碼和 MFA。
- 在不同使用者之間共用相同的憑證。
- 沒有針對可疑的登入使用偵測控制。

未建立此最佳實務時的曝險等級：高

實作指引

真人身分可採用數種方法來登入 AWS。AWS 最佳實務是依賴使用聯合 (在 AWS IAM 與集中式 IdP 之間的直接 SAML 2.0 聯合或使用 AWS IAM Identity Center) 的集中式身分提供者，向 AWS 進行身分驗證。在這種情況下，以您的身分提供者或 Microsoft Active Directory 建立安全的登入程序。

當您第一次開啟 AWS 帳戶時，是從 AWS 帳戶 根使用者開始。您應僅使用帳戶根使用者來設定使用者的存取權 (以及 [需要根使用者的任務](#))。務必在開啟 AWS 帳戶之後，立即為帳戶根使用者開啟多重要素驗證 (MFA)，並使用 [AWS 最佳實務指南](#) 來保護根使用者。

AWS IAM Identity Center 是專為員工使用者所設計，您可以在服務內建立和管理使用者身分，並使用 MFA 保護登入程序。AWS 另一方面，Cognito 是專為客戶身分和存取管理 (CIAM) 所設計，它為應用程式中的外部使用者身分提供使用者集區和身分提供者。

如果您在 AWS IAM Identity Center 中建立使用者，請保護該服務中的登入程序，並 [開啟 MFA](#)。對於您應用程式中的外部使用者身分，您可以使用 [Amazon Cognito 使用者集區](#) 並保護該服務中的登入程序，或透過 Amazon Cognito 使用者集區支援的其中一個身分提供者。

此外，對於 AWS IAM Identity Center 中的使用者，您可以在將存取權授予 AWS 資源之前，使用 [AWS Verified Access](#) 來確認使用者的身分和裝置狀態，藉此提供一層額外的保護。

如果您使用 [AWS Identity and Access Management \(IAM\)](#) 使用者，請使用 IAM 保護登入程序。

您可以同時使用 AWS IAM Identity Center 和直接 IAM 聯合來管理 AWS 的存取權。您可以使用 IAM 聯合來管理 AWS Management Console 和服務的存取權，以及使用 IAM Identity Center 來管理 Amazon QuickSight 或 Amazon Q Business 等業務應用程式的存取權。

無論登入方法為何，強制執行強式登入政策至關重要。

實作步驟

以下是一般的強式登入建議。您設定的實際設定應由貴公司政策來規定，或使用如 [NIST 800-63](#) 的標準。

- Require MFA (需要 MFA)。 [IAM 最佳實務是對真人身分和工作負載要求 MFA](#)。開啟 MFA 可提供多一層安全防護，要求使用者提供登入憑證和一次性密碼 (OTP)，或是從硬體裝置以密碼編譯方式驗證和產生的字串。
- 強制密碼長度下限，此為密碼強度的要素。
- 強制密碼複雜性，使密碼更難猜測。
- 允許使用者變更自己的密碼。
- 建立個別身分，而不是共用憑證。透過建立個別身分，您可以為每個使用者提供一組獨一無二的安全憑證。個別使用者可讓您稽核每個使用者的活動。

IAM Identity Center 建議：

- 當使用預設目錄來建立密碼長度、複雜性和重複使用需求時，IAM Identity Center 提供預先定義的 [密碼政策](#)。
- 當身分來源為預設目錄、AWS Managed Microsoft AD 或 AD Connector 時，[開啟 MFA](#) 並為 MFA 設定內容感知或永遠開啟設定。
- 允許使用者 [註冊自己的 MFA 裝置](#)。

Amazon Cognito 使用者集區目錄建議：

- 設定 [密碼強度](#) 設定。
- 使用者 [需要 MFA](#)。

- 針對[適應性身分驗證](#) (這可封鎖可疑登入) 等功能使用 Amazon Cognito 使用者集區[進階安全設定](#)。

IAM 使用者建議：

- 在理想情況下，您使用 IAM Identity Center 或直接聯合。然而，您可能需要 IAM 使用者。在這種情況下，為 IAM 使用者[設定密碼政策](#)。您可以使用密碼政策來定義需求，例如最短長度或是密碼是否需要非字母字元等。
- 建立 IAM 政策以[強制 MFA 登入](#)，以便允許使用者管理自己的密碼和 MFA 裝置。

資源

相關的最佳實務：

- [SEC02-BP03 安全地儲存和使用機密](#)
- [SEC02-BP04 仰賴集中式身分提供者](#)
- [SEC03-BP08 在組織內安全地共用資源](#)

相關文件：

- [AWS IAM Identity Center 密碼政策](#)
- [IAM 使用者密碼政策](#)
- [設定 AWS 帳戶 根使用者密碼](#)
- [Amazon Cognito 密碼政策](#)
- [AWS 憑證](#)
- [IAM 安全最佳實務](#)

相關影片：

- [使用 AWS IAM Identity Center 大規模管理使用者許可](#)
- [在每一層都能掌握身分](#)

SEC02-BP02 使用臨時憑證

當進行任何類型的驗證時，最好是使用臨時憑證，而不是長期憑證，以降低或消除風險，例如憑證遭到意外洩露、共用或遭竊。

預期成果：為了降低長期憑證的風險，對於人員和機器身分，請盡可能使用臨時憑證。長期憑證會產生許多風險，例如，經由上傳到公有儲存庫而暴露。透過使用臨時憑證，您可大幅降低憑證遭到入侵的可能性。

常見的反模式：

- 開發人員使用取自 IAM 使用者的長期存取金鑰，而不是使用聯合從 CLI 取得臨時憑證。
- 開發人員將長期存取金鑰內嵌在程式碼中，並將該程式碼上傳到公有 Git 儲存庫。
- 開發人員將長期存取金鑰內嵌在行動應用程式中，之後在應用程式商店中提供該行動應用程式。
- 使用者與其他使用者共用長期存取金鑰，或是擁有長期存取金鑰的離職員工仍持有金鑰。
- 對機器身分可以使用臨時憑證時，卻使用長期存取金鑰。

未建立此最佳實務時的曝險等級：高

實作指引

對所有 AWS API 和 CLI 請求使用暫時安全憑證，而不是長期憑證。幾乎在任何情況下，對 AWS 服務的 API 和 CLI 請求都必須使用 [AWS 存取金鑰](#) 簽署。您可以使用暫時或長期憑證簽署這些請求。您唯一應使用長期憑證 (又稱為長期存取金鑰) 的時候是在使用 [IAM 使用者](#) 或 [AWS 帳戶 根使用者](#) 時。當您聯合至 AWS 或透過其他方法擔任 [IAM 角色](#) 時，系統會產生臨時憑證。每當您使用登入憑證存取 AWS Management Console 時，系統會為您產生臨時憑證以呼叫 AWS 服務。在幾種情況下，您將需要長期憑證，並能夠使用臨時憑證完成幾乎所有任務。

避免使用長期憑證而改用臨時憑證，同時實行減少使用 IAM 使用者並支持聯合和 IAM 角色的策略。雖然對人類和機器身分過去以來都是使用 IAM 使用者，我們現在建議不要使用它們以避免使用長期存取金鑰的風險。

實作步驟

真人身分

對於人力資源身分，例如員工、管理員、開發人員、操作員和客戶：

- 您應 [仰賴集中式身分提供者](#) 並 [要求人類使用者以聯合搭配身分提供者](#)，使用臨時憑證存取 AWS。您可以 [直接聯合至各個 AWS 帳戶](#) 或使用 [AWS IAM Identity Center](#) 和您選擇的身分提供者，為您的使用者進行聯合。與使用 IAM 使用者相比，聯合除了可消除長期憑證外，還提供一些優勢。您的使用者也可以從命令列進行 [直接聯合](#)，或使用 [IAM Identity Center](#) 請求臨時憑證。這表示有少數使用案例會需要 IAM 使用者，或使用者需要長期憑證。

對於第三方身分：

- 當授權讓第三方 (例如軟體即服務 (SaaS) 提供者) 存取 AWS 帳戶 中的資源時，您可以使用[跨帳戶角色](#)和[以資源為基礎的政策](#)。此外，您可以針對 B2B SaaS 客戶或合作夥伴，使用 [Amazon Cognito OAuth 2.0 授權](#) 用戶端憑證流程。

透過 Web 瀏覽器、用戶端應用程式、行動應用程式或互動式命令列工具存取您的 AWS 資源的使用者身分。

- 如果需要授權應用程式供消費者或客戶存取您的 AWS 資源，您可以使用 [Amazon Cognito 身分集區](#) 或 [Amazon Cognito 使用者集區](#) 來提供臨時憑證。憑證的許可是透過 IAM 角色來設定。您也可以為未驗證的訪客使用者，另外定義有限許可的 IAM 角色。

機器身分

對於機器身分，您可能需要使用長期憑證。在這些情況下，您應[要求工作負載使用臨時憑證，並以 IAM 角色存取 AWS](#)。

- 對於 [Amazon Elastic Compute Cloud \(Amazon EC2\)](#)，您可以使用[適用於 Amazon EC2 的角色](#)。
- [AWS Lambda](#) 可讓您設定 [Lambda 執行角色](#)，[授予服務許可](#) 可以使用臨時憑證執行 AWS 動作。有許多其他類似的模型可供 AWS 服務使用 IAM 角色授予臨時憑證。
- 對於 IoT 裝置，您可以使用 [AWS IoT Core 憑證提供者](#) 來請求臨時憑證。
- 對於內部部署系統或是在 AWS 之外執行並需要存取 AWS 資源的系統，您可以使用 [IAM Roles Anywhere](#)。

在某些情況下不支援臨時憑證，因此需要使用長期憑證。在這些情況下，[定期稽核和輪換這些憑證並定期輪換存取金鑰](#)。對於有嚴格限制的 IAM 使用者存取金鑰，請考慮下列其他安全措施：

- 授予有嚴格限制的許可：
 - 遵守最低權限原則 (具體指定動作、資源和條件)。
 - 考慮僅授予 IAM 使用者一個特定角色的 AssumeRole 操作。根據內部部署架構而定，此方法有助於隔離和保護長期 IAM 憑證。
- 在 IAM 角色信任政策中，限制允許的網路來源和 IP 位址。
- 監控用量並設定未使用許可或濫用的警示 (使用 AWS CloudWatch Logs 指標篩選器和警報)。
- 強制執行[許可界限](#) (服務控制政策 (SCP) 和許可界限來彼此互補 - SCP 較為粗略，而許可界限較為精細)。

- 實作程序來佈建並安全地儲存 (在內部部署保存庫中) 憑證。

其他適用於需要長期憑證的案例的選項包括：

- 建置自己的權杖供應 API (使用 Amazon API Gateway)。
- 在您必須使用長期憑證，或是使用 AWS 存取金鑰以外的憑證 (例如資料庫登入) 的情況下，您可以使用專為管理機密而設計的服務，例如 [AWS Secrets Manager](#)。Secrets Manager 可簡化加密、轉換及安全儲存加密機密的工作。許多 AWS 服務都可使用 Secrets Manager 支援[直接整合](#)。
- 對於多重雲端整合，您可以根據來源憑證服務提供者 (CSP) 憑證使用聯合身分 (請參閱 [AWS STS AssumeRoleWithWebIdentity](#))。

如需有關轉換長期憑證的詳細資訊，請參閱[轉換存取金鑰](#)。

資源

相關的最佳實務：

- [SEC02-BP03 安全地儲存和使用機密](#)
- [SEC02-BP04 仰賴集中式身分提供者](#)
- [SEC03-BP08 在組織內安全地共用資源](#)

相關文件：

- [暫時安全憑證](#)
- [AWS 憑證](#)
- [IAM 安全最佳實務](#)
- [IAM 角色](#)
- [IAM Identity Center](#)
- [身分提供者與聯合](#)
- [轉換存取金鑰](#)
- [安全合作夥伴解決方案：存取與存取控制](#)
- [AWS 帳戶根使用者](#)
- [使用 Google Cloud Platform 原生工作負載身分存取 AWS](#)
- [如何使用 AWS Security Token Service 存取 Microsoft Entra ID 租用戶中的 AWS 資源](#)

相關影片：

- [使用 AWS IAM Identity Center 大規模管理使用者許可](#)
- [在每一層都能掌握身分](#)

SEC02-BP03 安全地儲存和使用機密

工作負載需要能夠自動向資料庫、資源和第三方資源證明其身分。這需使用私密存取憑證來完成，例如 API 存取金鑰、密碼和 OAuth 字符。使用專用服務來儲存、管理和輪換這些憑證有助於降低這些憑證遭到入侵的可能性。

預期成果：實作安全管理應用程式憑證的機制，以達成下列目標：

- 識別工作負載需要何種機密。
- 盡可能以短期憑證取代長期憑證，來減少所需的長期憑證數目。
- 建立安全的存放區並自動輪換其餘的長期憑證。
- 稽核對存在於工作負載中的機密的存取。
- 持續監控以確認原始程式碼在開發過程中沒有內嵌機密。
- 降低憑證遭意外洩露的可能性。

常見的反模式：

- 沒有輪換憑證。
- 將長期憑證存放在原始程式碼或設定檔中。
- 未加密儲存靜態憑證。

建立此最佳實務的優勢：

- 已加密儲存靜態和傳輸中的機密。
- 透過 API 限制憑證的存取 (將其視為憑證自動販賣機)。
- 稽核並記錄對憑證的存取 (包括讀寫)。
- 區隔顧慮：由不同的元件執行憑證輪換，而該元件可與其餘的架構分離。
- 自動將機密隨需散發到軟體元件並集中進行輪換。
- 可以精細的方式控制對憑證的存取。

未建立此最佳實務時的風險暴露等級：高

實作指引

以往，憑證用於向資料庫進行驗證，而第三方 API、字符和其他機密可能內嵌在原始程式碼或環境檔案中。AWS 提供數種機制以安全儲存這些憑證，自動輪換並稽核它們的使用情況。

著手機密管理的最佳方法是遵循移除、取代和輪換的指引。最安全的憑證是您不用儲存、管理或處理的憑證。有些憑證對於工作負載的運作不再是必要的，故能夠安全移除。

對於工作負載適當運作仍舊是必要的憑證，可能有機會以暫時或短期憑證取代長期憑證。例如，與其對 AWS 私密存取金鑰進行硬式編碼，考慮使用 IAM 角色以臨時憑證取代長期憑證。

部分長期存留的機密可能無法移除或取代。您可以將這些機密儲存在 [AWS Secrets Manager](#) 之類的服務中，進行集中儲存、管理和定期輪換。

對工作負載的原始程式碼和設定檔的稽核，可能顯現多種類型的憑證。下表概述處理常見憑證類型的策略：

憑證類型	描述	建議策略
IAM 存取金鑰	用於在工作負載內擔任 IAM 角色的 AWS IAM 存取和私有金鑰	取代：改用指派給運算執行個體的 IAM 角色 (例如 Amazon EC2 或 AWS Lambda)。為了與需要存取您的 AWS 帳戶中的資源的第三方進行互通，請詢問他們是否支援 AWS 跨帳戶存取權 。對於行動應用程式，請考慮透過 Amazon Cognito 身分集區 (聯合身分) 使用臨時憑證。對於 AWS 外部執行的工作負載，請考慮 IAM Roles Anywhere 或 AWS Systems Manager 混合啟用 。關於容器，請參閱 Amazon ECS 任務 IAM 角色 或 Amazon EKS 節點 IAM 角色 。

憑證類型	描述	建議策略
SSH 金鑰	用於手動或做為自動化程序的一部分登入 Linux EC2 執行個體的 Secure Shell 私有金鑰	取代：使用 AWS Systems Manager 或 EC2 Instance Connect 透過 IAM 角色提供對 EC2 執行個體的程式設計和手動存取。
應用程式和資料庫憑證	密碼 – 純文字字串	輪換：將憑證儲存在 AWS Secrets Manager 中並建立自動輪換 (如果可能)。
Amazon RDS 和 Aurora Admin 資料庫憑證	密碼 – 純文字字串	取代：使用 Secrets Manager 與 Amazon RDS 的整合 或 Amazon Aurora 。此外，某些 RDS 資料庫類型可以在某些使用案例中使用 IAM 角色而非密碼 (如需詳細資訊，請參閱 IAM 資料庫身分驗證)。
OAuth 字符	私密字符 – 純文字字串	輪換：將字符儲存在 AWS Secrets Manager 中並設定自動輪換。
API 字符和金鑰	私密字符 – 純文字字串	輪換：儲存 AWS Secrets Manager 中，並在可能的情況下建立自動輪換。

常見的反模式是將 IAM 存取金鑰內嵌在原始程式碼、組態檔案或行動應用程式內。當需要 IAM 存取金鑰與 AWS 服務通訊時，請使用 [暫時 \(短期\) 安全憑證](#)。您可以透過 [適用於 EC2 的 IAM 角色](#) 執行個體、[執行角色](#) (用於 Lambda 函數)、[Cognito IAM 角色](#) (用於行動使用者存取)，以及 [IoT Core 政策](#) (用於 IoT 裝置) 提供這些短期憑證。與第三方互動時，偏好 [委派 IAM 角色的存取權](#) (包含對帳戶資源的必要存取權)，而不是設定 IAM 使用者並將該使用者的私密存取金鑰傳送給第三方。

在很多情況下，工作負載需要儲存機密才能與其他服務和資源相互操作。[AWS Secrets Manager](#) 是專為安全管理這些憑證所打造的，可儲存、使用和輪換 API 字符、密碼和其他憑證。

AWS Secrets Manager 提供五項重要功能以確保敏感憑證的安全儲存和處理：[靜態加密](#)、[傳輸中加密](#)、[全面性稽核](#)、[精細存取控制](#)，以及[可擴充的憑證輪換](#)。來自 AWS 合作夥伴的其他機密管理服務，或本機開發並提供類似功能和保證的解決方案也可接受。

擷取機密時，您可以使用 Secrets Manager 用戶端快取元件進行快取以供未來使用。擷取快取的秘密比從 Secrets Manager 中擷取要快。此外，由於呼叫 Secrets Manager API 會產生費用，因此使用快取可以降低成本。如需您可以擷取機密的所有方法，請參閱[取得機密](#)。

Note

某些語言可能會要求您針對用戶端快取實作自己的記憶體內加密。

實作步驟

1. 使用 [Amazon CodeGuru](#) 等自動工具識別包含硬式編碼憑證的程式碼路徑。
 - a. 使用 Amazon CodeGuru 掃描您的程式碼儲存庫。檢閱完成後，在 CodeGuru 中篩選 Type=Secrets 以尋找有問題的程式碼行。
2. 識別可移除或取代的憑證。
 - a. 識別不再需要的憑證並標示以進行移除。
 - b. 對於內嵌在原始程式碼中的 AWS 機密金鑰，請將其取代為與必要資源關聯的 IAM 角色。如果您部分的工作負載位於 AWS 之外，但需要 IAM 憑證來存取 AWS 資源，請考慮 [IAM Roles Anywhere](#) 或 [AWS Systems Manager 混合啟用](#)。
3. 對於其他第三方長期存留且需要使用輪換策略的機密，將 Secrets Manager 整合至程式碼中以便在執行時期擷取第三方機密。
 - a. CodeGuru 主控台可以使用已探索的憑證自動 [在 Secrets Manager 中建立機密](#)。
 - b. 將 Secrets Manager 的機密擷取整合至您的應用程式程式碼中。
 - i. 無伺服器 Lambda 函數可以使用與語言無關的 [Lambda 延伸](#)。
 - ii. 對於 EC2 執行個體或容器，AWS 提供範例 [用戶端程式碼](#)，可以數種熱門的程式設計語言從 [Secrets Manager 擷取機密](#)。
4. 定期審查您的程式碼庫並重新掃描，以確認程式碼中未加入新的機密。
 - a. 考慮使用 [git-secrets](#) 之類的工具以防將新機密認可到您的原始程式碼儲存庫。
5. [監控 Secrets Manager 活動](#) 以尋找非預期使用、不當私密存取或嘗試刪除機密的跡象。
6. 減少對憑證的人員接觸。將讀寫和修改憑證的存取權限於專門用於此用途的 IAM 角色，並且只將擔任該角色的存取權提供給一小組可操作的使用者子集。

資源

相關的最佳實務：

- [SEC02-BP02 使用臨時憑證](#)
- [SEC02-BP05 定期稽核和輪換憑證](#)

相關文件：

- [開始使用 AWS Secrets Manager](#)
- [身分提供者與聯合](#)
- [Amazon CodeGuru 推出機密偵測器](#)
- [AWS Secrets Manager 使用 AWS Key Management Service 的方式](#)
- [Secrets Manager 中的機密加密和解密](#)
- [Secrets Manager 部落格文章](#)
- [Amazon RDS 宣佈與 AWS Secrets Manager 整合](#)

相關影片：

- [大規模管理、擷取和輪換機密的最佳實務](#)
- [使用 Amazon CodeGuru 機密偵測器尋找硬式編碼的機密](#)
- [使用 AWS Secrets Manager 保護混合式工作負載的機密](#)

相關研討會：

- [在 AWS Secrets Manager 中儲存、擷取和管理敏感憑證](#)
- [AWS Systems Manager 混合啟用](#)

SEC02-BP04 仰賴集中式身分提供者

人力身分 (員工和承包商) 可仰賴身分供應商來集中管理身分。由於您是從單一位置建立、指派、管理、撤銷和稽核存取權，因此這樣一來可以更好管理多個應用程式和系統中的存取權。

預期成果：擁有集中式身分提供者，可集中管理員工使用者、身分驗證政策 (例如，要求多重要素驗證 (MFA))，以及對系統和應用程式進行授權 (例如，根據使用者的群組成員資格或屬性指派存取權)。您

的員工使用者登入集中身分提供者並聯合 (單一登入) 至內部和外部應用程式，如此一來，使用者就不需記住多個憑證。您的身分提供者與您的人力資源 (HR) 系統整合，因此人事變更會自動同步至您的身分提供者。例如，若有人離開您的組織，您可以自動撤銷聯合應用程式和系統 (包括 AWS) 的存取權。您已在身分提供者中啟用詳細稽核日誌記錄，並監控這些日誌以找出不尋常的使用者行為。

常見的反模式：

- 您未使用聯合和單一登入。您的員工使用者在多個應用程式和系統中建立了不同的使用者帳戶和憑證。
- 您尚未將員工使用者的身分生命週期自動化，例如透過整合身分提供者與您的 HR 系統。使用者離開您的組織或變更職務時，您採取手動程序在多個應用程式和系統中刪除或更新記錄。

建立此最佳實務的優勢：透過使用集中式身分提供者，您就可以從單一位置管理員工使用者身分和政策，而且能夠將應用程式存取權指派給使用者和群組，並監控使用者登入活動。透過與您的人力資源 (HR) 系統整合，使用者變更職務時，這些變更就會同步至身分提供者，並自動更新指派的應用程式和許可。使用者離開您的組織時，系統會自動停用他們在身分提供者中的身分，並撤銷他們對聯合應用程式和系統的存取權。

未建立此最佳實務時的風險暴露等級：高

實作指引

員工使用者存取 AWS 的指引：員工使用者 (例如組織中的員工和承包商) 可能需要使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 存取 AWS 來執行其工作職能。您可以透過從集中式身分提供者，在兩個層級聯合至 AWS 的方式，將 AWS 存取權授予員工使用者：直接聯合至各個 AWS 帳戶，或聯合至您的 [AWS 組織](#) 中的多個帳戶。

若要將您的員工使用者直接與各個 AWS 帳戶 聯合，您可以使用集中式身分提供者來聯合至該帳戶中的 [AWS Identity and Access Management](#)。IAM 的彈性可讓您啟用單獨的 [SAML 2.0](#) 或 [Open ID Connect \(OIDC\)](#) 身分提供者用於各個 AWS 帳戶，並使用聯合身分使用者屬性進行存取控制。您的員工使用者將透過提供憑證 (例如密碼和 MFA 字符代碼) 的方式，使用自己的 Web 瀏覽器登入身分提供者。身分提供者會向其瀏覽器發出 SAML 聲明，並提交至 AWS Management Console 登入 URL，以允許使用者透過 [擔任 IAM 角色對 AWS Management Console](#) 進行單一登入。您的使用者也可以取得暫時 AWS API 憑證，以便在 [AWS CLI](#) 或 [AWS SDK](#) (從 [AWS STS](#)) 中使用，方法是 [使用來自身分提供者的 SAML 聲明擔任 IAM 角色](#)。

若要將您的員工使用者與 AWS 組織中的多個帳戶聯合，您可以使用 [AWS IAM Identity Center](#) 集中管理員工使用者對 AWS 帳戶 和應用程式的存取權。您可以為組織啟用 Identity Center，並設定您的身分來源。IAM Identity Center 提供了預設身分來源目錄，您可以使用此目錄來管理您的使用者和群組。或

者，您可以選擇外部身分來源，方法是使用 SAML 2.0 [連線至您的外部身分提供者](#)，並使用 SCIM [自動佈建](#)使用者和群組，或是使用 [AWS Directory Service 連線至您的 Microsoft AD 目錄](#)。身分來源設定後，您可以透過在您的[許可集](#)中定義最低許可政策的方式，指派使用者和群組對 AWS 帳戶的存取權。您的員工使用者可以進行身分驗證的方式包括：透過您的集中身分提供者登入 [AWS 存取入口網站](#)以及對 AWS 帳戶和指派給他們的雲端應用程式進行單一登入。您的使用者可以設定 [AWS CLI v2](#) 以透過 Identity Center 進行身分驗證，並取得憑證來執行 AWS CLI 命令。Identity Center 也允許透過單一登入方式存取 AWS 應用程式，例如 [Amazon SageMaker AI Studio](#) 和 [AWS IoT IoT Sitewise Monitor 入口網站](#)。

依照上述指引進行後，您的員工使用者在 AWS 上管理工作負載時，將不再需要使用 IAM 使用者和群組，可直接正常操作。反之，您的使用者和群組將在 AWS 外部進行管理，而且使用者能夠以聯合身分存取 AWS 資源。聯合身分會使用您的集中式身分提供者所定義的群組。您應該識別並移除您的 AWS 帳戶中不再需要的 IAM 群組、IAM 使用者和長期存在的使用者憑證 (密碼和存取金鑰)。您可以使用 [IAM 憑證報告找到未使用的憑證](#)，[刪除相應的 IAM 使用者](#)和[刪除 IAM 群組](#)。您可以對組織套用[服務控制政策 \(SCP\)](#) 以協助防止建立新的 IAM 使用者和群組，並強制透過聯合身分存取 AWS。

Note

您負責處理 SCIM 存取權杖的輪換，如[自動佈建](#)文件中所述。此外，您也負責輪換支援您的聯合身分的憑證。

應用程式使用者的指引：您可以使用 [Amazon Cognito](#) 做為您的集中式身分提供者，來管理應用程式 (例如行動應用程式) 使用者的身分。Amazon Cognito 可為您的 Web 和行動應用程式啟用身分驗證、授權和使用者管理功能。Amazon Cognito 提供了可擴展至數百萬使用者的身分存放區、可支援社交與企業聯合身分，並且提供進階安全功能來協助保護您的使用者和業務。您可以將自訂 Web 或行動應用程式與 Amazon Cognito 整合，在幾分鐘內就能在應用程式中新增使用者身分驗證和存取控制。Amazon Cognito 是以 SAML 和 Open ID Connect (OIDC) 等開放身分標準為基礎所建置，可支援各種不同的合規法規，並與前端和後端開發資源整合。

實作步驟

員工使用者存取 AWS 的步驟

- 使用下列其中一種方法，透過集中式身分提供者將您的員工使用者聯合至 AWS：
 - 使用 IAM Identity Center 透過與您的身分提供者聯合，對您的 AWS 組織中的多個 AWS 帳戶啟用單一登入。
 - 使用 IAM 將您的身分提供者直接連接到各個 AWS 帳戶，以實現聯合的精細存取。

- 識別並移除已由聯合身分取代的 IAM 使用者和群組。

應用程式使用者的步驟

- 使用 Amazon Cognito 作為應用程式的集中式身分提供者。
- 使用 OpenID Connect 和 OAuth 將您的自訂應用程式與 Amazon Cognito 整合。您可以使用 Amplify 程式庫來開發自訂應用程式，這些程式庫提供了簡單的介面，可與各種不同的 AWS 服務進行整合，例如用於身分驗證的 Amazon Cognito。

資源

相關的最佳實務：

- [SEC02-BP06 採用使用者群組和屬性](#)
- [SEC03-BP02 授予最低權限存取權](#)
- [SEC03-BP06 根據生命週期管理存取](#)

相關文件：

- [AWS 中的聯合身分](#)
- [IAM 中的安全最佳實務](#)
- [AWS Identity and Access Management 最佳實務](#)
- [IAM Identity Center 委派管理入門](#)
- [如何在 IAM Identity Center 中針對進階使用案例使用客戶受管政策](#)
- [AWS CLI v2 : IAM Identity Center 憑證提供者](#)

相關影片：

- [AWS re:Inforce 2022 - AWS Identity and Access Management \(IAM\) 深入剖析](#)
- [AWS re:Invent 2022 - 使用 IAM Identity Center 簡化現有的員工存取權](#)
- [AWS re:Invent 2018 : 在每一層都能掌握身分](#)

相關範例：

- [研討會：使用 AWS IAM Identity Center 實現強大的身分管理](#)

- [研討會：無伺服器身分](#)

相關工具：

- [AWS 安全能力合作夥伴：身分和存取管理](#)
- [saml2aws](#)

SEC02-BP05 定期稽核和輪換憑證

定期稽核和輪換憑證以限制憑證可用來存取資源的時限。長期憑證會產生許多風險，而透過定期輪換長期憑證可以降低這些風險。

預期成果：實作憑證輪換以協助降低與使用長期憑證關聯的風險。定期稽核和修復不符合憑證輪換政策的情況。

常見的反模式：

- 沒有稽核憑證的使用。
- 不必要地使用長期憑證。
- 使用長期憑證並且未定期輪換。

未建立此最佳實務時的曝險等級：高

實作指引

當您無法倚賴臨時憑證且需要長期憑證時，請稽核憑證以確保強制執行既定的控制 (例如 [多重要素驗證 \(MFA\)](#))、定期輪換，並且具備適當的存取層級。

定期驗證 (最好是透過自動化工具) 是確認強制執行正確的控制項的必要項目。對於真人身分，您應要求使用者定期變更密碼，並淘汰存取金鑰而改用臨時憑證。隨著您從 AWS Identity and Access Management (IAM) 使用者移向集中式身分，您可以 [產生憑證報告](#) 以稽核您的使用者。

我們也建議您在身分提供者中強制和監控 MFA。您可以設定 [AWS Config 規則](#) 或使用 [AWS Security Hub 安全標準](#) 來監控使用者是否已啟用 MFA。請考慮使用 [IAM Roles Anywhere](#) 為機器身分提供臨時憑證。在無法使用 IAM 角色和臨時憑證的情況下，必須經常稽核和輪換存取金鑰。

實作步驟

- 定期稽核憑證：稽核身分提供者和 IAM 中設定的身分有助於確認只有已授權的身分才能存取您的工作負載。此類身分可能包括但不限於 IAM 使用者、AWS IAM Identity Center 使用者、Active Directory 使用者，或不同上游身分提供者中的使用者。例如，移除離職的人員和移除不再需要的跨帳戶角色。對於 IAM 實體存取的服務，備妥程序來定期稽核許可。這有助您識別需要修改的政策，以移除任何不使用的許可。使用憑證報告和 [AWS Identity and Access Management Access Analyzer](#) 來稽核 IAM 憑證和許可。您可以使用 [Amazon CloudWatch](#) 來設定對特定 API 呼叫 (在 [AWS 環境中呼叫](#)) 的警示。[Amazon GuardDuty](#) 也可以向您提醒未預期的活動，這可指出對 IAM 憑證過於寬鬆的存取或意外存取。
- 定期輪換憑證：當您無法使用臨時憑證時，定期輪換長期 IAM 存取金鑰 (最長每 90 天)。如果在您不知情的情況下意外洩漏了存取金鑰，這可限制憑證可用來存取資源的時限。如需有關輪換 IAM 使用者的存取金鑰的詳細資訊，請參閱 [輪換存取金鑰](#)。
- 審查 IAM 許可：為提升 AWS 帳戶的安全，請定期審查和監控每個 IAM 政策。確認政策遵守最低權限的原則。
- 考慮自動化 IAM 資源建立和更新：[IAM Identity Center](#) 會自動執行許多 IAM 任務，例如角色和政策管理。或者，可以使用 AWS CloudFormation 自動化 IAM 資源 (包括角色和政策) 的部署，以減少人為錯誤，因為可針對範本進行驗證和版本控制。
- 對於機器身分，使用 IAM Roles Anywhere 取代 IAM 使用者：[IAM Roles Anywhere](#) 可讓您在傳統上無法使用的區域 (例如內部部署伺服器) 使用角色。IAM Roles Anywhere 使用可信的 [X.509 憑證](#) 向 AWS 進行身分驗證及接收臨時憑證。使用 IAM Roles Anywhere 讓您無需輪換這些憑證，因為長期憑證不再儲存於內部部署環境中。請注意，您將需要監視 X.509 憑證，並在快到期時輪換。

資源

相關的最佳實務：

- [SEC02-BP02 使用臨時憑證](#)
- [SEC02-BP03 安全地儲存和使用機密](#)

相關文件：

- [開始使用 AWS Secrets Manager](#)
- [IAM 最佳實務](#)
- [身分提供者與聯合](#)

- [安全合作夥伴解決方案：存取與存取控制](#)
- [暫時安全憑證](#)
- [取得 AWS 帳戶 的憑證報告](#)

相關影片：

- [大規模管理、擷取和輪換機密的最佳實務](#)
- [使用 AWS IAM Identity Center 大規模管理使用者許可](#)
- [在每一層都能掌握身分](#)

相關範例：

- [Well-Architected 實驗室 - 自動化 IAM 使用者清理](#)
- [Well-Architected 實驗室 - 自動化 IAM 群組和角色的部署](#)

SEC02-BP06 採用使用者群組和屬性

根據使用者群組和屬性定義許可，有助於減少政策的數量並降低複雜性，因而更容易實現最低權限原則。您可以利用使用者群組，根據人員在組織中的職務，從單一位置管理多人的許可。部門、專案或位置這類屬性可在人員職務相似的情況下，針對不同的資源子集提供另一層許可範圍。

預期成果：您可以根據職務，針對執行該職務的所有使用者套用許可變更。群組成員資格和屬性會控管使用者許可，進而減少管理個別使用者層級許可的需求。您在身分提供者 (IdP) 中定義的群組和屬性會自動傳播到您的 AWS 環境。

常見的反模式：

- 管理個別使用者的許可，並且針對許多使用者重複此操作。
- 為群組定義的層級過高，授予的許可範圍過廣。
- 為群組定義的層級過於精細，致使成員資格發生重複和混淆的情形。
- 在可以使用屬性替代的情況下，仍使用在資源子集中具有重複許可的群組。
- 未透過整合在您 AWS 環境中的標準化身分提供者管理群組、屬性和成員資格。
- 使用 AWS IAM Identity Center 工作階段時使用角色鏈結

未建立此最佳實務時的曝險等級：中

實作指引

AWS 許可是在與主體 (例如使用者、群組、角色或資源) 相關聯且稱為政策的文件中定義。您可以根據工作職能、工作負載及 SDLC 環境來組織許可指派 (群組、許可、帳戶)，藉此調整許可管理的規模。針對您的員工，這可讓您根據使用者為組織執行的職務，而不是所存取的資源來定義群組。例如，WebAppDeveloper 群組可能已附加政策，用於設定開發帳戶內的 Amazon CloudFront 等服務。AutomationDeveloper 群組可能有部分許可與 WebAppDeveloper 群組重疊。您可在個別政策中擷取這些常用的許可，並讓這些許可同時與這兩個群組相關聯，而不要讓這兩種職務的使用者屬於 CloudFrontAccess 群組。

除了群組之外，您還可以使用屬性來進一步設定存取範圍。例如，您的 WebAppDeveloper 群組中的使用者可能有「專案」屬性，可用來將其專案特定的資源設定至存取範圍內。如果使用此技術，則不需要在所擁有許可相同的情況下，針對處理不同專案的應用程式開發人員建立不同的群組。您在許可政策中參照屬性的方式是根據其來源，無論其定義為聯合通訊協定的一部分 (例如 SAML、OIDC 或 SCIM)、為自訂 SAML 聲明，還是在 IAM Identity Center 內設定。

實作步驟

1. 確定您將定義群組和屬性的位置：
 - a. 依照 [SEC02-BP04 仰賴集中式身分提供者](#) 中的指引，您可以確定是否需要在身分提供者內、IAM Identity Center 內或在特定帳戶中使用 IAM 使用者群組定義群組和屬性。
2. 定義群組：
 - a. 根據職務和所需的存取範圍確定您的群組。考慮使用階層式結構或命名慣例來有效組織群組。
 - b. 如果在 IAM Identity Center 內定義，請建立群組，並使用許可集關聯所需的存取層級。
 - c. 如果在外部身分提供者內定義，請確定提供者是否支援 SCIM 通訊協定，並考慮在 IAM Identity Center 內啟用自動佈建。此功能可在您的提供者與 IAM Identity Center 之間同步群組的建立、成員資格和刪除。
3. 定義屬性：
 - a. 如果您使用外部身分提供者，SCIM 和 SAML 2.0 通訊協定預設都會提供特定屬性。您可以使用 SAML 聲明與 `https://aws.amazon.com/SAML/Attributes/PrincipalTag` 屬性名稱來定義和傳遞其他屬性。如需定義和設定自訂屬性的指引，請參閱身分提供者的文件。
 - b. 如果您在 IAM Identity Center 內定義角色，請啟用屬性型存取控制 (ABAC) 功能並視需要定義屬性。考慮符合組織結構或資源標記策略的屬性。

如果您需要透過 IAM Identity Center 擔任的 IAM 角色進行 IAM 角色鏈結，則 `source-identity` 和 `principal-tags` 這類值就不會傳播。如需詳細資訊，請參閱 [啟用和設定存取控制的屬性](#)。

1. 根據群組和屬性設定許可的範圍：

- a. 考慮在許可政策中加入條件，用來比較您主體的屬性與所存取資源的屬性。例如，您可以定義一項條件，規定僅在 PrincipalTag 條件索引鍵的值與相同名稱的 ResourceTag 索引鍵的值相符時，才允許對相關資源的存取。
- b. 定義 ABAC 政策時，請遵循 [ABAC 授權](#) 最佳實務和範例中的指引。
- c. 隨著組織的需求發展，定期審查並更新群組和屬性結構，以確保最佳的許可管理。

資源

相關的最佳實務：

- [SEC02-BP04 仰賴集中式身分提供者](#)
- [SEC03-BP02 授予最低權限存取權](#)
- [COST02-BP04 實作群組和角色](#)

相關文件：

- [IAM 最佳實務](#)
- [在 IAM Identity Center 中管理身分](#)
- [AWS 的 ABAC 是什麼？](#)
- [IAM Identity Center 中的 ABAC](#)
- [ABAC 政策範例](#)

相關影片：

- [使用 AWS IAM Identity Center 大規模管理使用者許可](#)
- [在每一層都能掌握身分](#)

許可管理

管理許可，以控制對需要存取 AWS 和工作負載的人員和機器身分的存取。您可利用許可來控制誰可以在何種條件下存取哪些內容。將許可設定為特定的真人和機器身分，即可授予對特定資源上特定服務動作的存取權。此外，您可以指定必須為 true 才能授予存取的條件。

有多種方法可以授予對不同類型資源的存取權。其中一種方法是使用不同的政策類型。

IAM 中 [以身分為基礎的政策](#) 是受管或內嵌的政策，並連接到 IAM 身分，包括使用者、群組或角色。這些政策可讓您指定該身分可以執行哪些動作 (其許可)。可以進一步將身分型政策分類。

受管政策：為獨立存在的身分型政策，可連接到您 AWS 帳戶中的多個使用者、群組和角色。受管政策有兩種：

- AWS 受管政策：由 AWS 建立和管理的受管政策。
- 客戶管理政策：您在 AWS 帳戶中建立和管理的受管政策。客戶受管政策比 AWS 受管政策更能精確地控制您的政策。

受管政策是套用許可的慣用方法。不過，您也可以使用內嵌政策，直接將其新增到單一使用者、群組或角色。內嵌政策可在政策與身分之間維持嚴格的一對一關聯性。刪除身分時也會刪除內嵌政策。

在大多數情況下，您應該建立自己的客戶受管政策，並遵循 [最低權限原則](#)。

[以資源為基礎的政策](#) 會連接至資源。例如，S3 儲存貯體是資源型政策。這些政策會將許可授予與資源位於同一帳戶的主體，或位於另一個帳戶的主體。如需支援以資源為基礎的政策服務清單，請參閱 [可搭配 IAM 運作的 AWS 服務](#)。

[許可界限](#) 使用受管政策，設定管理員可以設定的最大許可。這可讓您將建立和管理許可的功能委派給開發人員，例如建立 IAM 角色，但限制其可以授予的許可，讓他們無法利用自己建立的內容提升許可。

AWS 中的 [屬性型存取控制 \(ABAC\)](#) 可讓您根據屬性授予許可權，稱為標籤。這些標籤可以連接至 IAM 主體 (使用者或角色) 和 AWS 資源。管理員可以建立可重複使用的 IAM 政策，而這些政策會根據 IAM 主體的屬性套用許可權。例如，身為管理員的您可以使用單一 IAM 政策，授予組織中的開發人員權限以存取符合其專案標籤的 AWS 資源。隨著開發人員團隊將資源新增至專案，會根據屬性自動套用許可權，這樣就不需要針對每個新資源來更新政策。

[組織服務控制政策 \(SCP\)](#) 為組織或組織單位 (OU) 的帳戶成員定義最大許可。SCP 會限制身分型政策或資源型政策為帳戶中實體 (使用者或角色) 授予的許可，但不會授予許可。

[工作階段政策](#) 會擔任角色或聯合身分使用者。在使用 AWS CLI 或 AWS API 工作階段政策時傳遞工作階段政策，以限制角色或使用者的身分型政策授予工作階段的許可。這些政策會限制已建工作階段的許可，但不會授予許可。如需詳細資訊，請參閱 [工作階段政策](#)。

最佳實務

- [SEC03-BP01 定義存取需求](#)
- [SEC03-BP02 授予最低權限存取權](#)

- [SEC03-BP03 建立緊急存取程序](#)
- [SEC03-BP04 持續減少許可](#)
- [SEC03-BP05 為您的組織定義許可防護機制](#)
- [SEC03-BP06 根據生命週期管理存取](#)
- [SEC03-BP07 分析公有和跨帳戶存取權](#)
- [SEC03-BP08 在組織內安全地共用資源](#)
- [SEC03-BP09 安全地與第三方共用資源](#)

SEC03-BP01 定義存取需求

管理員、終端使用者或其他元件都需要存取工作負載的每個元件或資源。請明確定義應該有權存取每個元件的人員和機器，選擇適當的身分類型及驗證和授權方法。

常見的反模式：

- 將機密硬式編碼或儲存在應用程式中。
- 為每名使用者授予自訂許可。
- 使用長期憑證。

未建立此最佳實務時的曝險等級：高

實作指引

管理員、終端使用者或其他元件都需要存取工作負載的每個元件或資源。請明確定義應該有權存取每個元件的人員和機器，選擇適當的身分類型及驗證和授權方法。

應提供組織內對 AWS 帳戶 的定期存取 (使用[聯合存取](#)或集中式身分提供者)。您應集中進行身分管理，並確保有既定的實務，可將 AWS 存取整合至員工的存取生命週期。例如，當員工改為擔任具有不同存取層級的任務角色時，其群組成員資格也應變更，以反映新的存取需求。

為非真人身分定義存取需求時，請判斷哪些應用程式和組成部分需要存取權，以及如何授予許可。使用透過最低權限存取模型建置的 IAM 角色是建議的方法。[AWS 受管政策](#)提供預先定義的 IAM 政策，其中涵蓋最常見的使用案例。

AWS 服務 (例如 [AWS Secrets Manager](#) 和 [AWS Systems Manager Parameter Store](#)) 可以協助在無法使用 IAM 角色的情況下，將機密從應用程式或工作負載中安全地分離。在 Secrets Manager 中，您可

以為憑證建立自動輪換。您可以使用 Systems Manager 來參考指令碼、命令、SSM 文件、組態和自動化工作流程中的參數，方法是使用您在建立參數時指定的唯一名稱。

您可以使用 [AWS IAM Roles Anywhere](#) 取得 [IAM 中的臨時安全憑證](#)，以用於在 AWS 外部執行的工作負載。工作負載可以透過與 AWS 應用程式搭配使用的相同 [IAM 政策](#) 和 [IAM 角色](#) 來存取 AWS 資源。

可能的話，請選擇短期臨時憑證，而不是長期靜態憑證。對於您希望使用者具備程式設計存取權和長期憑證的情況，請使用 [存取金鑰上次使用的資訊](#) 來輪換和移除存取金鑰。

若使用者想要與 AWS Management Console 之外的 AWS 互動，則需要程式設計存取權。授予程式設計存取權的方式取決於存取 AWS 的使用者類型。

若要授與使用者程式設計存取權，請選擇下列其中一個選項。

哪個使用者需要程式設計存取權？	到	根據
人力資源身分 (IAM Identity Center 中管理的使用者)	使用臨時憑證簽署對 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計請求。	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> 關於 AWS CLI，請參閱 AWS Command Line Interface 使用者指南 中的 設定 AWS CLI 來使用 AWS IAM Identity Center。 關於 AWS SDKs、工具和 AWS APIs，請參閱 AWSSDKs 和工具參考指南 中的 IAM Identity Center 驗證。
IAM	使用臨時憑證簽署對 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計請求。	請遵循 IAM 使用者指南中 使用臨時憑證搭配 AWS 資源 中的指示。
IAM	(不建議使用) 使用長期憑證簽署 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計要求。	請依照您要使用的介面所提供的指示操作。

哪個使用者需要程式設計存取權？	到	根據
		<ul style="list-style-type: none">• 關於 AWS CLI，請參閱 AWS Command Line Interface 使用者指南 中的 使用 IAM 使用者憑證進行驗證。• 關於 AWS SDKs 和工具，請參閱 AWSSDKs 和工具參考指南 中的 使用長期憑證進行驗證。• 關於 AWS API，請參閱 IAM 使用者指南 中的 管理 IAM 使用者的存取金鑰。

資源

相關文件：

- [屬性型存取控制 \(ABAC\)](#)
- [AWS IAM Identity Center](#)
- [IAM Roles Anywhere](#)
- [IAM Identity Center 的 AWS 受管政策](#)
- [AWS IAM 政策條件](#)
- [IAM 使用案例](#)
- [移除不必要的憑證](#)
- [使用 政策](#)
- [如何根據 AWS 帳戶、OU 或組織控制對 AWS 資源的存取權](#)
- [使用 AWS Secrets Manager 中增強的搜尋功能來輕鬆識別、安排和管理機密](#)

相關影片：

- [在 60 分鐘內精通 IAM 政策](#)

- [責任區隔、最低權限、委派和 CI/CD](#)
- [簡化身分和存取管理，以促進創新](#)

SEC03-BP02 授予最低權限存取權

僅授予使用者在特定情況下對特定資源執行特定動作所需的存取權。使用群組和身分屬性大規模動態設定許可，而不是定義個別使用者的許可。例如，您可以允許一組開發人員的存取權，以只管理其專案的資源。如此一來，當開發人員退出專案時，其存取權將自動被撤銷，而無需變更基礎存取政策。

預期成果：使用者只擁有其特定工作職能所需的最低許可。您可以分開 AWS 帳戶來隔離開發人員與實際執行環境。當開發人員需要存取特定任務的實際執行環境時，他們會獲得授予有限且受控的存取權，且僅限於任務期間有效。他們的實際執行存取權會在完成必要的工作後立即撤銷。您定期審查許可，並在不再需要時立即撤銷許可，例如，使用者變更角色或離開組織時。您可以將管理員權限侷限在一小群可信任的群組，以減少暴險。您僅給予機器或系統帳戶執行其預期任務所需的最低許可。

常見的反模式：

- 根據預設，您會對使用者授予管理員許可。
- 您使用根使用者帳戶從事日常活動。
- 您建立過度寬鬆的政策，而未適當界定範圍。
- 您不常檢閱許可，這會導致許可滲透。
- 您完全依賴屬性型存取控制來隔離環境或管理許可。

未建立此最佳實務時的曝險等級：高

實作指引

[最低權限](#)原則指出，僅應允許身分執行完成特定任務所需的最小動作集。這平衡了可用性、效率和安全性。根據此原則運作有助於限制意外存取，也有助於追蹤誰有權存取哪些資源。IAM 使用者和角色在預設情況下沒有任何許可。根使用者預設擁有完整存取權，應受到嚴格監控，並僅用於[需要根存取權的任務](#)。

IAM 政策用於明確授予許可給 IAM 角色或特定資源。例如，以身分為基礎的政策可以連接到 IAM 群組，而 S3 儲存貯體可由以資源為基礎的政策控制。

建立 IAM 政策時，您就可以指定必須為 true 的服務動作、資源和條件，以便 AWS 允許或拒絕存取。AWS 支援各種條件，以協助您縮減存取權範圍。例如，透過使用 PrincipalOrgID [條件金鑰](#)，就可以在請求者不屬於您的 AWS 組織時拒絕動作。

此外，您還可以使用 [CalledVia](#) 條件金鑰控制 AWS 服務代您發出的請求，例如建立 AWS Lambda 函數的 AWS CloudFormation。您可以將不同的政策類型分層，以便建立深度防禦並限制使用者的整體許可。您還可以限制在什麼條件下，可以授予哪些許可。例如，您可以允許工作負載團隊為他們建置的系統建立自己的 IAM 政策，但前提是須套用 [許可界限](#) 來限制他們可授予的最大許可。

實作步驟

- **實作最低權限政策**：將具有最低權限的存取政策指派給 IAM 群組和角色，以反映您已定義的使用者角色或職能。
- **透過分隔 AWS 帳戶 來隔離開發和實際執行環境**：針對開發和實際執行環境使用不同的 AWS 帳戶，並使用 [服務控制政策](#)、資源政策和身分政策來控制兩者之間的存取權。
- **API 使用方式的基本政策**：確定所需許可的一種方法是檢閱 AWS CloudTrail 日誌。您可以利用這種檢閱根據使用者在 AWS 內實際執行的動作，建立適合的許可。[IAM Access Analyzer](#) 可根據存取活動 [自動產生](#) IAM 政策。您可以在組織或帳戶層級使用 IAM Access Advisor 來 [追蹤特定政策的最後存取資訊](#)。
- **考慮使用 [AWS 受管政策來執行工作職能](#)**：當您開始制定精細的許可政策時，針對帳單、資料庫管理員和資料科學家等常見的任務角色使用 AWS 受管政策會很實用。這些政策可在您決定如何實施最低權限政策的同時，協助縮小使用者的存取權。
- **移除不必要的許可**：偵測並移除未使用的 IAM 實體、憑證和許可，以實踐最低權限原則。您可以使用 [IAM Access Analyzer](#) 來識別外部和未使用的存取權，而 [IAM Access Analyzer 政策產生](#) 功能可協助微調許可政策。
- **確保使用者對生產環境具有有限的存取權**：使用者應只能存取具有有效使用案例的生產環境。在使用者執行完需要生產存取權的特定任務後，就應撤銷存取權。限制對生產環境的存取，有助於防止發生會影響生產的意外事件，也能降低意外存取的影響範圍。
- **考慮使用許可界限**：[許可界限](#) 是一種使用受管政策功能，其主要設定以身分為基礎的政策可授予 IAM 實體的最大許可權限。實體的許可界限可讓其只執行由身分類型政策和其許可界限同時允許的動作。
- **使用屬性型存取控制和資源標籤讓存取權更精簡**：(若支援) 使用資源標籤的 [屬性型存取控制 \(ABAC\)](#) 可用來精簡許可。您可以使用 ABAC 模型，此模型會比較主體標籤與資源標籤，以根據您定義的自訂維度來精簡存取權。這種方法可以簡化並減少組織中的許可政策數量。
 - 建議僅在主體和資源皆為您的 AWS Organization 所擁有時，才使用 ABAC 進行存取控制。外部對象可以將與您的組織相同的標籤名稱和值，用於其自有主體和資源。如果您完全依賴這些名稱-值對來對外部對象的主體或資源授予存取權，則可能會提供意外的許可。
- **使用 AWS Organizations 的服務控制政策**：[服務控制政策](#) 可集中控制組織中成員帳戶的最大可用許可數。重要的是，您可以使用服務控制政策來限制成員帳戶中的根使用者許可。此外，還可以考

慮使用 AWS Control Tower，它提供規範性受管控制，可以豐富 AWS Organizations。您也可以 Control Tower 中定義自己的控制。

- 為您的組織制定使用者生命週期政策：使用者生命週期政策定義了當使用者上線至 AWS、變更職務或範圍或不再需要存取 AWS 時要執行的任務。在使用者生命週期的每個步驟進行許可審查，以確定許可受到適當限制並避免許可滲透的問題。
- 建立定期審查許可的排程，並移除任何不需要的許可：您應該定期審查使用者存取權，確認使用者沒有過度寬鬆的存取權。[AWS Config](#) 和 IAM Access Analyzer 可在使用者許可稽核過程中提供協助。
- 建立職務矩陣：職務矩陣會以視覺化的方式顯示您 AWS 足跡內所需的各種角色和存取層級。利用職務對照表，即可根據組織內的使用者職責定義和區分許可。使用群組，而不是將許可直接套用至個別使用者或角色。

資源

相關文件：

- [授予最低權限](#)
- [IAM 實體的許可界限](#)
- [用於寫入最低權限 IAM 政策的技巧](#)
- [IAM Access Analyzer 會根據存取活動產生 IAM 政策，因此更容易實作最低權限許可](#)
- [使用 IAM 許可界限將許可管理委派給開發人員](#)
- [使用上次存取的資訊精簡許可](#)
- [IAM 政策類型以及何時使用這些政策](#)
- [使用 IAM 政策模擬器測試 IAM 政策](#)
- [AWS Control Tower 中的防護機制](#)
- [零信任架構：AWS 觀點](#)
- [如何使用 CloudFormation StackSets 實作最低權限原則](#)
- [屬性型存取控制 \(ABAC\)](#)
- [檢視使用者活動以縮小政策範圍](#)
- [檢視角色存取](#)
- [使用標記來組織您的環境並推動問責制](#)
- [AWS 標記策略](#)

- [標記 AWS 資源](#)

相關影片：

- [下一代許可管理](#)
- [零信任：AWS 觀點](#)

相關範例：

- [實驗室：IAM 許可界限委派角色建立](#)
- [實驗室：EC2 的 IAM 標籤型存取控制](#)

SEC03-BP03 建立緊急存取程序

建立一項程序，在集中式身分提供者發生問題時，緊急存取您的工作負載。

您必須針對可能導致緊急事件發生的不同故障模式設計不同的程序。例如，正常情況下，您的員工使用者會使用集中式身分提供者 ([SEC02-BP04](#)) 聯合至雲端，以管理其工作負載。不過，如果您的集中式身分提供者發生錯誤，或是雲端中聯合的組態經過修改，則您的員工使用者可能無法連至雲端。緊急存取程序可讓授權的管理員透過替代方式 (例如聯合或直接使用者存取的替代形式) 存取您的雲端資源，以修正聯合組態或工作負載的問題。緊急存取程序會持續使用，直到恢復正常聯合機制為止。

預期成果：

- 您已定義並記載可視為緊急情況的故障模式：請考慮正常情況以及使用者用來管理工作負載的系統。考慮這些相依性如何發生錯誤並導致緊急情況。您可以在[可靠性支柱](#)中找到問題與最佳實務，有助於識別故障模式並架構更具彈性的系統，以盡量降低故障的可能性。
- 您已記載確認故障為緊急情況須遵循的步驟。例如，您可以要求身分管理員檢查主要和待命身分提供者的狀態，如果兩者都無法使用，則發布身分提供者發生錯誤緊急事件。
- 您已針對每一種緊急或故障模式類型定義了緊急存取程序。明確定義可減少部分使用者過度使用一般程序，來處理所有類型的緊急情況。您的緊急存取程序描述了各個程序應在何種情況下使用，以及不應在哪些情況下使用，並指出可能適用的替代程序。
- 您的程序完整記載了詳細指示和程序手冊，可快速有效地遵循。請記住，緊急事件對使用者來說可能會非常緊張，他們可能面對極大的時間壓力，因此程序的設計應盡可能簡單。

常見的反模式：

- 您沒有詳細記載且經過充分測試的緊急存取程序。您的使用者未準備好面對緊急情況，而在緊急事件發生時只能隨機應變。
- 您的緊急存取程序與正常存取機制依賴相同的系統 (例如集中式身分提供者)。這表示，一旦這類系統發生故障，就可能同時影響您的正常和緊急存取機制，並損及您從故障復原的能力。
- 您的緊急存取程序用在非緊急情況。例如，您的使用者經常濫用緊急存取程序，因為他們發現直接進行變更透過管道提交變更容易。
- 您的緊急存取程序未產生足夠的日誌來稽核程序，或是日誌未受監控，無法在發生可能濫用程序的情況時發出提醒。

建立此最佳實務的優勢：

- 只要有詳細記載且經充分測試的緊急存取程序，就能縮短使用者回應和解決緊急事件所花的時間。這樣就能進一步減少停機時間，並為客戶帶來更高的服務可用性。
- 您可以追蹤每一項緊急存取請求，以及偵測未經授權的人士試圖濫用程序來處理非緊急事件的情況，並發出提醒。

未建立此最佳實務時的風險暴露等級：中

實作指引

本節提供建立緊急存取程序的指引，用於處理與 AWS 上部署的工作負載相關的數種故障模式，一開始先介紹適用於所有故障模式的通用指引，接著再根據故障模式類型說明特定指引。

適用所有故障模式的通用指引

為故障模式設計緊急存取程序時，請考慮下列事項：

- 記載程序的前提和假設：應該和不應該使用程序的時機。這樣做有助於詳細說明故障模式並記載假設，例如其他相關系統的狀態。舉例來說，故障模式 2 的程序假設身分提供者可以使用，但 AWS 上的組態已經過修改或已過期。
- 預先建立緊急存取程序所需的資源 ([SEC10-BP05](#))。例如，在所有工作負載帳戶中預先建立具有 IAM 使用者和角色的緊急存取 AWS 帳戶，以及跨帳戶 IAM 使用者角色。這樣就可確定這些資源在緊急事件發生時立即可用。透過預先建立資源，您就不必依賴 AWS [控制平面](#) API (用來建立和修改 AWS 資源)，因為它們在緊急情況下可能無法使用。此外，預先建立 IAM 資源就不需考慮[因最終一致性而可能發生的延遲](#)。
- 請將緊急存取程序納入您的事件管理計畫中 ([SEC10-BP02](#))。記載緊急事件的追蹤方式，並傳達給組織中的其他人，例如同儕團隊、您的領導階層，以及適時向外傳達給您的客戶和業務合作夥伴。

- 在您現有的服務請求工作流程系統 (若有的話) 中定義緊急存取請求程序。一般來說，這類工作流程系統可讓您建立接收表單來收集有關請求的資訊、在工作流程的每個階段追蹤請求，以及新增自動和手動核准步驟。將每一個請求與事件管理系統中追蹤的對應緊急事件建立關聯。採用統一的緊急存取系統，可讓您在單一系統中追蹤這些請求、分析使用趨勢並改善程序。
- 確認您的緊急存取程序只能由經授權的使用者啟動，並且視情況要求使用者同儕或管理層的核准。核准程序在營業時間內外都要能夠有效運作。定義在主要核准者沒有空的情況下，如何由次要核准者核准請求，以及如何在您的管理鏈中向上呈報，直到請求獲得核准。
- 實作可靠的記錄、監控和警示機制，以因應緊急存取程序和機制。同時針對成功和失敗的嘗試產生詳細的稽核日誌，以便取得緊急存取權。在事件管理系統中將活動與正在發生的事件相互關聯，並在動作發生於預期時段之外，或在正常操作期間使用緊急存取帳戶時發出警示。緊急存取帳戶只能在緊急情況下存取，因為破窗程序也可視為後門。與您的安全資訊和事件管理 (SIEM) 工具或 [AWS Security Hub](#) 整合，以在緊急存取期間回報和稽核所有活動。恢復正常操作時，自動輪換緊急存取憑證，並通知相關團隊。
- 定期測試緊急存取程序，以確認步驟是否清楚，並且快速有效地授予正確的存取層級。您的緊急存取程序應在事件回應模擬的過程中 ([SEC10-BP07](#)) 和災難復原測試中 ([REL13-BP03](#)) 進行測試。

故障模式 1：用於聯合至 AWS 的身分提供者無法使用

如 [SEC02-BP04 仰賴集中化的身分提供者](#) 中所述，我們建議您仰賴集中式身分提供者來聯合您的員工使用者，以授予 AWS 帳戶的存取權。您可以使用 IAM Identity Center 聯合至您 AWS 組織中的多個 AWS 帳戶，或是使用 IAM 聯合至個別 AWS 帳戶。在這兩種情況下，員工使用者都會先透過集中式身分提供者進行身分驗證，然後才重新導向至 AWS 登入端點進行單一登入。

萬一您的集中式身分提供者無法使用，您的員工使用者就無法聯合至 AWS 帳戶 或管理其工作負載。在此緊急事件中，您可以提供緊急存取程序讓一小群管理員存取 AWS 帳戶，以便執行無法等到集中式身分提供者恢復連線後才處理的重要工作。例如，您的身分提供者停擺了 4 小時，而在此期間，您需要修改生產帳戶中 Amazon EC2 Auto Scaling 群組的上限，以處理客戶流量意外暴增的情況。您的緊急管理員應遵循緊急存取程序，才能獲得特定生產 AWS 帳戶的存取權並進行必要的變更。

緊急存取程序依賴預先建立的緊急存取 AWS 帳戶，該帳戶單純用於緊急存取，並擁有 AWS 資源 (例如 IAM 角色和 IAM 使用者) 可支援緊急存取程序。在正常操作期間，任何人都不應該存取緊急存取帳戶，而且您必須監控濫用此帳戶的情況並發出提醒 (如需詳細資訊，請參閱前一節「通用指引」)。

緊急存取帳戶具有緊急存取 IAM 角色，有權在需要緊急存取的 AWS 帳戶中擔任跨帳戶角色。這些 IAM 角色會預先建立並設定信任政策，以便信任緊急帳戶的 IAM 角色。

緊急存取程序可以使用下列其中一種方法：

- 您可以預先建立一組 [IAM 使用者](#) 並包含關聯的強式密碼和 MFA 字符，以供緊急存取帳戶中的緊急管理員使用。這些 IAM 使用者有權擔任 IAM 角色，且後續可在需要緊急存取時跨帳戶存取 AWS 帳戶。我們建議這類使用者的數量越少越好，並且將每一位使用者指派給單一緊急管理員。在緊急情況下，緊急管理員使用者會使用其密碼和 MFA 字符代碼登入緊急存取帳戶，切換到緊急帳戶中的緊急存取 IAM 角色，最後再切換到工作負載帳戶中的緊急存取 IAM 角色，以執行緊急變更動作。這種方法的優點是，每個 IAM 使用者都會指派給一名緊急管理員，而且您可以透過檢閱 CloudTrail 事件得知登入的使用者。缺點是，您必須維護多個 IAM 使用者及其關聯的長期存在密碼和 MFA 字符。
- 您可以使用緊急存取 [AWS 帳戶 根使用者](#) 來登入緊急存取帳戶、擔任緊急存取的 IAM 角色，並且在工作負載帳戶中擔任跨帳戶角色。我們建議您為根使用者設定強式密碼和多個 MFA 字符。同時也建議您，將密碼和 MFA 字符儲存在強制執行強式身分驗證和授權的安全企業憑證保存庫中。您應確保密碼和 MFA 字符重設要素的安全性：將帳戶的電子郵件地址設定為受到您的雲端安全管理員監控的電子郵件分發清單，並將帳戶的電話號碼設定為同樣受到安全管理員監控的共用電話號碼。這種方法的優點是，只需管理一組根使用者憑證。缺點是，由於這是共用使用者，因此有多個管理員能夠以根使用者的身分登入。您必須稽核企業保存庫日誌事件，以確定哪個管理員簽出了根使用者密碼。

故障模式 2：AWS 上的身分提供者組態已經過修改或已過期

若要讓您的員工使用者聯合至 AWS 帳戶，您可以使用外部身分提供者設定 IAM Identity Center，或建立 IAM 身分提供者 ([SEC02-BP04](#))。一般來說，您可以匯入身分提供者提供的 SAML 中繼資料 XML 文件來進行這些設定。中繼資料 XML 文件包含一個 X.509 憑證，對應於身分提供者用來簽署其 SAML 聲明的私有金鑰。

AWS 端的這些組態可能遭到管理員誤改或誤刪。另一種情況是，匯入 AWS 中的 X.509 憑證可能過期，而具有新憑證的新中繼資料 XML 尚未匯入 AWS 中。這兩種情況都可能使員工使用者的 AWS 聯合中斷，導致緊急情況發生。

在這類緊急事件中，您可以提供 AWS 的存取權給身分管理員，以修正聯合問題。例如，您的身分管理員使用緊急存取程序登入緊急存取 AWS 帳戶、切換為 Identity Center 管理員帳戶中的角色，並透過從您的身分提供者匯入最新的 SAML 中繼資料 XML 文件來更新外部身分提供者組態，以重新啟用聯合。聯合修復後，您的員工使用者繼續使用正常操作程序來聯合至其工作負載帳戶。

您可以依照先前「故障模式 1」中詳述的方法來建立緊急存取程序。您可以將最低權限許可授予身分管理員，以限制他們只能存取 Identity Center 管理員帳戶以及在該帳戶中對 Identity Center 執行動作。

故障模式 3：Identity Center 中斷

萬一發生 IAM Identity Center 或 AWS 區域中斷的情況，建議您設定一個可用來暫時存取 AWS Management Console 的組態。

緊急存取程序會在緊急帳戶中，使用您的身分提供者對 IAM 的直接聯合。如需有關程序和設計考量的詳細資訊，請參閱[設定 AWS Management Console 的緊急存取](#)。

實作步驟

適用所有故障模式的通用步驟

- 建立緊急存取程序專用的 AWS 帳戶。在帳戶中預先建立所需的 IAM 資源，例如 IAM 角色或 IAM 使用者，也可以選擇建立 IAM 身分提供者。此外，在工作負載 AWS 帳戶中預先建立跨帳戶 IAM 角色，並與緊急存取帳戶中對應的 IAM 角色建立信任關係。您可以使用 [AWS CloudFormation StackSets 搭配 AWS Organizations](#) 在組織的成員帳戶中建立此類資源。
- 建立 AWS Organizations [服務控制政策](#) (SCP) 以拒絕刪除和修改成員 AWS 帳戶中的跨帳戶 IAM 角色。
- 為緊急存取 AWS 帳戶啟用 CloudTrail，並將追蹤事件傳送到日誌收集 AWS 帳戶中的中央 S3 儲存貯體。如果您使用 AWS Control Tower 來設定和管控您的 AWS 多帳戶環境，則您使用 AWS Control Tower 建立或在 AWS Control Tower 中註冊的每個帳戶都會預設啟用 CloudTrail，並傳送至專用日誌封存 AWS 帳戶中的 S3 儲存貯體。
- 透過建立在主控台登入時比對的 EventBridge 規則來監控活動的緊急存取帳戶，以及透過緊急 IAM 角色監控 API 活動。當活動於事件管理系統中追蹤的持續緊急事件之外發生時，傳送通知給您的安全營運中心。

適用「故障模式 1：用於聯合至 AWS 的身分提供者無法使用」及「故障模式 2：AWS 上的身分提供者組態已經過修改或已過期」的其他步驟

- 根據您選擇的緊急存取機制預先建立資源：
 - 使用 IAM 使用者：預先建立 IAM 使用者並設定強式密碼和關聯的 MFA 裝置。
 - 使用緊急帳戶根使用者：設定根使用者使用強式密碼，並將密碼儲存在您的企業憑證保存庫中。將多個實體 MFA 裝置與根使用者建立關聯，並將裝置儲存在您的緊急管理員小組成員可快速存取的位置。

適用「故障模式 3：Identity Center 中斷」的其他步驟

- 如[設定 AWS Management Console 的緊急存取](#)中所述，在緊急存取 AWS 帳戶中，建立 IAM Identity Provider 身分提供者，以從您的身分提供者啟用直接 SAML 聯合。
- 在 IdP 中建立緊急操作群組，但不新增任何成員。

- 在緊急存取帳戶中建立對應於緊急操作群組的 IAM 角色。

資源

相關 Well-Architected 的最佳實務：

- [SEC02-BP04 仰賴集中式身分提供者](#)
- [SEC03-BP02 授予最低權限存取權](#)
- [SEC10-BP02 制定事件管理計畫](#)
- [SEC10-BP07 執行演練日](#)

相關文件：

- [設定 AWS Management Console 的緊急存取](#)
- [使 SAML 2.0 聯合身分使用者能夠存取 AWS Management Console](#)
- [緊急存取](#)

相關影片：

- [AWS re:Invent 2022 - 使用 IAM Identity Center 簡化現有的員工存取權](#)
- [AWS re:Inforce 2022 - AWS Identity and Access Management \(IAM\) 深入剖析](#)

相關範例：

- [AWS 緊急存取角色](#)
- [AWS 客戶程序手冊架構](#)
- [AWS 事件回應程序手冊範例](#)

SEC03-BP04 持續減少許可

在團隊確定所需的存取權時，請移除不需要的許可，並建立審查程序以達到最低權限的許可。持續監視人類和機器存取權，並移除不使用的身分和許可。

預期成果：許可政策應遵循最低權限原則。隨著工作職責和角色的定義變得更具體，您需要審查許可政策以移除不必要的許可。若憑證遭到意外洩露或以其他方式在未經授權下遭存取，此方法可縮小影響範圍。

常見的反模式：

- 預設授予使用者管理員許可。
- 建立過於寬鬆的政策，但不具完整的管理員權限。
- 保留不再需要的許可政策。

未建立此最佳實務時的曝險等級：中

實作指引

在團隊和專案剛開始時，可使用寬鬆的許可政策來激發創新和敏捷性。例如，在開發或測試環境中，可以讓開發人員存取廣泛的 AWS 服務。我們建議您持續評估存取權，並將存取權限於完成目前工作所需的這些服務和服務動作。我們建議對人類和機器身分進行此項評估。機器身分有時候稱為系統或服務帳戶，是提供 AWS 存取權給應用程式或伺服器的身分。此存取權在生產環境中尤為重要，因為過於寬鬆的許可可能影響廣大而且可能暴露客戶資料。

AWS 提供多種方法可協助識別未使用的使用者、角色、許可和憑證。AWS 也有助於分析 IAM 使用者和角色的存取活動，包括關聯的存取金鑰，以及對 AWS 資源的存取，例如 Amazon S3 儲存貯體中的物件。AWS Identity and Access Management Access Analyzer 政策產生可協助您根據某主體進行互動的實際服務和動作來建立限制性許可。[屬性型存取控制 \(ABAC\)](#) 可以協助簡化許可管理，因為您可以使用使用者的屬性提供許可給使用者，而不是將許可政策直接連接至每個使用者。

實作步驟

- 使用 [AWS Identity and Access Management Access Analyzer](#)：IAM Access Analyzer 可協助您識別組織和帳戶中[與外部實體共用的](#)資源，例如 Amazon Simple Storage Service (Amazon S3) 儲存貯體或 IAM 角色。
- 使用 [IAM Access Analyzer 政策產生](#)：IAM Access Analyzer 政策產生可協助您[根據 IAM 使用者或角色的存取活動建立精細的許可政策](#)。
- 在實際執行前測試較低環境的許可：一開始先使用[較不重要的沙盒和開發環境](#)，利用 IAM Access Analyzer 來測試各種不同工作職能所需的許可。然後在測試、品質保證和預備環境逐步加強並驗證這些許可，再將許可套用至實際執行環境。較低的環境一開始可以擁有較寬鬆的許可，因為服務控制政策 (SCP) 會透過限制授予的許可數上限來強制執行防護機制。

- 為 IAM 使用者和角色確定可接受的時間範圍和使用政策：使用[上次存取的時間戳記](#)以[識別未使用的使用者和角色](#)並將其移除。檢閱服務和動作上次存取的資訊，以識別和[設定特定使用者和角色的許可範圍](#)。例如，您可以使用上次存取的資訊來識別您的應用程式角色所需的特定 Amazon S3 動作，並將該角色的存取權僅限於這些動作。AWS Management Console 中有提供「上次存取的資訊」功能，並且可讓您透過程式設計的方式將這些功能併入基礎設施工作流程和自動化工具中。
- 考慮在[AWS CloudTrail 中日誌記錄資料事件](#)：在預設情況下，CloudTrail 不會記錄資料事件，例如 Amazon S3 物件層級活動 (如 GetObject 和 DeleteObject) 或 Amazon DynamoDB 資料表活動 (如 PutItem 和 DeleteItem)。考慮對這些事件使用日誌記錄功能，以確定哪些使用者和角色需要存取特定 Amazon S3 物件或 DynamoDB 資料表項目。

資源

相關文件：

- [授予最低權限](#)
- [移除不必要的憑證](#)
- [什麼是 AWS CloudTrail ?](#)
- [使用 政策](#)
- [DynamoDB 中的日誌記錄和監控](#)
- [對您的 Amazon S3 儲存貯體和物件使用 CloudTrail 事件日誌記錄](#)
- [取得 AWS 帳戶 的憑證報告](#)

相關影片：

- [在 60 分鐘內精通 IAM 政策](#)
- [責任區隔、最低權限、委派和 CI/CD](#)
- [AWS re:Inforce 2022 - AWS Identity and Access Management \(IAM\) 深入剖析](#)

SEC03-BP05 為您的組織定義許可防護機制

使用許可防護機制縮小可授予主體的可用許可範圍。許可政策評估鏈包括您的防護機制，可在進行授權決策時確定主體的有效許可。您可以採用分層方式定義防護機制。對整個組織廣泛套用一些防護機制，另外對暫時存取工作階段套用一些精細的防護機制。

預期成果：您可以使用個別 AWS 帳戶對環境進行明確的隔離。服務控制政策 (SCP) 用於定義整個組織的許可防護機制。較寬鬆的防護機制設定於最靠近組織根目錄的階層層級，較嚴謹的防護機制則設定於較靠近個別帳戶的層級。

在受支援的情況下，資源政策會定義主體必須滿足才能取得資源存取權的條件。資源政策也會適時縮小允許的動作範圍。許可界限會設置在管理工作負載許可的主體上，以將許可管理委派給個別工作負載擁有者。

常見的反模式：

- 在 [AWS 組織](#) 內建立成員 AWS 帳戶，但未使用 SCP 來限制其根憑證適用的用途和許可。
- 根據最低權限指派許可，但未對可授予的許可集上限設置防護機制。
- 仰賴 AWS IAM 的隱含拒絕基礎來限制許可，相信政策不會授予不需要的明確允許許可。
- 在相同 AWS 帳戶中執行多個工作負載環境，然後仰賴 VPC、標籤或資源政策等機制來強制執行許可界限。

建立此最佳實務的優勢：許可防護機制有助於建立信心，確保不會有不需要的許可授予情況，即使許可政策嘗試這樣做也不必擔心。此最佳實務可透過縮小需考量的許可範圍上限來簡化定義和管理許可。

未建立此最佳實務時的曝險等級：中

實作指引

建議您採用分層方式為您的組織定義許可防護機制。此方法能夠隨著套用額外的分層，有系統地減少可能的許可集上限。這可協助您根據最低權限原則授予存取權，降低了因政策組態錯誤導致意外存取的風險。

建立許可防護機制的第一步，是將您的工作負載和環境隔離到個別 AWS 帳戶中。在沒有明確許可的情況下，某一帳戶中的主體無法存取另一帳戶中的資源，即使兩個帳戶在相同 AWS 組織中或在相同 [組織單位 \(OU\)](#) 下亦是如此。您可以使用 OU 將您要管理的帳戶分組為一個單位。

下一步是減少您可授予組織的成員帳戶內主體的許可集上限。您可以使用 [服務控制政策 \(SCP\)](#) 達到此目的，這些政策可套用至 OU 或帳戶。SCP 可強制執行通用的存取控制，例如限制對特定 AWS 區域的存取、協助防止資源遭到刪除，或停用有潛在風險的服務動作。您套用至組織根目錄的 SCP 只會影響其成員帳戶，而不會影響管理帳戶。SCP 只會控管組織內的主體。您的 SCP 不會控管組織外部存取您資源的主體。

如果您使用 [AWS Control Tower](#)，則可以利用其 [控制項](#) 和 [登陸區域](#) 做為許可防護機制和多帳戶環境的基礎。登陸區域提供預先設定的安全基準環境，當中採用不同的帳戶來處理不同的工作負載和應用程

式。防護機制透過服務控制政策 (SCP)、AWS Config 規則及其他組態的組合，強制執行有關安全、操作和合規的強制性控制。不過，同時使用 Control Tower 防護機制和登陸區域與自訂 Organization SCP 時，務必遵循 AWS 文件中所述的最佳實務，以避免衝突並確保適當的控管。如需在 Control Tower 環境中管理 SCP、帳戶和組織單位 (OU) 的詳細建議，請參閱 [AWS Organizations 的 AWS Control Tower 指引](#)。

只要遵循這些指引，您就能有效利用 Control Tower 的防護機制、登陸區域和自訂 SCP，同時減少前在衝突，並確保適當控管與控制多帳戶 AWS 環境。

再下一步是使用 [IAM 資源政策](#) 來設定您可對其控管的資源執行的可用動作範圍，以及設定執行動作的主體必須符合的任何條件。這個範圍可以很廣泛，例如只要主體屬於組織的一部分就允許所有動作 (使用 PrincipalOrgId [條件金鑰](#))，也可以很精細，例如只允許特定 IAM 角色執行特定動作。您可以在 IAM 角色信任政策中採取類似方法，並附帶條件。如果資源或角色信任政策明確指名相同帳戶中的某個主體做為其控管的角色或資源，則該主體不需要有授予相同許可的連接 IAM 政策。如果主體位於與資源不同的帳戶中，則該主體確實需要有授予這些許可的連接 IAM 政策。

通常，工作負載團隊會希望管理其工作負載需要的許可。這樣一來，他們便需要建立新的 IAM 角色和許可政策。您可以擷取允許團隊在 [IAM 許可界限](#) 中授予的許可範圍上限，並將此文件與 IAM 角色關聯，之後團隊就可使用該角色來管理其 IAM 角色和許可。這種方法可讓他們彈性地完成其工作，同時降低擁有 IAM 管理存取權的風險。

更詳細的步驟是實作特殊權限存取管理 (PAM) 和暫時提升存取管理 (TEAM) 技術。PAM 的範例是，要求主體在採取特殊權限動作之前執行多重要素驗證。如需詳細資訊，請參閱 [設定受 MFA 保護的 API 存取](#)。TEAM 需要使用解決方案來管理允許主體擁有已提升存取權的核准和時間範圍。其中一種方法是暫時將主體新增至具有已提升存取權之 IAM 角色的角色信任政策中。另一種方法是在正常操作情況下，使用 [工作階段政策](#) 縮小 IAM 角色授予主體的許可範圍，然後在核准時段暫時解除此限制。若要進一步了解已經過 AWS 和精選合作夥伴驗證的解決方案，請參閱 [暫時提升的存取權](#)。

實作步驟

1. 將您的工作負載和環境隔離到個別 AWS 帳戶中。
2. 使用 SCP 減少可授予組織的成員帳戶內主體的許可集上限。
 - a. 定義 SCP 以減少可授予組織成員帳戶中主體的許可集上限時，您可以選擇允許清單或拒絕清單方法。允許清單策略會明確指定允許的存取權，並隱含地封鎖所有其他存取權。拒絕清單策略會明確指定不允許的存取權，並預設允許所有其他存取權。這兩種策略各有其優點和權衡，而適當的選擇取決於組織的特定要求和風險模型。如需詳細資訊，請參閱 [使用 SCP 的策略](#)。
 - b. 此外，請查看 [服務控制政策範例](#)，了解如何有效地建構 SCP。

3. 使用 IAM 資源政策可縮小資源上許可動作的範圍並指定條件。在 IAM 角色信任政策中使用條件來建立擔任角色的限制。
4. 將 IAM 許可界限指派至 IAM 角色，之後工作負載團隊可以使用這些角色來管理自己的工作負載 IAM 角色和許可。
5. 根據您的需求評估 PAM 和 TEAM 解決方案。

資源

相關文件：

- [AWS 上的資料周邊](#)
- [使用資料周邊建立許可防護機制](#)
- [Policy 評估邏輯](#)

相關範例：

- [服務控制政策範例](#)

相關工具：

- [AWS 解決方案：暫時提升存取管理](#)
- [適用 TEAM 的已驗證安全合作夥伴解決方案](#)

SEC03-BP06 根據生命週期管理存取

監控並調整授予您的主體 (使用者、角色和群組) 在組織內其整個生命週期的許可。隨著使用者變更角色調整群組成員資格，並在使用者離開組織時移除存取權。

預期成果您可在組織內監控並調整主體整個生命週期的許可，進而降低不必要權限帶來的風險。您可在建立使用者時授予適當的存取權。您可以隨著使用者的職責變更修改存取權，並且在使用者不再為作用中狀態或離開組織時移除存取權。您可以集中管理使用者、角色和群組的變更。您使用自動化的方式將變更傳播到您的 AWS 環境。

常見的反模式：

- 事先授予身分過多或過廣的存取權限，超過最初所需的範圍。

- 當身分的角色和職責隨時間發生變更後，未能審查並調整存取權限。
- 未移除非作用中或已終止身分的作用中存取權限。此舉會增加未經授權存取的風險。
- 未利用自動化來管理身分的生命週期。

未建立此最佳實務時的風險暴露等級：中

實作指引

在身分的整個生命週期中，仔細管理和調整您授予身分 (例如使用者、角色、群組) 的存取權限。此生命週期涵蓋初始入職階段、後續角色和職責變更，以及最終離職或終止。根據生命週期階段主動管理存取權，以維護適當的存取層級。遵守最低權限原則，以降低過度或不必要存取權限帶來的風險。

您可以直接在 AWS 帳戶內管理 IAM 使用者的生命週期，或透過從人力資源身分提供者至 [AWS IAM Identity Center](#) 的聯合來進行管理。針對 IAM 使用者，您可以在 AWS 帳戶內建立、修改和刪除使用者及其關聯的許可。針對聯合身分使用者，您可以使用 IAM Identity Center 管理生命週期，方法是透過 [跨網域身分管理系統 \(SCIM\)](#) 協定，同步源自您組織身分提供者的使用者和群組資訊。

SCIM 是開放標準協定，可在不同系統中自動佈建和解除佈建使用者身分。透過將身分提供者與使用 SCIM 的 IAM Identity Center 整合，您就可以自動同步使用者和群組資訊，協助驗證組織是否根據其授權身分來源的變更授予、修改或撤銷存取權限。

隨著組織內員工的角色和職責改變，調整他們的存取權限。您可以使用 IAM Identity Center 的許可集來定義不同的工作角色或職責，並將其與適當的 IAM 政策和許可關聯。當員工的角色變更時，您可以更新其指派的許可集，以反映他們的新職責。確認他們具有必要的存取權，同時遵守最低權限原則。

實作步驟

1. 定義並記錄存取管理生命週期流程，包括授予初始存取權、定期審查和離職的程序。
2. 實作 [IAM 角色、群組和許可界限](#)，以共同管理存取權，並強制執行受允許的最高存取層級。
3. 使用 IAM Identity Center 與 [聯合身分提供者](#) (例如 Microsoft Active Directory、Okta、Ping Identity) 整合，使其做為使用者和群組資訊的授權來源。
4. 使用 [SCIM](#) 協定可將來自身分提供者的使用者和群組資訊同步到 IAM Identity Center 的身分存放區中。
5. 在 IAM Identity Center 中建立 [許可集](#)，以代表組織內不同的工作角色或職責。為每個許可集定義適當的 IAM 政策和許可。
6. 實作定期存取權審查、提示撤銷存取權，以及持續改進存取權管理生命週期流程。

7. 為員工提供有關存取權管理最佳實務的培訓和認知。

資源

相關的最佳實務：

- [SEC02-BP04 仰賴集中式身分提供者](#)

相關文件：

- [管理您的身分來源](#)
- [管理 IAM Identity Center 中的身分](#)
- [使用 AWS Identity and Access Management Access Analyzer](#)
- [IAM Access Analyzer 政策產生](#)

相關影片：

- [AWS re:Inforce 2023 - 使用 AWS IAM Identity Center 管理暫時提升的存取權](#)
- [AWS re:Invent 2022 - 使用 IAM Identity Center 簡化現有的員工存取權](#)
- [AWS re:Invent 2022 - 使用 Access Analyzer 掌握 IAM 政策之力並駕馭許可](#)

SEC03-BP07 分析公有和跨帳戶存取權

持續監控強調公有和跨帳戶存取權的調查結果。減少僅對需要此存取之特定資源的公有存取權和跨帳戶存取權。

預期成果：知道您共用了哪些 AWS 資源以及共用的對象。持續監控和稽核您共用的資源以確認這些資源僅與已授權主體共用。

常見的反模式：

- 沒有維持共用資源的詳細目錄。
- 未遵循程序來核准跨帳戶或資源的公有存取權。

未建立此最佳實務時的曝險等級：低

實作指引

如果您的帳戶位於 AWS Organizations 中，您可以將資源的存取權授予整個組織、特定組織單位或個別帳戶。如果您的帳戶不是組織的成員，您可以與個別帳戶共用資源。您可以使用以資源為基礎的政策 (例如 [Amazon Simple Storage Service \(Amazon S3\) 儲存貯體政策](#)) 來授予直接跨帳戶存取權，或允許另一個帳戶中的主體擔任您帳戶中的 IAM 角色。當使用資源政策時，確認僅將該存取權授予已授權的主體。定義程序，來核准所有需要公開提供的資源。

[AWS Identity and Access Management Access Analyzer](#) 採用 [可證明的安全性](#) 來識別從其帳戶外部存取資源的所有路徑。它會持續審查資源政策，並報告公有和跨帳戶存取權的調查結果，讓您輕鬆分析潛在的各種存取。考慮使用 AWS Organizations 設定 IAM Access Analyzer，以確認您對所有帳戶具有可見性。IAM Access Analyzer 也允許您在部署資源許可之前 [預覽調查結果](#)。這可讓您驗證政策變更是否僅授予對您資源的預期公有和跨帳戶存取權。設計多帳戶存取權時，您可以使用 [信任政策](#) 來控制可以擔任角色的情況。例如，您可以使用 [PrincipalOrgId 條件金鑰來拒絕嘗試從 AWS Organizations 外部擔任角色的動作](#)。

[AWS Config](#) 可以報告設定不當的資源，並可透過 AWS Config 政策檢查來偵測已設定公開存取的資源。諸如 [AWS Control Tower](#) 和 [AWS Security Hub](#) 的服務可簡化在 AWS Organizations 間部署控制和防護機制的作業，以識別和修正公開暴露的資源。例如，AWS Control Tower 具備受管的防護機制，可偵測是否有任何 [可由 AWS 帳戶 還原的 Amazon EBS 快照](#)。

實作步驟

- 考慮對 [AWS Organizations 使用 AWS Config](#)：AWS Config 可讓您將 AWS Organizations 內來自多個帳戶的調查結果彙總到一個委派的管理員帳戶。這提供了全面性檢視，並讓您 [在帳戶間部署 AWS Config 規則](#) 以偵測公開可存取的資源。
- 設定 AWS Identity and Access Management Access Analyzer：IAM Access Analyzer 可協助您識別組織及帳戶中 [與外部實體共用的資源](#)，例如 Amazon S3 儲存貯體或 IAM 角色。
- 使用 AWS Config 中的自動修復以回應 Amazon S3 儲存貯體的公開存取設定中的變更：[您可以自動開啟 Amazon S3 儲存貯體的封鎖公開存取設定](#)。
- 實作監控和提醒以識別 Amazon S3 儲存貯體是否已變為公有：您必須設立 [監控和提醒](#) 以識別何時關閉 Amazon S3 封鎖公開存取，以及 Amazon S3 儲存貯體是否變為公有。此外，如果您正在使用 AWS Organizations，可以建立 [服務控制政策](#) 來防止對 Amazon S3 公開存取政策進行變更。[AWS Trusted Advisor](#) 會檢查具有公開存取許可的 Amazon S3 儲存貯體。將上傳或刪除存取權授予每個人的儲存貯體許可，可讓任何人在儲存貯體中新增、修改或移除項目，進而產生潛在的安全問題。Trusted Advisor 檢查會分析明確的儲存貯體許可，以及可能覆寫儲存貯體許可的關聯儲存貯體政策。您也可以使用 AWS Config 來監控 Amazon S3 儲存貯體的公開存取。如需詳細資訊，請參閱 [如何使用 AWS Config 來監控及回應允許公開存取的 Amazon S3 儲存貯體](#)。

檢閱 Amazon S3 儲存貯體的存取控制時，請務必考量儲存於其中的資料性質。[Amazon Macie](#) 是一項服務，其設計在於協助您探索和保護敏感資料，例如個人身分識別資訊 (PII)、受保護健康資訊 (PHI)，以及私有金鑰或 AWS 存取金鑰等憑證。

資源

相關文件：

- [使用 AWS Identity and Access Management Access Analyzer](#)
- [AWS Control Tower 控制程式庫](#)
- [AWS 基礎安全最佳實務標準](#)
- [AWS Config 受管規則](#)
- [AWS Trusted Advisor 檢查參考](#)
- [使用 Amazon EventBridge 監控 AWS Trusted Advisor 檢查結果](#)
- [管理組織內所有帳戶間的 AWS Config 規則](#)
- [AWS Config 和 AWS Organizations](#)
- [在 Amazon EC2 中公開發提供您的 AMI](#)

相關影片：

- [保護多帳戶環境的最佳實務](#)
- [深入了解 IAM Access Analyzer](#)

SEC03-BP08 在組織內安全地共用資源

隨著工作負載數量增加，您可能需要在這些工作負載內共用對資源的存取，或在多個帳戶間多次佈建資源。您可能具備劃分環境 (例如擁有開發、測試和生產環境) 的建構模組。然而，擁有分隔建構模組並不會限制您安全共用的能力。透過共用重疊的元件，您可以降低營運負擔並允許一致的體驗，而不用猜測在多次建立相同的資源時可能錯過了什麼。

預期成果：使用安全方法在組織內共用資源，藉此充分減少意外存取，並協助您的資料外洩防護計畫。減輕與管理個別元件相較下的營運負擔，減少多次手動建立相同元件的錯誤，以及增加工作負載的可擴展性。您可以從多點失敗案例中更短的解決時間獲益，並更有信心確定何時不再需要某元件。如需有關分析外部共用的資源的方案指引，請參閱 [SEC03-BP07 分析公有和跨帳戶存取權](#)。

常見的反模式：

- 缺乏可持續監控和自動發出意外外部共用通知的程序。
- 對於應該和不應該共用的內容缺乏基準。
- 預設採用廣泛的開放政策而不是在必要時明確共用。
- 必要時手動建立重疊的基礎資源。

未建立此最佳實務時的曝險等級：中

實作指引

建構您的存取控制和模式來管控安全地取用共用資源並只與信任的實體共用。監控共用資源並持續審查共用資源存取，在不當或意外共用時獲得提醒。審核[分析公開和跨帳戶存取權](#)協助您確立管控能力以減少外部存取，而僅限於需要存取的資源，以及建立程序持續監控並自動提供提醒。

在 AWS Organizations 內跨帳戶共用受到數個 AWS 服務的支援，例如 [AWS Security Hub](#)、[Amazon GuardDuty](#) 和 [AWS Backup](#)。這些服務允許將資料共用到中央帳戶，從中央帳戶存取，或從中央帳戶管理資源和資料。例如，AWS Security Hub 可以將調查結果從個別帳戶轉移到中央帳戶，讓您能夠檢視所有調查結果。AWS Backup 可以對資源進行備份並在帳戶之間共用。您可以使用 [AWS Resource Access Manager](#) (AWS RAM) 來共用其他常見的資源，例如 [VPC 子網路](#)和 [Transit Gateway 附件](#)、[AWS Network Firewall](#) 或 [Amazon SageMaker AI 管道](#)。

若要將您的帳戶限制為僅共用組織內的資源，請使用[服務控制政策 \(SCP\)](#) 防止存取外部主體。當共用資源時，結合身分型控制和網路控制為您的組織建立資料周邊，以協助預防意外存取。資料周邊是一組預防性防護機制，可協助確認只有可信的身分從預期網路存取可信的資源。這些控制應適當限制可以共用哪些資源，並防止共用或公開不應該允許的資源。例如，作為資料周邊的一部分，您可以使用 VPC 端點政策和 `AWS:PrincipalOrgId` 條件來確保存取 Amazon S3 儲存貯體的身分屬於您的組織。請務必注意，[SCP 不適用於連結服務的角色或 AWS 服務主體](#)。

當使用 Amazon S3 時，請[關閉 Amazon S3 儲存貯體的 ACL](#) 並使用 IAM 政策來定義存取控制。若要限制從 [Amazon CloudFront](#) 對 Amazon S3 原點的存取，請從原始存取身分 (OAI) 遷移至原始存取控制 (OAC)，後者支援額外的功能，包括使用 [AWS Key Management Service](#) 的伺服器端加密。

在某些情況下，您可能會想要允許在組織外部共用資源或將資源的存取權授予第三方。如需有關管理許可以在外部共用資源的方案指引，請參閱[許可管理](#)。

實作步驟

1. 使用 AWS Organizations：AWS Organizations 是一項帳戶管理服務，可讓您將多個 AWS 帳戶整合到您所建立和集中管理的組織中。您可以將帳戶編組成組織單位 (OU) 並將不同的政策連接至各個

- OU，以協助滿足您的預算、安全和合規需求。您也可以控制 AWS 人工智慧 (AI) 和機器學習 (ML) 服務收集和儲存資料的方式，並使用與 Organizations 整合的 AWS 服務的多帳戶管理功能。
2. 將 AWS Organizations 與 AWS 服務整合：當您啟用 AWS 服務代表您在組織的成員帳戶中執行任務時，AWS Organizations 會在每個成員帳戶中為該服務建立 IAM 服務連結角色 (SLR)。您應該使用 AWS Management Console、AWS API 或 AWS CLI 來管理可信存取。如需有關開啟可信存取的方案指引，請參閱[使用 AWS Organizations 與其他 AWS 服務](#)以及[您可以搭配 Organizations 使用的 AWS 服務](#)。
 3. 建立資料周邊：資料周邊提供明確的信任和所有權界限。在 AWS 上，它通常表示為 AWS Organizations 所管理的組織 AWS，以及存取您的 AWS 資源的任何內部部署網路或系統。資料周邊的目標是要確認若身分可信、資源可信且是預期的網路，則允許存取。不過，建立資料周邊並非一體適用的方法。請根據您的特定安全風險模型和須求，評估和採用在[AWS 上建立周邊白皮書](#)中所述的控制目標。您應仔細考慮您獨特的風險狀態，並實作符合您安全需求的周邊控制。
 4. 使用 AWS 服務中的資源共用並適當限制：許多 AWS 服務都可讓您與另一個帳戶共用資源，或鎖定另一個帳戶中的資源，例如 [Amazon Machine Image \(AMI\)](#) 和 [AWS Resource Access Manager \(AWS RAM\)](#)。限制 ModifyImageAttribute API 以指定可信帳戶來共用 AMI。當使用 AWS RAM 來限制僅共用至您的組織時，指定 ram:RequestedAllowsExternalPrincipals 條件來協助防止不受信任的身分的存取。如需方案指引和考量，請參閱[資源共用和外部目標](#)。
 5. 使用 AWS RAM 在帳戶中或與其他 AWS 帳戶 安全地共用：[AWS RAM](#) 可協助您安全地將您使用帳戶中的角色和使用者所建立的資源與其他 AWS 帳戶 共用。在多帳戶環境中，AWS RAM 可讓您建立資源一次並與其他帳戶 共用。這個方法有助於降低營運負擔，同時透過與 Amazon CloudWatch 和 AWS CloudTrail 的整合提供一致性、可見性和可稽核性，這是在使用跨帳戶存取權時所沒有的。

如果您擁有之前使用以資源為基礎的政策共用的資源，可以使用

[PromoteResourceShareCreatedFromPolicy API](#) 或同等項目將資源共用升級到完整 AWS RAM 資源共用。

在某些情況下，您可能需要採取額外步驟來共用資源。例如，若要共用加密快照，您需要[共用 AWS KMS 金鑰](#)。

資源

相關的最佳實務：

- [SEC03-BP07 分析公有和跨帳戶存取權](#)
- [SEC03-BP09 安全地與第三方共用資源](#)
- [SEC05-BP01 建立網路層](#)

相關文件：

- [儲存貯體擁有者將跨帳戶許可授予非其擁有的物件](#)
- [如何使用信任政策搭配 IAM](#)
- [在 AWS 上建置資料周邊](#)
- [向第三方授予對 AWS 資源的存取權限時如何使用外部 ID](#)
- [可與 AWS Organizations 搭配使用的 AWS 服務](#)
- [在 AWS 上建立資料周邊：僅允許可信身分存取公司資料](#)

相關影片：

- [使用 AWS Resource Access Manager 進行精密的存取](#)
- [使用 VPC 端點確保資料周邊的安全](#)
- [在 AWS 上建立資料周邊](#)

相關工具：

- [資料周邊政策範例](#)

SEC03-BP09 安全地與第三方共用資源

您雲端環境的安全並不止於您的組織。您的組織可能仰賴第三方來管理您的部分資料。針對第三方管理的系統的許可管理應該遵循即時存取的做法，採用最低權限的原則搭配臨時憑證。透過與第三方密切合作，您可以同時減少影響範圍以及意外存取的風險。

預期成果：您會避免使用長期 AWS Identity and Access Management (IAM) 憑證，例如存取金鑰和私密金鑰，因為這些金鑰遭到濫用的情況下，會帶來安全風險。您改用 IAM 角色和臨時憑證來改善您的安全狀態，並將管理長期憑證的營運負擔降至最低。授予第三方存取權時，您使用通用唯一識別碼 (UUID) 做為 IAM 信任政策中的外部 ID，並持續掌控與角色連接的 IAM 政策，以確保最低權限存取。如需有關分析外部共用資源的方案指引，請參閱 [SEC03-BP07 分析公開和跨帳戶存取權](#)。

常見的反模式：

- 無條件地使用預設 IAM 信任政策。
- 使用長期 IAM 憑證和存取金鑰。
- 重複使用外部 ID。

未建立此最佳實務時的曝險等級：中

實作指引

您可能會想要允許在 AWS Organizations 之外共用資源或將帳戶存取權授予第三方。例如，第三方可能提供監控解決方案，而該解決方案需要存取您帳戶中的資源。在這些情況下，僅以第三方需要的權限來建立 IAM 跨帳戶角色。此外，請使用[外部 ID 條件](#)定義信任政策。當使用外部 ID 時，您或第三方可以為每個客戶、第三方或租用戶產生唯一 ID。在建立唯一 ID 後，其不應該受除了您之外的任何人控制。第三方必須實作程序，以安全、可稽核且可重新產生的方式將外部 ID 與客戶關聯。

您還可以使用 [IAM Roles Anywhere](#) 為 AWS 之外使用 AWS API 的應用程式管理 IAM 角色。

如果第三方不再需要存取您的環境，請移除該角色。避免為第三方提供長期憑證。持續關注其他支援共用的 AWS 服務，例如，允許與其他 AWS 帳戶 [共用工作負載](#) 的 AWS Well-Architected Tool，以及可協助您與其他帳戶安全共用您所擁有 AWS 資源的 [AWS Resource Access Manager](#)。

實作步驟

1. 使用跨帳戶角色提供存取權給外部帳戶。[跨帳戶角色](#)可減少外部帳戶和第三方為了服務客戶所儲存的敏感資訊量。跨帳戶角色允許您在帳戶中將 AWS 資源的存取權安全地授予第三方，例如 AWS 合作夥伴或組織內的其他帳戶，同時維持管理和稽核該存取權的能力。第三方可能從混合式基礎設施為您提供服務，或將資料提取至異地。[IAM Roles Anywhere](#) 可協助您讓第三方工作負載安全地與您的 AWS 工作負載進行互動，並進一步減少使用長期憑證的需要。

您不應該使用與使用者關聯的長期憑證或存取金鑰來提供外部帳戶存取權。反而應該使用跨帳戶角色來提供跨帳戶存取權。

2. 執行盡職調查並確保第三方 SaaS 提供者的安全存取。與第三方 SaaS 提供者共用資源時，請執行徹底的盡職調查，以確保他們採取安全且負責任的方法來存取您的 AWS 資源。評估他們的共同責任模型，以了解他們提供哪些安全措施，以及您的責任範圍。確保 SaaS 提供者採取安全且可稽核的程序來存取您的資源，包括使用[外部 ID](#) 和最低權限存取原則。使用外部 ID 有助於解決[權利義務混淆問題](#)。

實作安全控制，以確保在授予存取權給第三方 SaaS 提供者時，能夠安全存取並遵守最低權限原則。這可能包括使用外部 ID、通用唯一識別碼 (UUID) 和 IAM 信任政策，這些會將存取權限於絕對必要的身分。與 SaaS 提供者密切合作，以建立安全存取機制、定期審查他們對您 AWS 資源的存取，並進行稽核以確保遵循您的安全需求。

3. 棄用客戶提供的長期憑證。棄用長期憑證並使用跨帳戶角色或 IAM Roles Anywhere。如果您必須使用長期憑證，請制定計畫以遷移至角色型存取。如需有關管理金鑰的詳細資訊，請參閱[身分管理](#)。

另外也與您的 AWS 帳戶 團隊和第三方合作建立風險緩解執行手冊。如需有關回應和緩解潛在安全事件的衝擊的方案指引，請參閱[事件回應](#)。

4. 確認設定具有方案指引且已自動化。外部 ID 不會被視為機密，但外部 ID 不能是容易猜測的值，例如電話號碼、名稱或帳戶 ID。將外部 ID 設為唯讀欄位，而使外部 ID 不能為了冒充設定的目的而遭到變更。

您或第三方可以產生外部 ID。定義程序以決定由誰負責產生 ID。無論建立外部 ID 的實體為何，第三方都要在客戶間一致地強制唯一性和格式。

您的帳戶中為跨帳戶存取權建立的政策必須遵循[最低權限原則](#)。第三方必須提供角色政策文件，或使用 AWS CloudFormation 範本或對您來說同等的自動設定機制。這可減少發生與手動政策建立相關聯之錯誤的機率，並提供可稽核的記錄。如需有關使用 AWS CloudFormation 範本來建立跨帳戶角色的詳細資訊，請參閱[跨帳戶角色](#)。

第三方應該提供自動化、可稽核的設定機制。然而，透過使用概述所需存取權的角色政策文件，您應該可自動設定角色。使用 AWS CloudFormation 範本或同等項目，您應該透過偏移偵測來監控變更，以做為稽核實務的一部份。

5. 將變更列入考量。您的帳戶結構、對第三方的需求或他們提供的服務方案可能發生變更。您應該預期變更和失敗，並透過合適的人員、程序和技术相應進行規劃。定期稽核您提供的存取層級，並實作偵測方法以在發生意外變更時通知您。監控和稽核角色和外部 ID 資料儲存的使用。您應該準備好在發生意外變更或存取模式時撤銷第三方存取權，無論是暫時或永久撤銷。另外，衡量撤銷作業的衝擊，包括執行所花的時間、牽涉的人員、成本，以及對其他資源的衝擊。

如需有關偵測方法的方案指引，請參閱[偵測最佳實務](#)。

資源

相關的最佳實務：

- [SEC02-BP02 使用臨時憑證](#)
- [SEC03-BP05 為您的組織定義許可防護機制](#)
- [SEC03-BP06 根據生命週期管理存取](#)
- [SEC03-BP07 分析公有和跨帳戶存取權](#)
- [SEC04 偵測](#)

相關文件：

- [儲存貯體擁有者將跨帳戶許可授予非其擁有的物件](#)
- [如何搭配 IAM 角色使用信任政策](#)
- [使用 IAM 角色在 AWS 帳戶 間委派存取權](#)
- [如何使用 IAM 存取另一個 AWS 帳戶 中的資源？](#)
- [IAM 中的安全最佳實務](#)
- [跨帳戶政策評估邏輯](#)
- [將 AWS 資源的存取權授予第三方時如何使用外部 ID](#)
- [從在外部帳戶中使用自訂資源建立的 AWS CloudFormation 資源收集資訊](#)
- [安全地使用外部 ID 存取其他人擁有的 AWS 帳戶](#)
- [使用 IAM Roles Anywhere 將 IAM 角色擴展到 IAM 之外的工作負載](#)

相關影片：

- [如何允許不同 AWS 帳戶 中的使用者或角色存取我的 AWS 帳戶？](#)
- [AWS re:Invent 2018：在 60 分鐘內精通 IAM 政策](#)
- [AWS 知識中心直播：IAM 最佳實務和設計決策](#)

相關範例：

- [Well-Architected 實驗室 - Lambda 跨帳戶 IAM 角色擔任 \(Level 300\)](#)
- [設定對 Amazon DynamoDB 的跨帳戶存取權](#)
- [AWS STS 網路查詢工具](#)

偵測

偵測由兩部分組成：偵測意外或不需要的組態變更，以及偵測意外行為。第一個部分可以發生在應用程式交付生命週期中的多個位置。使用基礎設施即程式碼 (例如，CloudFormation 範本)，您可以透過在 CI/CD 管道或來源控制中實作檢查，在部署工作負載之前檢查是否有不需要的組態。然後，將工作負載部署到非生產和生產環境時，您可以使用原生 AWS、開放原始碼或 AWS 合作夥伴工具檢查組態。這些檢查可以針對不符合安全原則或最佳實務的組態，或者針對在所測試組態和所部署組態之間進行的變更。對於執行中應用程式，您可以檢查組態是否以意外方式變更，包括在已知部署或自動擴展事件之外。

對於偵測的第二部分 (意外行為)，您可以使用工具，或在特定類型的 API 呼叫增加時發出提醒。使用 Amazon GuardDuty，您可以在您的 AWS 帳戶內發生意外且可能未經授權或惡意的活動時收到提醒。您還應該明確地監控您不期望在工作負載中使用的變異 API 呼叫，以及變更安全態勢的 API 呼叫。

偵測可讓您識別潛在的安全組態錯誤、威脅或未預期的行為。這是安全生命週期的重要部分，可用來支援品質程序、法律或合規義務，以及用於識別威脅和回應工作。有不同類型的偵測機制。例如，您可以分析工作負載的日誌，了解是否有正在被利用的漏洞。您應該定期審核與工作負載相關的偵測機制，以確保符合內部和外部的政策和要求。自動提醒和通知應根據已定義的條件，可讓您的團隊或工具進行調查。這些機制是重要的反應式因素，可以協助您的組織識別和了解異常活動的範圍。

在 AWS 中，處理偵測機制時有多種方法可用。以下幾節介紹如何使用這些方法：

最佳實務

- [SEC04-BP01 設定服務和應用程式日誌記錄](#)
- [SEC04-BP02 在標準化的位置擷取日誌、調查結果和指標](#)
- [SEC04-BP03 建立安全提醒的相互關聯並增添其豐富性](#)
- [SEC04-BP04 針對不合規資源實施補救措施](#)

SEC04-BP01 設定服務和應用程式日誌記錄

保留服務和應用程式的安全事件日誌。這是稽核、調查和操作使用案例的基礎原則，以及由管控、風險和合規 (GRC) 標準、政策和程序所推動的常見安全需求。

預期成果：組織應該能夠在需要執行內部程序或義務時 (例如安全事件回應)，以可靠且一致的方式及時從 AWS 服務和應用程式擷取安全事件日誌。考慮集中日誌以達到最佳的營運成果。

常見的反模式：

- 日誌存放太久或太早刪除。
- 每個人都能存取日誌。
- 日誌的管控和使用完全仰賴手動程序。
- 儲存每一種日誌以備不時之需。
- 只在必要時檢查日誌完整性。

建立此最佳實務的優勢：對安全事件和證據來源實作根本原因分析 (RCA) 機制，以履行管控、風險和合規義務。

未建立此最佳實務時的曝險等級：高

實作指引

根據您的需求進行安全調查或其他使用案例期間，您需要能夠審查相關日誌以記錄和了解該事件的全部範圍和時間表。產生提醒也需要日誌，以指出特定關注的動作已發生。選取、開啟、儲存和設定查詢與擷取機制和提醒至關重要。

實作步驟

- 選取並使用日誌來源。在安全調查之前，您需要擷取相關日誌以追溯的方式重新建構 AWS 帳戶中的活動。選取與您的工作負載相關的日誌來源。

日誌來源選擇條件應該根據您的業務所需的使用案例。使用 AWS CloudTrail 或 AWS Organizations 線索為每個 AWS 帳戶 建立線索，以及為其設定 Amazon S3 儲存貯體。

AWS CloudTrail 是一種日誌記錄服務，會追蹤對 AWS 帳戶的 API 呼叫以擷取 AWS 服務活動。預設開啟時，此服務會保留 90 天的管理活動，而其能使用 AWS Management Console、AWS CLI 或 AWS SDK [透過 CloudTrail 事件歷史記錄擷取](#)。如需較長的保留期間和資料事件的能見度，可[建立 CloudTrail 線索](#)並將其與 Amazon S3 儲存貯體關聯，也可以選擇與 Amazon CloudWatch 日誌群組關聯。或者，您可以建立 [CloudTrail Lake](#)，這會將 CloudTrail 日誌保留長達七年，並提供以 SQL 為基礎的查詢設施

AWS 建議使用 VPC 的客戶分別使用 [VPC 流程日誌](#)和 [Amazon Route 53 解析器查詢日誌](#)來開啟網路流量和 DNS 日誌，並將它們串流處理到 Amazon S3 儲存貯體或 CloudWatch 日誌群組。您可以為 VPC、子網路或網路介面建立 VPC 流程日誌。對於 VPC 流程日誌，您可以選擇何時何地地使用流程日誌來降低成本。

AWS CloudTrail 日誌、VPC 流程日誌和 Route 53 解析器查詢日誌是在 AWS 中支援安全調查的基本日誌記錄來源。您還可以使用 [Amazon Security Lake](#) 以 Apache Parquet 格式和 Open Cybersecurity Schema Framework (OCSF) 收集、正規化並儲存此日誌資料，此種格式隨時可供查詢。Security Lake 還支援其他 AWS 日誌以及來自第三方來源的日誌。

AWS 服務可產生基本日誌來源未擷取的日誌，例如 Elastic Load Balancing 日誌、AWS WAF 日誌、AWS Config 記錄器日誌、Amazon GuardDuty 調查結果、Amazon Elastic Kubernetes Service (Amazon EKS) 稽核日誌和 Amazon EC2 執行個體作業系統及應用程式日誌。如需記錄和監控選項的完整清單，請參閱 [AWS Security Incident Response Guide](#) 的 [附錄 A：雲端功能定義 – 日誌記錄和事件](#)。

- 每個 AWS 服務和應用程式的研究日誌記錄功能：每個 AWS 服務和應用程式都為您提供日誌儲存的選項，而其各自有自己的保留和生命週期功能。兩個最常見的日誌儲存服務是 Amazon Simple Storage Service (Amazon S3) 和 Amazon CloudWatch。如需長期保留，建議使用具成本效益和彈性生命週期功能的 Amazon S3。如果主要日誌記錄選項是 Amazon CloudWatch Logs，則您應該考慮將較不常存取的日誌封存到 Amazon S3，作為一種選項。
- 選取日誌儲存：日誌儲存的選擇通常與您使用的查詢工具、保留功能、熟悉度和成本相關。日誌儲存的主要選項是 Amazon S3 儲存貯體或 CloudWatch 日誌群組。

Amazon S3 儲存貯體提供符合成本效益、耐用的儲存方式，並且具備可選擇的生命週期政策。儲存在 Amazon S3 儲存貯體的日誌可使用 Amazon Athena 之類的服務進行查詢。

CloudWatch 日誌群組透過 CloudWatch Logs Insights 提供耐用的儲存方式和內建查詢設施。

- 識別適當的日誌保留時間：當您使用 Amazon S3 儲存貯體或 CloudWatch 日誌群組來儲存日誌時，您必須為每個日誌來源建立充分的生命週期，以最佳化儲存和擷取成本。客戶一般擁有三個月到一年的時間使日誌隨時可供查詢，並且最長可保留七年。可用性和保留時間的選擇應該配合您的安全需求與各種法令、法規和業務規定。
- 依照適當的保留和生命週期政策對每個 AWS 服務和應用程式使用日誌記錄：對於組織內的每個 AWS 服務或應用程式，尋找特定的日誌記錄組態指引：
 - [設定 AWS CloudTrail 線索](#)
 - [設定 VPC 流程日誌](#)
 - [設定 Amazon GuardDuty 調查結果匯出](#)
 - [設定 AWS Config 記錄](#)
 - [設定 AWS WAF Web ACL 流量](#)
 - [設定 AWS Network Firewall 網路流量日誌](#)
 - [設定 Elastic Load Balancing 存取日誌](#)

- [設定 Amazon Route 53 解析器查詢日誌](#)
- [設定 Amazon RDS 日誌](#)
- [設定 Amazon EKS 控制平面日誌](#)
- [為 Amazon EC2 執行個體和內部部署伺服器設定 Amazon CloudWatch 代理程式](#)
- 為日誌選取並實作查詢機制：對於日誌查詢，您可以使用 [CloudWatch Logs Insights](#) (適用於儲存在 CloudWatch 日誌群組中的資料) 以及 [Amazon Athena](#) 和 [Amazon OpenSearch Service](#) (適用於儲存在 Amazon S3 中的資料)。您還可以使用第三方查詢工具，例如安全資訊和事件管理 (SIEM) 服務。

選取日誌查詢工具的過程中應該考慮安全營運的人員、程序 and 技術層面。選取符合營運、業務和安全需求的工具，並且可供存取和長期維護。請記住，將要掃描的日誌數目維持在日誌查詢工具限制之內，以便以最佳狀態運作。因為成本或技術限制的關係，擁有多個查詢工具十分常見。

例如，您可能使用第三方安全資訊和事件管理 (SIEM) 工具對過去 90 天的資料執行查詢，但由於 SIEM 的日誌擷取成本，而使用 Athena 來執行超過 90 天的查詢。無論實作方式為何，請確認您的方法將所需的工具數量最小化以最大化營運效率，尤其是在安全事件調查期間。

- 使用日誌提供提醒：AWS 透過數種安全服務提供提醒功能：
 - [AWS Config](#) 可監控和記錄 AWS 資源組態，並讓您根據所需的組態自動評估和修復。
 - [Amazon GuardDuty](#) 是威脅偵測服務，可持續監控惡意活動和未經授權的行為，以保護 AWS 帳戶 和工作負載。GuardDuty 會擷取、彙總和分析來自如 AWS CloudTrail 管理和資料事件、DNS 日誌、VPC 流程日誌和 Amazon EKS 稽核日誌等來源的資訊。GuardDuty 會直接從 CloudTrail、VPC 流程日誌、DNS 查詢日誌和 Amazon EKS 提取獨立資料串流。您不需要管理 Amazon S3 儲存貯體或修改您收集和儲存日誌的方式。仍舊建議您保留這些日誌，供自身調查和合規用途。
 - [AWS Security Hub](#) 提供以單一位置從多個 AWS 服務和選用的第三方產品將安全提醒或調查結果加以彙總、組織和排列優先順序，為您提供安全提醒和合規狀態的全面檢視。

您還可以使用自訂提醒產生引擎，取得這些服務未涵蓋的安全提醒或與您的環境相關的特定提醒。如需有關建置這些提醒和偵測的資訊，請參閱 [《AWS 安全事件回應指南》](#) 中的偵測。

資源

相關的最佳實務：

- [SEC04-BP02 在標準化的位置擷取日誌、調查結果和指標](#)
- [SEC07-BP04 定義可擴展的資料生命週期管理](#)

- [SEC10-BP06 部署前工具](#)

相關文件：

- [AWS 安全事件回應指南](#)
- [Amazon Security Lake 入門](#)
- [入門：Amazon CloudWatch Logs](#)

相關影片：

- [AWS re:Invent 2022 - Amazon Security Lake 簡介](#)

相關範例：

- [Assisted Log Enabler for AWS](#)
- [AWS Security Hub 調查結果歷史匯出](#)

SEC04-BP02 在標準化的位置擷取日誌、調查結果和指標

安全團隊仰賴日誌和調查結果來分析可能代表未經授權活動或意外變更的事件。為了簡化此分析，您可在標準化的位置擷取安全日誌和調查結果。這樣就能提供關注的資料點來建立相互關聯，並且可簡化工具整合。

預期成果：您採用標準化方法來收集、分析和視覺化日誌資料、調查結果和指標。安全團隊能夠有效率地跨分散的系統建立相互關聯、分析和視覺化安全資料，藉此探索可能發生的安全事件並識別異常。安全資訊和事件管理 (SIEM) 系統或其他機制經整合後，即可查詢和分析日誌資料，以便及時回應、追蹤和向上呈報安全事件。

常見的反模式：

- 多個團隊各自擁有並管理日誌記錄和指標收集工作，而他們的工作方式卻與組織的日誌記錄策略不一致。
- 團隊沒有適當的存取控制可用來限制所收集資料的可見性和更改。
- 團隊未將控管安全日誌、調查結果和指標納入其資料分類政策中。
- 團隊在設定資料收集時，忽略了資料主權和本地化需求。

建立此最佳實務的優勢：擁有標準化的日誌記錄解決方案可用來收集和查詢日誌資料和事件，如此就能改善從內含的資訊中產生的洞察。為收集的日誌資料設定自動化生命週期，可降低日誌儲存所伴隨的成本。您可以根據團隊所需的資料敏感度和存取模式，為收集的日誌資訊建置精細的存取控制。您可以整合工具來建立資料的相互關聯、視覺化資料，以及從資料中產生洞察。

未建立此最佳實務時的風險暴露等級：中

實作指引

隨著組織內 AWS 的使用增加，分散的工作負載和環境數量也會增加。由於這些工作負載和環境會各自產生其內部活動的相關資料，因此在本地擷取和儲存這些資料會為安全營運方面帶來挑戰。安全團隊會使用安全資訊和事件管理 (SIEM) 系統等工具從分散的來源收集資料，並進行相互關聯、分析和回應工作流程。這需要管理一組複雜的許可來存取各種資料來源，還會在操作擷取、轉換和載入 (ETL) 程序上帶來額外的負擔。

為了克服這些挑戰，請考慮依照[使用多個帳戶整理您的 AWS 環境](#)所述，將所有相關的安全日誌資料來源彙總到日誌封存帳戶中。這包括來自您的工作負載和 AWS 服務所產生日誌的所有安全相關資料，例如 [AWS CloudTrail](#)、[AWS WAF](#)、[Elastic Load Balancing](#) 和 [Amazon Route 53](#)。在標準化的位置透過具有適當跨帳戶許可的個別 AWS 帳戶擷取此資料有幾個好處。此實務有助於防止受害的工作負載和環境內的日誌遭到竄改、可為其他工具提供單一整合點，並提供更簡化的模式來設定資料保留和生命週期。評估資料主權、合規範圍和其他法規的影響，以判斷是否需要多個安全資料儲存位置和保留期。

為了輕鬆擷取和標準化日誌和調查結果，請在您的日誌封存帳戶中評估 [Amazon Security Lake](#)。您可以將 Security Lake 設定為自動從常見的來源擷取資料，例如 CloudTrail、Route 53、[Amazon EKS](#) 和 [VPC 流程日誌](#)。您也可以將 AWS Security Hub 設定為 Security Lake 中的資料來源，如此就能將其他 AWS 服務 (例如 [Amazon GuardDuty](#) 和 [Amazon Inspector](#)) 的調查結果與您的日誌資料相互關聯。您也可以使用第三方資料來源整合，或設定自訂資料來源。所有整合都會將您的資料標準化為 [Open Cybersecurity Schema Framework \(OCSF\)](#) 格式，並以 Parquet 檔案形式儲存在 [Amazon S3](#) 儲存貯體中，因此不需要進行 ETL 處理。

將安全資料儲存在標準化的位置可提供進階分析功能。AWS 建議您將在 AWS 環境中操作的安全分析工具部署到[安全工具](#)帳戶中，與您的日誌封存帳戶加以區隔。此方法可讓您深入實作控制，以保護日誌和日誌管理程序的完整性和可用性，有別於用於存取日誌的工具。考慮使用 [Amazon Athena](#) 等服務來執行與多個資料來源相互關聯的隨需查詢。您也可以整合視覺化工具，例如 [Amazon QuickSight](#)。採用 AI 技術的解決方案越來越普遍可得，並且能夠執行許多功能，例如將調查結果轉譯成人類可讀的摘要以及自然語言互動等。擁有標準化的資料儲存位置可供查詢，通常會更容易整合這些解決方案。

實作步驟

1. 建立日誌封存和安全工具帳戶
 - a. 使用 AWS Organizations，在安全組織單位下[建立日誌封存和安全工具帳戶](#)。如果您使用 AWS Control Tower 來管理組織，則會自動為您建立日誌封存和安全工具帳戶。視需要設定存取和管理這些帳戶的角色與許可。
2. 設定標準化的安全資料位置
 - a. 確定您用來建立標準化安全資料位置的策略。您可以透過通用資料湖架構方法、第三方資料產品或 [Amazon Security Lake](#) 等選項達成此目的。AWS 建議您從為帳戶設為[選擇加入](#)的 AWS 區域擷取安全資料 (即使沒有在作用中)。
3. 設定將資料來源發佈到標準化位置
 - a. 識別安全資料的來源，並將它們設定為發佈到您的標準化位置。評估以所需格式自動匯出資料的選項，而不是需要開發 ETL 程序的選項。透過 Amazon Security Lake，您可以從受支援的 AWS 來源和經過整合的第三方系統[收集資料](#)。
4. 設定工具以存取標準化位置
 - a. 設定 Amazon Athena、Amazon QuickSight 或第三方解決方案等工具，使其具備存取您的標準化位置所需的權限。設定這些工具，以適時透過對日誌封存帳戶的跨帳戶讀取存取權在安全工具帳戶之外操作。[在 Amazon Security Lake 中建立訂閱者](#)，以便為這些工具提供對您資料的存取權。

資源

相關的最佳實務：

- [SEC01-BP01 使用帳戶區隔工作負載](#)
- [SEC07-BP04 定義資料生命週期管理](#)
- [SEC08-BP04 強制存取控制](#)
- [OPS08-BP02 分析工作負載日誌](#)

相關文件：

- [AWS 白皮書：使用多個帳戶整理您的 AWS 環境](#)
- [AWS 方案指引：AWS 安全性參考架構 \(AWS SRA\)](#)
- [AWS 方案指引：應用程式擁有者的日誌記錄和監控指南](#)

相關範例：

- [使用 Amazon Athena 和 Amazon QuickSight 彙總、搜尋和視覺化來自分散來源的日誌資料](#)
- [如何使用 Amazon QuickSight 視覺化 Amazon Security Lake 調查結果](#)
- [使用 Amazon SageMaker AI Studio 和 Amazon Bedrock 為 Amazon Security Lake 產生 AI 技術支援的洞察](#)
- [使用 Amazon SageMaker AI 識別 Amazon Security Lake 資料中的網路安全異常](#)
- [擷取和轉換 Amazon Security Lake 發布的事件並傳遞至 Amazon OpenSearch Service](#)
- [使用 CloudTrail Lake 中的自然語言查詢生成功能來簡化 AWS CloudTrail 日誌分析](#)

相關工具：

- [Amazon Security Lake](#)
- [Amazon Security Lake 合作夥伴整合](#)
- [Open Cybersecurity Schema Framework \(OCSF\)](#)
- [Amazon Athena](#)
- [Amazon QuickSight](#)
- [Amazon Bedrock](#)

SEC04-BP03 建立安全提醒的相互關聯並增添其豐富性

非預期的活動可能會導致不同來源產生多個安全提醒，因此需要進一步在建立這些來源之間的相互關聯並增添豐富性，才能了解完整的內容。實作安全提醒的自動化相互關聯並增添其豐富性，有助於更準確地識別和回應事件。

預期成果：當活動在您的工作負載和環境內產生不同的提醒時，自動化機制會建立資料的相互關聯，並使用其他資訊增添該資料的豐富性。此預先處理程序呈現了對事件更詳細的了解，進而協助調查人員判斷事件的關鍵性，以及它是否構成需要正式回應的事件。此程序可減輕監控和調查團隊的負擔。

常見的反模式：

- 不同組合的人員對不同系統產生的調查結果和提醒進行調查 (除非是在因職責區分需求而另有規定的情況下)。
- 您的組織將所有安全調查結果和提醒資料收集到標準位置，但要求調查人員手動建立相互關聯和添加資訊。

- 您只仰賴威脅偵測系統的情報來回報調查結果和確定關鍵性。

建立此最佳實務的優勢：自動建立提醒的相互關聯和增添其豐富性，有助於減輕調查人員的整體認知負擔和手動準備資料的負荷。這種做法可縮短判斷事件是否為「事件」及正式回應所需的時間。額外的內容還可協助您準確評估事件的嚴重性，因為實際的嚴重性可能高於或低於任何提醒表明的嚴重性。

未建立此最佳實務時的曝險等級：低

實作指引

安全提醒可能來自 AWS 內多個不同的來源，包括：

- [Amazon GuardDuty](#)、[AWS Security Hub](#)、[Amazon Macie](#)、[Amazon Inspector](#)、[AWS Config](#)、[AWS Identity and Access Management Access Analyzer](#) 和 [網路存取分析器](#) 等服務
- 來自 AWS 服務、基礎設施和應用程式日誌的自動化分析的提醒，例如來自 [Security Analytics for Amazon OpenSearch Service](#)。
- 回應帳單活動變更的警示，來自下列來源：[Amazon CloudWatch](#)、[Amazon EventBridge](#) 或 [AWS Budgets](#)。
- 第三方來源，例如來自 AWS Partner Network 的威脅情報摘要和 [安全合作夥伴解決方案](#)
- [來自 AWS 信任與安全或其他來源的聯絡資訊](#)，例如客戶或內部員工。

提醒基本上包含有關誰 (主體或身分)、做了什麼 (採取的動作)，以及對象是誰 (受影響的資源) 的資訊。對於其中每個來源，請確定是否有能夠在這些身分、動作和資源的識別碼之間建立映射的方式，以做為建立相互關聯的基礎。可能的形式包括整合提醒來源與安全資訊和事件管理 (SIEM) 工具，以便為您自動建立相互關聯、建置您自己的資料管道和處理流程，或結合上述兩者。

可為您建立相互關聯的服務範例為 [Amazon Detective](#)。偵測會持續接收來自各種 AWS 和第三方來源的提醒，並使用不同形式的情報來構成視覺化圖形以呈現其關係，進而協助調查。

雖然提醒的初始關鍵性可協助判斷優先順序，但提醒發生的環境則決定了其真實的關鍵性。舉例來說，[Amazon GuardDuty](#) 可以針對您工作負載內的 Amazon EC2 執行個體正在查詢非預期的網域名稱而發出警示。GuardDuty 可能會自行對此提醒指派「低關鍵性」標籤。然而，如果在提醒發出之時與其他活動自動建立相互關聯，您就可能發現有數百個 EC2 執行個體是由相同身分部署的，這種情況會增加整體營運成本。在此事件中，此相互關聯事件內容會發布新的安全提醒，且關鍵性可能會調整為高，進而加速採取進一步行動。

實作步驟

1. 識別安全提醒資訊的來源。了解來自這些系統的提醒如何表示身分、動作和資源，以確定可能的相互關聯。
2. 建立一個機制來擷取不同來源的提醒。考慮用於這類用途的服務，例如 Security Hub、EventBridge 和 CloudWatch。
3. 識別資料相互關聯和增添豐富性的來源。範例來源包括 [AWS CloudTrail](#)、[VPC 流程日誌](#)、[Route 53 Resolver 日誌](#)，以及基礎設施和應用程式日誌。這些日誌的任一種或全部都可能透過與 [Amazon Security Lake](#) 的單一整合來取用。
4. 將提醒與資料相互關聯和增添豐富性的來源整合在一起，以建立更詳細的安全事件內容並構成關鍵性。
 - a. Amazon Detective、SIEM 工具或其他第三方解決方案可以自動執行特定層級的擷取、相互關聯和豐富性。
 - b. 您也可以使用 AWS 服務來自行建置。例如，您可以調用 AWS Lambda 函數來對 AWS CloudTrail 或 Amazon Security Lake 執行 Amazon Athena 查詢，並將結果發布至 EventBridge。

資源

相關的最佳實務：

- [SEC10-BP03 準備鑑識功能](#)
- [OPS08-BP04 建立可付諸行動的提醒](#)
- [REL06-BP03 傳送通知 \(即時處理和警示\)](#)

相關文件：

- [AWS 安全事件回應指南](#)

相關範例：

- [如何使用帳戶中繼資料增添 AWS Security Hub 調查結果的豐富性](#)

相關工具：

- [Amazon Detective](#)
- [Amazon EventBridge](#)
- [AWS Lambda](#)
- [Amazon Athena](#)

SEC04-BP04 針對不合規資源實施補救措施

您的偵測控制可能針對不符合您組態需求的資源發出提醒。您可以手動或自動實施以程式設計方式定義的補救措施，以修正這些資源並協助盡可能將影響降到最低。以程式設計方式定義補救措施時，您可以採取快速且一致的行動。

雖然自動化可以增強安全操作，但您應謹慎實作和管理自動化程序。設置適當的監督和控制機制，以確認自動化回應是否有效、準確，且合乎組織政策和風險偏好。

預期成果：您會定義資源組態標準，以及在偵測到資源不合規時的修復步驟。在可能的情況下，您已透過程式設計方式定義補救措施，以供人員手動或透過自動化方式啟動。已設置偵測系統，其可識別不合規的資源，並在由安全人員監控的集中式工具中發布提醒。這些工具支援手動或自動執行程式設計的補救措施。自動化補救措施設有適當的監督和控制機制來控管理其使用。

常見的反模式：

- 您實作自動化，但未徹底測試和驗證補救動作。這可能會導致意外的後果，例如中斷正當的業務營運或導致系統不穩定。
- 您透過自動化改善回應時間和程序，但未設置適當的監控與機制，無法在需要時允許人為介入和判斷。
- 您只仰賴補救措施，而不是將補救措施視為更廣泛的事件回應和復原計畫的一部分。

建立此最佳實務的優勢：自動化補救措施能夠比手動流程更快回應組態錯誤，進而有助於將可能對業務造成的影響降至最低，並且減少意外使用的機會。當您以程式設計方式定義補救措施時，就能一致套用這些措施，進而降低人為錯誤的風險。自動化還可同時處理更大量的提醒，這點對於大規模操作的環境來說尤其重要。

未建立此最佳實務時的曝險等級：中

實作指引

如 [SEC01-BP03 識別和驗證控制目標](#) 中所述，[AWS Config](#) 和 [AWS Security Hub](#) 等服務可協助您監控帳戶中資源的組態，以確保符合您的需求。偵測到不合規的資源時，AWS Security Hub 這類服務可

協助適當路由警示和修復。這些解決方案為您的安全調查人員提供了一個集中的位置，方便監控問題並採取修正動作。

有些不合規資源的情況獨特，需要人為判斷來進行修復，有些情況則有標準回應，您可透過程式設計方式定義這類回應。例如，對於設定錯誤的 VPC 安全群組，其標準回應可能是移除不允許的規則並通知擁有者。您可以在 [AWS Lambda](#) 函數中、[AWS Systems Manager Automation](#) 文件中，或透過您慣用的其他程式碼環境來定義回應。確定環境能夠使用具有採取修正動作所需之最低許可權的 IAM 角色來對 AWS 進行身分驗證。

定義所需的補救措施後，您就可以決定您偏好的補救措施實施方法。AWS Config 可以為您**實施補救措施**。如果您使用 Security Hub，則可透過**自訂動作**來執行此操作，該動作會將調查結果資訊發布至 [Amazon EventBridge](#)。然後，EventBridge 規則就能實施補救措施。您可以透過 Security Hub 設定自動或手動執行補救措施。

對於程式化的補救措施，建議您留存所執行動作的完整日誌和稽核，以及其結果。檢閱並分析這些日誌，以評估自動化流程的有效性，並識別改進之處。擷取 [Amazon CloudWatch Logs](#) 中的日誌，以及 Security Hub 中做為**調查結果備註**的修復結果。

一開始，請考慮 [AWS 上的自動化安全性回應](#)，其中包含預先建置的補救措施，可用來解決常見的安全錯誤組態。

實作步驟

1. 分析提醒並排定優先順序。
 - a. 將來自各種不同 AWS 服務的安全提醒合併到 Security Hub 中，以提供集中查看、排定優先順序和修復的方式。
2. 制定補救措施。
 - a. 使用 Systems Manager 和 AWS Lambda 等服務來執行程式化的補救措施。
3. 設定實施補救措施的方式。
 - a. 使用 Systems Manager 定義將調查結果發布到 EventBridge 的自訂動作。設定手動或自動啟動這些動作。
 - b. 您也可以使用 [Amazon Simple Notification Service \(SNS\)](#) 傳送通知和提醒給相關的利害關係人 (例如，安全團隊或事件回應團隊)，以便在必要時進行人為介入或向上呈報。
4. 檢閱並分析補救措施日誌，以了解有效性和改進之處。
 - a. 將日誌輸出傳送至 CloudWatch Logs。擷取 Security Hub 中做為調查結果備註的結果。

資源

相關的最佳實務：

- [SEC06-BP03 減少手動管理和互動式存取](#)

相關文件：

- [AWS 安全事件回應指南 - 偵測](#)

相關範例：

- [AWS 上的自動化安全回應](#)
- [使用 AWS Config 監控 EC2 執行個體金鑰對](#)
- [使用 AWS CloudFormation Guard 政策建立 AWS Config 自訂規則](#)
- [自動修復未加密的 Amazon RDS 資料庫執行個體和叢集](#)

相關工具：

- [AWS Systems Manager Automation](#)
- [AWS 上的自動化安全回應](#)

基礎設施保護

基礎設施保護包括符合最佳實務和組織或監管義務所必需的控制方法，例如深度防禦。這些方法的使用對於雲端的成功持續營運至關重要。

基礎設施保護是資訊安全計畫的關鍵部分。它可以確保工作負載中的系統和服務受到保護，以防止意外和未經授權的存取以及潛在漏洞。例如，您將定義信任邊界 (例如，網路和帳戶邊界)、系統安全組態和維護 (例如，強化、最小化和修補)、作業系統身分驗證和授權 (例如，使用者、金鑰和存取層級)，以及其他適當的政策強制執行點 (例如，Web 應用程式防火牆和/或 API 閘道)。

區域、可用區域、AWS Local Zones 和 AWS Outposts

確定您熟悉區域、可用區域、[AWS Local Zones](#) 和 [AWS Outposts](#)，這些都是 AWS 安全全球基礎設施的元件。

AWS 具有區域概念，是我們在全世界將資料中心叢集化的實體位置。我們會將邏輯資料中心的每個群組稱為可用區域 (AZ)。每個 AWS 區域由地理區域內多個隔離且實際上分開的 AZ 組成。如果您有資料落地需求，則可以選擇靠近您所需位置的 AWS 區域。您保留對資料實際所在區域的完全控制和擁有權，這有助於符合您的區域合規和資料落地需求。每個 AZ 都有獨立的電源、冷卻和實體安全性。如果應用程式跨 AZ 分割，則您可以獲得更好的隔離和保護，讓您免於停電、雷擊、龍捲風、地震等問題。AZ 與任何其他 AZ 實際上相距一段有意義的距離 (數公里)，但它們彼此的距離都在 100 公里 (60 英里) 內。AWS 區域中的所有 AZ 都是使用完全冗餘的專用都會光纖 (在 AZ 之間提供高輸送量、低延遲網路)，搭配高頻寬、低延遲聯網來互連。AZ 之間的所有流量都是加密的。專注於高可用性的 AWS 客戶可以將其應用程式設計為在多個 AZ 中執行，以實現更高的容錯。AWS 區域符合最高階的安全性、合規和資料保護。

AWS Local Zones 將運算、儲存、資料庫和其他精選 AWS 服務置於更靠近最終使用者的位置。使用 AWS Local Zones，您可以輕鬆執行要求最終使用者十毫秒延遲的高要求應用程式，例如媒體和娛樂內容創作、即時遊戲、水庫模擬、電子設計自動化和機器學習。每個 AWS Local Zone 位置都是 AWS 區域的延伸，您可以在其中執行對延遲敏感的應用程式，方法為使用地理上接近最終使用者的 Amazon EC2、Amazon VPC、Amazon EBS、Amazon File Storage 和 Elastic Load Balancing 等 AWS 服務。AWS Local Zones 會提供高頻寬、保護本機工作負載與在 AWS 區域中執行的工作負載之間的連線，進而允許您透過相同的 API 和工具集無縫連線到各種區域內服務。

AWS Outposts 可將原生 AWS 服務、基礎設施和操作模式用於幾乎所有的資料中心、主機代管空間或內部部署設施。您可以跨內部部署設施和 AWS 雲端使用相同的 AWS API、工具和基礎設施，以提供真正一致的混合體驗。AWS Outposts 專為連線環境而設計，而且可以用來支援由於低延遲或本機資料處理需求而必須保留在內部部署的工作負載。

在 AWS 中，有多種方法可用於基礎設施保護。下列幾節介紹如何使用這些方法。

主題

- [保護網路](#)
- [保護運算](#)

保護網路

您的員工和客戶中的使用者可以位於任何地方。您需要從傳統模式轉變，這些模式信任可以存取您網路的任何人和任何事物。當您遵循在所有層套用安全性的原則時，您可以使用[零信任](#)方法。零信任安全是一種模式，其中應用程式元件或微型服務被認為是彼此獨立的，並且沒有任何元件或微型服務信任彼此。

仔細規劃和管理網路設計，可奠定如何為工作負載內的資源提供隔離和邊界的基礎。由於工作負載中的許多資源在 VPC 中運作並繼承安全屬性，因此自動化支援的檢測和保護機制適用於設計非常重要。同樣地，對於在 VPC 外部操作的工作負載，使用純邊緣服務和/或無伺服器，最佳實務適用於更簡化的方法。如需無伺服器安全的特定指引，請參閱 [AWS Well-Architected 無伺服器應用程式聚焦](#)。

最佳實務

- [SEC05-BP01 建立網路層](#)
- [SEC05-BP02 控制網路層內的流量流程](#)
- [SEC05-BP03 實作以檢查為基礎的保護](#)
- [SEC05-BP04 自動化網路保護](#)

SEC05-BP01 建立網路層

以工作負載元件的邏輯分組為基礎，根據其資料敏感性和存取需求將您的網路拓樸區分成不同層。將需要從網際網路進行傳入存取的元件 (例如公用 Web 端點) 和只需要內部存取的元件 (例如資料庫) 加以區分。

預期結果：您網路的圖層是整體 defense-in-depth 安全方法的一部分，可補充工作負載的身分驗證和授權策略。根據資料敏感性和存取需求妥善區分的網路層，且具有適當的流量流程和控制機制。

常見的反模式：

- 您可以在單一 VPC 或子網路中建立所有資源。
- 您建構網路層時未考量資料敏感性需求、元件行為或功能。

- 您使用 VPCs 和子網路作為所有網路層考量的預設，且不考慮 AWS 受管服務如何影響您的拓撲。

建立此最佳實務的優勢：建立網路層是透過網路限制非必要路徑的第一步，尤其是前往關鍵系統和資料的路徑。這可讓未經授權的行為者更難存取您的網路並瀏覽至網路內的其他資源。分散的網路層有利於縮小檢測系統的分析範圍，例如針對入侵偵測或防範惡意軟體。這樣可減少誤報和不必要的處理負擔。

未建立此最佳實務時的曝險等級：高

實作指引

在設計工作負載架構時，根據元件的責任將其劃分至不同層是常見的做法。例如，Web 應用程式可擁有呈現層、應用程式層和資料層。您可以在設計網路拓撲時採用類似的方法。基礎網路控制可協助強制執行工作負載的資料存取需求。例如，在三層 Web 應用程式架構中，您可以將靜態呈現層檔案存放在 [Amazon S3](#) 上，並從內容交付網路（CDN）提供這些檔案，例如 [Amazon CloudFront](#)。應用程式層可以具有 [Application Load Balancer（ALB）](#) 在 [Amazon VPC](#) 公有子網路（類似非隔離區域或 DMZ）中提供的公有端點，並將後端服務部署到私有子網路。託管資料庫和共用檔案系統等資源的資料層，可位於與應用程式層的資源所在位置不同的私有子網路。在每個層邊界（CDN、公有子網路、私有子網路）上，您可以部署控制項，僅允許授權流量周遊這些邊界。

類似於根據工作負載元件的功能用途建立網路層模型，請一併考量要處理的資料敏感性。以 Web 應用程式為例，雖然您所有的工作負載服務可能都位於應用程式層內，但不同的服務可能會處理不同敏感程度的資料。在此情況下，根據您的控制需求，使用多個私有子網路分割應用程式層、在相同的 VPCs 中不同 AWS 帳戶，或 AWS 帳戶甚至在每種資料敏感度層級 VPCs 的不同。

對網路層的進一步考量是工作負載元件的行為一致性。繼續以上述範例說明，在應用程式層中，您可能接受來自最終使用者或外部系統整合輸入的服務，而導向這些服務的輸入在本質上比對其他服務的輸入帶有更高風險。範例包括檔案上傳、要執行的程式碼指令碼、電子郵件掃描等。將這些服務放置在其自己的網路層中，有助於在其周圍建立更強大的隔離邊界，並可防止其獨特行為造成檢測系統中的誤報提醒。

在設計過程中，請考慮使用 AWS 受管服務如何影響您的網路拓撲。探索 [Amazon VPC Lattice](#) 等服務如何協助簡化跨網路層的工作負載元件互通性。使用時 [AWS Lambda](#)，除非有特定原因，否則請在 VPC 子網路中部署。判斷 VPC 端點和的位置 [AWS PrivateLink](#)，可以簡化遵守限制網際網路閘道存取的安全政策。

實作步驟

1. 檢閱您的工作負載架構。根據元件和服務提供的功能、處理的資料敏感性以及其行為，將其邏輯分組。

2. 對於回應網際網路請求的元件，請考慮使用負載平衡器或其他代理來提供公有端點。使用 [Amazon API Gateway](#)、CloudFront、Elastic Load Balancing 和 等受管服務 [AWS Amplify](#) 來託管公有端點，以探索安全控制項的轉移。
3. 對於在運算環境中執行的元件，例如 Amazon EC2 執行個體、[AWS Fargate](#) 容器或 Lambda 函數，請將這些元件部署到私有子網路中，以第一步驟中的群組為基礎。
4. 對於 [Amazon DynamoDB](#)、[Amazon Kinesis](#) 或 [Amazon SQS](#) 等完全受管 AWS 服務，請考慮使用 VPC 端點作為透過私有 IP 地址存取的預設。

資源

相關的最佳實務：

- [REL02 規劃您的網路拓撲](#)
- [PERF04-BP01 了解聯網如何影響效能](#)

相關影片：

- [AWS re : Invent 2023 - AWS networking 基礎](#)

相關範例：

- [VPC 範例](#)
- [ECS使用 AWS Fargate、AWS PrivateLink、和 Network Load Balancer 私下存取 Amazon 上的容器應用程式](#)
- [VPC使用 Amazon 在 Amazon S3 儲存貯體中透過 提供靜態內容 CloudFront](#)

SEC05-BP02 控制網路層內的流量流程

在網路層內，使用進一步的分隔方式，將流量限於每個工作負載所需的流程。首先，專注於控制網際網路或其他外部系統到工作負載與您的環境之間的流量 (南北流量)。接著查看不同元件和系統之間的流量 (東西流量)。

預期成果：您只允許工作負載的元件所需的網路流程互相通訊，以及與其用戶端和其相依的任何其他服務進行通訊。您的設計考量到公有與私有輸入和輸出之間的比較、資料分類、區域法規以及協定需求等因素。在最低權限設計的原則中，在可能的情況下，您會偏好 point-to-point 透過網路對等進行流程。

常見的反模式：

- 您採用以周邊為基礎的網路安全方法，僅控制網路層邊界處的流量流程。
- 您假設網路層內的所有流量都經過驗證和授權。
- 您只對輸入流量或輸出流量實施控制，而不是對兩者都實施。
- 您只仰賴工作負載元件和網路控制項來驗證和授權流量。

建立此最佳實務的優勢：此實務有助於降低網路內發生未經授權行動的風險，並且為您的工作負載增加一層額外的授權。透過執行流量流程控制，您就可以限制安全事件的影響範圍，並加快偵測和回應速度。

未建立此最佳實務時的曝險等級：高

實作指引

雖然網路層有助於建立工作負載中提供類似函數、資料敏感度層級和行為的元件邊界，但您可以使用技術來進一步分割這些層中遵循最低權限原則的元件，從而建立更精細的流量控制層級。在內 AWS，網路層主要是根據 Amazon 內的 IP 地址範圍使用子網路定義 VPC。也可以使用不同的定義層 VPCs，例如依業務網域分組微服務環境。使用多個時 VPCs，請使用調節路由 [AWS Transit Gateway](#)。雖然這使用安全群組和路由表提供第 4 層（IP 地址和連接埠範圍）的流量控制，但您可以使用其他服務獲得進一步控制，例如 [AWS PrivateLink](#)、[Amazon Route 53 Resolver DNS Firewall](#)、[AWS Network Firewall](#) 和 [AWS WAF](#)。

了解和清查工作負載的資料流程和通訊需求，包括連線起始方、連接埠、通訊協定和網路層。評估可用於建立連線和傳輸資料的通訊協定，以選取符合保護需求的通訊協定（例如，HTTPS 而非 HTTP）。在網路邊界和每一層內擷取這些需求。確定這些需求後，探索僅允許必要的流量流經每個連線點的選項。一個好的起點是使用內的安全群組 VPC，因為它們可以連接到使用彈性網路介面（ENI）的資源，例如 Amazon EC2 執行個體、Amazon ECS 任務、Amazon EKS Pod 或 Amazon RDS 資料庫。與第 4 層防火牆不同的是，安全群組可以設置一項規則來依識別碼允許來自另一個安全群組的流量，藉此盡量減少群組內的資源隨時間改變所需的更新。您也可以使用安全群組的傳入和傳出規則來篩選流量。

當流量在之間移動時 VPCs，通常會使用 VPC 對等進行簡單路由，或使用 AWS Transit Gateway 進行複雜路由。使用這些方法可讓來源和目的地網路的 IP 位址範圍之間的流量更順暢。不過，如果您的工作負載只需要不同中特定元件之間的流量，VPCs 請考慮使用 point-to-point 連線 [AWS PrivateLink](#)。若要這樣做，請確定哪些服務應作為生產者，哪些服務應作為取用者。部署生產者的相容負載平衡器，相應地 PrivateLink 開啟，然後接受消費者的連線請求。然後，生產者服務會從消費者的私有 IP 地址指派 VPC，供消費者用來提出後續請求。這種方法減少了對等網路的需求。包含資料處理和負載平衡的成本，作為評估的一部分 PrivateLink。

雖然安全群組和 PrivateLink 協助控制工作負載元件之間的流程，但另一個主要考量是如何控制資源允許存取的DNS網域（如果有的話）。根據的DHCP組態VPCs，您可以為此考慮兩種不同的AWS服務。大多數客戶會使用其CIDR範圍VPCs的 +2 地址可用的預設 Route 53 Resolver DNS服務（也稱為 Amazon DNS 伺服器或 AmazonProvidedDNS）。透過此方法，您可以建立DNS防火牆規則並將其與建立關聯VPC，以決定要針對您提供的網域清單採取哪些動作。

如果您不使用 Route 53 Resolver，或是想要用網域篩選以外更深入的檢測和流程控制功能來輔助 Resolver，請考慮部署 AWS Network Firewall。此服務會使用無狀態或有狀態規則來檢測個別封包，以確定是否拒絕或允許流量。您可以採用類似的方法，使用 AWS WAF篩選前往公有端點的傳入 Web 流量。如需這些服務的進一步指引，請參閱 [SEC05-BP03 實作檢查型保護](#)。

實作步驟

1. 識別工作負載元件之間的必要資料流程。
2. 使用傳入和傳出流量的方法 defense-in-depth套用多個控制項，包括使用安全群組和路由表。
3. 使用防火牆來定義對內、外和整個的網路流量的精細控制VPCs，例如 Route 53 Resolver DNS Firewall AWS Network Firewall和 AWS WAF。考慮使用 [AWS Firewall Manager](#) 集中設定和管理整個組織的防火牆規則。

資源

相關的最佳實務：

- [REL03-BP01 選擇如何分割工作負載](#)
- [SEC09-BP02 在傳輸中強制執行加密](#)

相關文件：

- [您的安全最佳實務 VPC](#)
- [AWS 網路最佳化秘訣](#)
- [上的網路安全指南 AWS](#)
- [在中保護 VPC的傳出網路流量 AWS 雲端](#)

相關工具：

- [AWS Firewall Manager](#)

相關影片：

- [AWS Transit Gateway 許多 的參考架構 VPCs](#)
- [使用 Amazon CloudFront、 AWS WAF和 應用程式加速與保護 AWS Shield](#)
- [AWS re:Inforce 2023：防火牆和設置位置](#)

相關範例：

- [實驗室： CloudFront 適用於 Web Application](#)

SEC05-BP03 實作以檢查為基礎的保護

在網路層之間設定流量檢測點，以確保傳輸中的資料符合預期的類別和模式。分析流量流程、中繼資料和模式，以協助更有效地識別、偵測及回應事件。

預期成果：在網路層之間穿梭的流量會經過檢測和授權。允許和拒絕決策取決於明確的規則、威脅情報以及偏離基準行為的程度。流量越接近敏感資料，防護就會越嚴格。

常見的反模式：

- 僅仰賴以連接埠和通訊協定為準的防火牆規則。未善加利用情報系統。
- 根據可能隨時變更的特定目前威脅模式制訂防火牆規則。
- 僅檢測從私有子網路傳輸到公有子網路的流量，或從公有子網路傳輸到網際網路的流量。
- 沒有網路流量基準點可比較，以識別行為異常。

建立此最佳實務的優勢：檢測系統可讓您制訂智慧型規則，例如，只有在流量資料內存在特定條件時，才允許或拒絕流量。根據最新的威脅情報，隨著威脅態勢隨時間的變化，從 AWS 和合作夥伴的受管規則集中受益。這可減輕維護規則和研究入侵指標的工作負擔，進而降低誤報的可能性。

未建立此最佳實務時的曝險等級：中

實作指引

使用 AWS Network Firewall、或其他[防火牆](#)和[入侵預防系統](#)（IPS）精細控制具狀態和無狀態的網路流量 AWS Marketplace，您可以在 [Gateway Load Balancer（GWLB）](#) 後方部署。AWS Network Firewall 支援 [Suricata 相容](#) 開放原始碼IPS規格，以協助保護您的工作負載。

AWS Network Firewall 和廠商解決方案都支援GWLB不同的內嵌檢查部署模型。例如，您可以按每個VPC基準執行檢查、集中在檢查中VPC，或在混合模型中部署，其中東西流量會流經檢查，VPC而網際網路傳入會按每個檢查VPC。另一個考量是解決方案是否支援取消包裝 Transport Layer Security (TLS)，可針對任一方向啟動的流量進行深度封包檢查。如需這些組態的相關資訊和深入資訊，請參閱 [AWS Network Firewall 最佳實務指南](#)。

如果您使用的是執行 out-of-band檢查的解決方案，例如從以半透明模式操作的網路介面進行封包資料的 pcap 分析，您可以設定[VPC流量鏡像](#)。鏡像流量會計入介面的可用頻寬，並且您需支付與非鏡像流量相同的資料傳輸費用。您可以查看這些設備的虛擬版本是否可在上使用[AWS Marketplace](#)，這可能支援在後方的內嵌部署GWLB。

對於透過 HTTP型通訊協定進行交易的元件，請使用 Web 應用程式防火牆 (WAF) 保護您的應用程式免受常見威脅。[AWS WAF](#) 是一種 Web 應用程式防火牆，可讓您在傳送至 Amazon API Gateway、Amazon CloudFront AWS AppSync 或 Application Load Balancer 之前，監控並封鎖符合您可設定規則的 HTTP (S) 請求。當您評估 Web 應用程式防火牆的部署時，請考慮進行深度封包檢查，因為有些會要求您在流量檢查TLS之前終止。若要開始使用 AWS WAF，您可以[AWS 受管規則](#)搭配自己的使用，或使用現有的[合作夥伴整合](#)。

您可以使用 集中管理整個 AWS 組織中的 AWS Network Firewall AWS WAF AWS Shield Advanced、和 Amazon VPC安全群組[AWS Firewall Manager](#)。

實作步驟

1. 判斷您是否可以廣泛地範圍檢查規則，例如透過檢查 VPC，或者是否需要更精細的VPC每種方法。
2. 對於內嵌檢測解決方案：
 - a. 如果使用 AWS Network Firewall，請建立規則、防火牆政策和防火牆本身。上述這些設定完成後，您可以[將流量路由至防火牆端點](#)以啟用檢測。
 - b. 如果搭配 Gateway Load Balancer (GWLB) 使用第三方設備，請在一或多個可用區域中部署和設定您的設備。然後，建立您的 GWLB、端點服務、端點，並設定流量的路由。
3. 對於 out-of-band檢查解決方案：
 1. 在應鏡像傳入和傳出流量的介面上開啟VPC流量鏡像。您可以使用 Amazon EventBridge 規則來叫用 AWS Lambda 函數，以在建立新資源時開啟介面上的流量鏡像。將流量鏡像工作階段指向處理流量的設備前方的 Network Load Balancer。
4. 對於傳入 Web 流量解決方案：
 - a. 若要設定 AWS WAF，請先設定 Web 存取控制清單 (web ACL)。Web ACL是具有序列處理預設動作 (ALLOW 或 DENY) 的規則集合，可定義您的 WAF 如何處理流量。您可以在 Web 中建立自己的規則和群組，或使用 AWS 受管規則群組ACL。

- b. 設定 Web ACL 後，請將 Web ACL 與 AWS 資源（例如 Application Load Balancer、API Gateway REST API 或 CloudFront 分佈）建立關聯，以開始保護 Web 流量。

資源

相關文件：

- [什麼是流量鏡像？](#)
- [使用第三方安全設備實作內嵌流量檢測](#)
- [AWS Network Firewall 具有路由的範例架構](#)
- [Gateway AWS Load Balancer 和 的集中式檢查架構 AWS Transit Gateway](#)

相關範例：

- [部署 Gateway Load Balancer 的最佳實務](#)
- [TLS 加密輸出流量和 的檢查組態 AWS Network Firewall](#)

相關工具：

- [AWS Marketplace IDS/IPS](#)

SEC05-BP04 自動化網路保護

使用 DevOps 實務自動化網路保護的部署，例如基礎設施，例如程式碼（IaC 和 CI/CD 管道。這些實務可協助您透過版本控制系統追蹤網路防護措施中的變更、縮短部署變更所需的時間，並協助您偵測網路防護措施是否偏離所需的組態。

預期成果：您會使用範本定義網路防護措施，並將其遞交至版本控制系統中。當進行新的變更時，自動化管道會因此而啟動，以協調其測試和部署。設置了政策檢查和其他靜態測試，可在部署前驗證變更。您可以將變更部署到模擬環境中，以驗證控制是否如預期運作。控制得到核准後，也會自動將其部署到實際執行環境中。

常見的反模式：

- 仰賴個別工作負載團隊各自定義自己的整套網路堆疊、防護措施和自動化程序。未集中發佈網路堆疊和防護措施的標準層面，供工作負載團隊取用。

- 仰賴中央網路團隊來定義網路、防護措施和自動化的所有層面。未將網路堆疊和防護措施的工作負載特定層面委派給該工作負載的團隊。
- 集中和委派的情況在網路團隊與工作負載團隊之間達到適當的平衡，但未對 IaC 範本和 CI/CD 管道整體實施一致的測試和部署標準。未在工具中擷取所需的組態，以致無法檢查範本是否遵循規範。

建立此最佳實務的優勢：使用範本定義網路防護措施，可讓您使用版本控制系統追蹤和比較一段時間的變化。使用自動化方式測試和部署變更可建立標準化和可預測性，提高成功部署的機會，並減少重複的手動組態。

未建立此最佳實務時的曝險等級：中

實作指引

[SEC05-BP02 控制網路層內的流量](#)和[SEC05-BP03 實作檢查型保護](#)中所述的一些網路保護控制項，隨附可依據最新威脅情報自動更新的受管規則系統。保護 Web 端點的範例包括[AWS WAF 受管規則](#)和[AWS Shield Advanced 自動應用程式層DDoS緩解](#)。使用[AWS Network Firewall 受管規則群組](#)隨時掌握有關信譽不良網域清單和威脅特徵的最新資訊。

除了受管規則之外，我們建議您使用 DevOps 實務來自動部署網路資源、保護和您指定的規則。您可以在[AWS CloudFormation](#) 或您選擇的其他基礎設施即程式碼 (IaC) 工具中擷取這些定義，將它們遞交至版本控制系統，並使用 CI/CD 管道部署它們。使用此方法取得 DevOps 管理網路控制項的傳統優點，例如更可預測的版本、使用等工具進行自動化測試[AWS CloudFormation Guard](#)，以及偵測部署環境與所需組態之間的偏離。

根據您在[SEC05-BP01 建立網路層](#)中所做的決策，您可能需要建立專用VPCs於輸入、輸出和檢查流程的中央管理方法。如[AWS 安全參考架構 \(AWS SRA\)](#) 中所述，您可以在VPCs專用[網路基礎設施帳戶](#)中定義這些屬性。您可以使用類似的技術集中定義其他帳戶中工作負載VPCs使用的、其安全群組、AWS Network Firewall 部署、Route 53 Resolver 規則和DNS防火牆組態，以及其他網路資源。您可以透過[AWS Resource Access Manager](#) 將這些資源與其他帳戶共用。使用此方法，您可以簡化對網路控制的自動測試並將該控制部署到網路帳戶的程序，同時只需管理一個目的地。您可以在混合模式中，透過集中部署和共用特定控制，並將其他控制委派給個別工作負載團隊及其各自的帳戶，來執行上述操作。

實作步驟

1. 建立擁有權來規範要集中定義網路和防護措施的哪些方面，以及您的工作負載團隊可以維護哪些方面。
2. 建立環境來測試變更，並將變更部署至您的網路及其防護措施。例如，使用網路測試帳戶和網路實際執行帳戶。

3. 確定如何在版本控制系統中儲存和維護範本。將儲存中央範本的儲存庫與工作負載儲存庫分開，而工作負載範本可以儲存在專屬於該工作負載的儲存庫中。
4. 建立 CI/CD 管道以測試和部署範本。定義測試以檢查是否有組態錯誤，以及範本是否符合您公司的標準。

資源

相關的最佳實務：

- [SEC01-BP06 自動化標準安全控制的部署](#)

相關文件：

- [AWS Security Reference Architecture - 網路帳戶](#)

相關範例：

- [AWS 部署管道參考架構](#)
- [NetDevSecOps 將 AWS 網路部署現代化](#)
- [將 AWS CloudFormation 安全測試與 AWS Security Hub 和 AWS CodeBuild 報告整合](#)

相關工具：

- [AWS CloudFormation](#)
- [AWS CloudFormation Guard](#)
- [cfn_nag](#)

保護運算

運算資源包括 EC2 執行個體、容器、AWS Lambda 函數、資料庫服務、IoT 裝置等。這些運算資源類型中的每一種都需要不同的方法來保護。不過，這些運算資源類型共用您需要考慮的常用策略：深度防禦、漏洞管理、減少受攻擊面、組態和操作的自動化，以及遠距離執行動作。在本節中，您將找到有關保護關鍵服務之運算資源的一般指引。對於使用的每項 AWS 服務，請您務必檢查服務文件中的特定安全建議。

最佳實務

- [SEC06-BP01 執行漏洞管理](#)
- [SEC06-BP02 從強化的影像佈建運算](#)
- [SEC06-BP03 減少手動管理和互動式存取](#)
- [SEC06-BP04 驗證軟體完整性](#)
- [SEC06-BP05 自動化運算保護](#)

SEC06-BP01 執行漏洞管理

經常掃描和修補程式碼、相依性和基礎設施中的漏洞，以協助防禦新的威脅。

預期成果：您擁有解決方案，可持續掃描工作負載以找出軟體漏洞、潛在缺陷和意料之外的網路暴露情形。您已建立流程和程序，可根據風險評估標準來識別、排定優先順序和修復這些漏洞。此外，您已為運算執行個體實作自動修補管理。您的漏洞管理計畫已整合至您的軟體開發生命週期，並且備有解決方案可在 CI/CD 管道中掃描原始程式碼。

常見的反模式：

- 沒有漏洞管理計畫。
- 執行系統修補而不考慮嚴重性或避免風險。
- 使用已過廠商提供的結束生命週期日期的軟體。
- 在分析程式碼的安全問題之前將其部署至生產環境。

未建立此最佳實務時的曝險等級：高

實作指引

漏洞管理是維護安全可靠的雲端環境的關鍵層面。它是一整套全方位的程序，包括安全性掃描、識別問題並排定優先順序，以及修補程式操作，以解決找到的漏洞。自動化在此過程中扮演關鍵角色，因為它有助於持續掃描工作負載，以找出潛在問題和意料之外的網路暴露情形，以及提供補救措施。

[AWS 共同責任模型](#)是支援漏洞管理的基本概念。根據此模型，AWS 負責保護基本的基礎設施，包括執行 AWS 服務的硬體、軟體、網路和設備。另一方面，您負責保護與 Amazon EC2 執行個體和 Amazon S3 物件等服務相關聯的資料、安全組態和管理任務。

AWS 提供一系列服務來支援漏洞管理計畫。[Amazon Inspector](#) 會持續掃描 AWS 工作負載，以找出軟體漏洞和意料之外的網路存取，而 [AWS Systems Manager Patch Manager](#) 會協助管理所有 Amazon

EC2 執行個體的修補工作。這些服務可與 [AWS Security Hub](#) 整合，這個雲端安全狀態管理服務可自動化 AWS 安全檢查、將安全提醒集中在一起，並提供全方位的視角來了解組織的安全狀態。此外，[Amazon CodeGuru 安全工具](#) 會在開發階段使用靜態程式碼分析，以識別 Java 和 Python 應用程式中的潛在問題。

透過將漏洞管理實務納入軟體開發生命週期，您就可以在漏洞進入實際執行環境之前主動解決漏洞，從而降低安全事件的風險，並將漏洞可能造成的影響降至最低。

實作步驟

1. 了解共同責任模型：檢閱 AWS 共同責任模型，以了解您在雲端保護工作負載和資料的責任。AWS 負責保護基礎雲端基礎設施，而您負責保護您使用的應用程式、資料和服務。
2. 實作漏洞掃描：設定漏洞掃描服務 (例如 Amazon Inspector) 以自動掃描您的運算執行個體 (例如，虛擬機器、容器或無伺服器函數)，藉此找出軟體漏洞、潛在缺陷和意料之外的網路暴露情形。
3. 建立漏洞管理程序：定義流程和程序來識別漏洞、排定優先順序和修復漏洞。這可能包括設定定期漏洞掃描排程、建立風險評估條件，以及根據漏洞嚴重性定義修復時間表。
4. 設定修補程式管理：使用修補程式管理服務來自動化修補運算執行個體的程序，包括作業系統和應用程式。您可以設定服務來掃描執行個體以找出缺少的修補程式，並依照排程自動進行安裝。考慮使用 AWS Systems Manager Patch Manager 來提供此功能。
5. 設定惡意軟體防護：實作各種機制來偵測您環境中的惡意軟體。例如，您可以使用 [Amazon GuardDuty](#) 等工具來分析和偵測 EC2 和 EBS 磁碟區中的惡意軟體並發出警示。GuardDuty 也可以掃描新上傳到 Amazon S3 的物件，以找出潛藏的惡意軟體或病毒，並採取行動來隔離它們，以防它們進入下游程序。
6. 在 CI/CD 管道中整合漏洞掃描：如果您使用 CI/CD 管道進行應用程式部署，請將漏洞掃描工具整合到您的管道中。Amazon CodeGuru 安全工具和開放原始碼選項等工具可掃描您的原始程式碼、相依性和成品，以找出潛在的安全問題。
7. 設定安全監控服務：設定安全監控服務 (例如 AWS Security Hub)，以全面檢視多種雲端服務的安全狀態。監控服務應從各種不同的來源收集安全調查結果，並以標準化格式呈現，以更容易排定優先順序和修復。
8. 實作 Web 應用程式滲透測試：如果您的應用程式是 Web 應用程式，且您的組織具備必要的技能或能夠取得外部協助，請考慮實作 Web 應用程式滲透測試，以找出應用程式中的潛在漏洞。
9. 利用基礎設施即程式碼實現自動化：使用基礎設施即程式碼 (IaC) 工具 (例如 [AWS CloudFormation](#)) 將資源的部署和組態自動化，包括上述安全服務在內。此做法可協助您在多個帳戶和環境中建立更一致且標準化的資源架構。
10. 監控並持續改進：持續監控漏洞管理計畫的有效性，並視需要改進。檢閱安全調查結果、評估補救措施的有效性，並據以調整您的程序和工具。

資源

相關文件：

- [AWS Systems Manager](#)
- [AWS Lambda 的安全概觀](#)
- [Amazon CodeGuru](#)
- [透過全新的 Amazon Inspector 改進、自動化雲端工作負載的漏洞管理](#)
- [使用 Amazon Inspector 和 AWS Systems Manager 自動化 AWS 中的漏洞管理和修復 – 第 1 部分](#)

相關影片：

- [保護無伺服器容器服務的安全](#)
- [Amazon EC2 執行個體中繼資料服務的安全最佳實務](#)

SEC06-BP02 從強化的映像佈建運算

透過從強化的映像部署，就可減少意外存取執行時期環境的機會。僅從受信任的登錄檔取得執行時期相依項 (例如容器映像和應用程式庫)，並驗證其簽章。建立自己的私有登錄檔來儲存受信任的映像和程式庫，以供您的建置和部署程序使用。

預期成果：您的運算資源是從強化的基準映像佈建。您只會從受信任的登錄檔擷取外部相依項 (例如容器映像和應用程式庫)，並驗證其簽章。這些都會儲存在私有登錄檔中，以供您的建置和部署程序參考。您會定期掃描和更新映像與相依項，以協助防禦任何新發現的漏洞。

常見的反模式：

- 從受信任的登錄檔取得映像和程式庫，但未先驗證其簽章或執行漏洞掃描，即逕行使用。
- 強化映像，但未定期測試映像以確認是否有新的漏洞或更新到最新版本。
- 安裝或未移除在預期的映像生命週期內不需要的軟體套件。
- 僅仰賴修補來讓實際執行運算資源保持最新狀態。單單是修補就仍有可能導致運算資源在經過一段時間後，偏離強化的標準。修補也可能無法移除威脅行為者在安全事件期間安裝的惡意軟體。

建立此最佳實務的優勢：強化映像有助於減少您的執行時期環境中可能成為未經授權使用者或服務意外存取路徑的數目。此外還能在發生任何意外存取的情況時，縮小影響的範圍。

未建立此最佳實務時的風險暴露等級：高

實作指引

若要強化您的系統，請從作業系統、容器映像和應用程式庫的最新版本開始。套用已知問題的修補程式。移除任何不需要的應用程式、服務、裝置驅動程式、預設使用者和其他憑證，藉此盡可能縮減系統規模。採取任何其他必要的行動，例如，停用連接埠以建立只有工作負載所需資源和功能的環境。以此為基準，您就可以安裝用於監控工作負載或管理漏洞等操作所需的軟體、代理程式或其他程序。

您可以使用受信任來源提供的指南來減輕強化系統的負擔，例如[網際網路安全中心 \(CIS\)](#) 和國防資訊系統局 (DISA) [安全技術實作指南 \(STIGs\)](#)。我們建議您從 AWS 或 APN 合作夥伴發佈的 [Amazon Machine Image \(AMI\)](#) 開始，並使用 AWS [EC2 Image Builder](#) 根據適當的 CIS 和 STIG 控制項組合來自動化組態。

雖然有可用的強化映像和 EC2 映像建置器配方會套用 CIS 或 DISASTIG 建議，但您可能會發現其組態阻止軟體順利執行。在這種情況下，您可以從非強化基本映像開始，安裝軟體，然後逐步套用 CIS 控制項來測試其影響。對於防止軟體執行的任何 CIS 控制項，請測試您是否可以在 DISA 中實作更精細的強化建議。追蹤您能夠成功套用的不同 CIS 控制項和 DISASTIG 組態。使用這些選項，相應地在 Image Builder 中定義您的 EC2 映像強化配方。

對於容器化工作負載，來自 Docker 的強化影像可在 [Amazon Elastic Container Registry \(ECR\) 公有儲存庫](#) 上取得。您可以使用 EC2 Image Builder 搭配 [來強化容器映像 AMIs](#)。

與作業系統和容器映像類似，您可以透過 pip、npm、Maven 和 等工具，從公有儲存庫取得程式碼套件（或程式庫）NuGet。我們建議您藉由整合私有儲存庫（例如在 [AWS CodeArtifact](#) 內）與受信任的公有儲存庫來管理程式碼套件。此整合可以 up-to-date 為您處理擷取、儲存和保留套件。然後，您的應用程式建置程序可以使用 Software Composition Analysis (SCA)、Static Application Security Testing (SAST) 和 Dynamic Application Security Testing () 等技術，取得並測試這些套件的最新版本 DAST。

對於使用的無伺服器工作負載 AWS Lambda，可簡化使用 [Lambda 層管理套件相依性](#)。使用 Lambda 層設定一組跨不同函數共用的標準相依項，並放入獨立的封存中。您可以透過自己的建置程序來建立和維護層，為您的函數提供保留的中央方式 up-to-date。

實作步驟

- 強化作業系統。使用信任來源的基本映像作為建置強化的基礎 AMIs。使用 [EC2 Image Builder](#) 協助自訂安裝在映像上的軟體。
- 強化容器化資源。設定容器化資源以符合安全最佳實務。使用容器時，請在建置管道中實作 [ECR 映像掃描](#)，並定期針對映像儲存庫實作映像掃描，以便在 CVEs 容器中尋找。

- 搭配使用無伺服器實作時 AWS Lambda，請使用 [Lambda 層](#) 來分隔應用程式函數程式碼和共用相依程式庫。為 Lambda 設定 [程式碼簽署](#)，確保只有受信任的程式碼能夠在您的 Lambda 函數中執行。

資源

相關的最佳實務：

- [OPS05-BP05 執行修補程式管理](#)

相關影片：

- [深入探索 AWS Lambda 安全性](#)

相關範例：

- [AMI使用 EC2 Image Builder 快速建置 STIG合規](#)
- [建置更好的容器映像](#)
- [使用 Lambda 層簡化開發流程](#)
- [使用無伺服器架構開發和部署 AWS Lambda 層](#)
- [使用開放原始碼 SCA、SAST和 DAST 工具建置 end-to-end AWS DevSecOps CI/CD 管道](#)

SEC06-BP03 減少手動管理和互動式存取

盡可能使用自動化方式來執行部署、組態、維護和調查任務。在發生緊急程序的情況下或在安全 (沙盒) 環境中無法啟用自動化時，請考慮手動存取運算資源。

預期成果：程式化的指令碼和自動化文件 (執行手冊) 會擷取運算資源上獲得授權的動作。這些執行手冊會自動啟動、透過變更偵測系統啟動，或是在需要人為判斷時手動啟動。只有在無法啟用自動化的緊急情況下，才能直接存取運算資源。所有手動活動都會加以記錄並納入審查程序中，以持續改善您的自動化功能。

常見的反模式：

- 使用 SSH或 等通訊協定互動式存取 Amazon EC2執行個體RDP。
- 維護個別使用者登入，例如 /etc/passwd 或 Windows 本機使用者。
- 在多個使用者之間共用密碼或私有金鑰以存取執行個體。

- 手動安裝軟體和建立或更新組態檔案。
- 手動更新或修補軟體。
- 登入執行個體以對問題進行疑難排解。

建立此最佳實務的優勢：透過自動化方式執行步驟，有助於降低意外變更和組態錯誤伴隨的操作風險。移除使用 Secure Shell (SSH) 和遠端桌面通訊協定 (RDP) 進行互動式存取會減少對運算資源的存取範圍。這樣也消除了常見的未經授權動作路徑。在自動化文件和程式化指令碼中寫入運算資源管理任務，提供了以更精細的細節程度定義和稽核完整的授權活動範圍的機制。

未建立此最佳實務時的曝險等級：中

實作指引

登入執行個體是系統管理的傳統方法。安裝伺服器作業系統後，使用者通常會手動登入以設定系統並安裝所需的軟體。在伺服器的生命週期內，使用者可能會登入以執行軟體更新、套用修補程式、變更組態及對問題進行疑難排解。

但是，手動存取伴隨著許多風險。它需要接聽請求的伺服器，例如 SSHRDP 或服務，其可提供未經授權存取的潛在路徑。此外，它也會增加執行手動步驟時發生人為錯誤的風險。這些都可能導致工作負載事件、資料損壞或銷毀，或其他安全問題。人為存取也需要設置防護措施來防止憑證共用行為，因而產生額外的管理負擔。

若要降低這些風險，您可以實作以代理程式為基礎的遠端存取解決方案，例如 [AWS Systems Manager](#)。AWS Systems Manager Agent (SSM 代理程式) 會啟動加密頻道，因此不依賴聆聽外部啟動的請求。請考慮設定 SSM 客服人員，透過 [VPC 端點 建立此頻道](#)。

Systems Manager 可讓您精細控制與受管執行個體互動的方式。您可以定義要執行的自動化程序、誰可以執行它們，以及何時可以執行。Systems Manager 不需互動式存取執行個體，即可套用修補程式、安裝軟體及進行組態變更。Systems Manager 也可以提供遠端 shell 的存取權，並在工作階段期間記錄調用的每個命令及其輸出，以記錄日誌和 [Amazon S3](#)。會 [AWS CloudTrail](#) 記錄 Systems Manager 的調用 APIs 以供檢查。

實作步驟

1. 在 Amazon EC2 執行個體上安裝 [AWS Systems Manager 代理程式](#) (SSM 代理程式)。檢查 SSM 客服人員是否包含在基本 AMI 組態中並自動啟動。
2. 驗證與您的 EC2 執行個體設定檔相關聯的 IAM 角色是否包含 AmazonSSMManagedInstanceCore [受管 IAM 政策](#)。

3. 停用在執行個體上執行的 RDP、SSH和其他遠端存取服務。您可以執行啟動範本的使用者資料區段中設定的指令碼，或使用 EC2 Image Builder 等AMIs工具建置自訂的指令碼來達成此目的。
4. 確認適用於EC2執行個體的安全群組傳入規則不允許存取連接埠 22/tcp (SSH) 或連接埠 3389/tcp () RDP。使用 AWS Config等服務實作偵測，並對設定錯誤的安全群組發出提醒。
5. 定義適當的自動化、執行手冊，並在 Systems Manager 中執行命令。使用IAM政策來定義誰可以執行這些動作，以及允許執行這些動作的條件。在非實際執行環境中完整測試這些自動化程序。在必要時調用這些自動化程序，而非以互動方式存取執行個體。
6. 在必要時，使用 [AWS Systems Manager Session Manager](#) 提供對執行個體的互動式存取。開啟工作階段活動記錄，以在 [Amazon CloudWatch Logs](#) 或 [Amazon S3](#) 中維護稽核追蹤。

資源

相關的最佳實務：

- [REL08-BP04 使用不可變的基礎設施部署](#)

相關範例：

- [使用 AWS Systems Manager 取代SSH存取以減少管理和安全額外負荷](#)

相關工具：

- [AWS Systems Manager](#)

相關影片：

- [在 Session Manager 中控制使用者 AWS Systems Manager 對執行個體的存取](#)

SEC06-BP04 驗證軟體完整性

使用加密驗證來驗證工作負載所使用之軟體成品 (包括映像) 的完整性。以加密方式簽署您的軟體，以防範未經授權的變更在您的運算環境內執行。

預期成果：所有成品都是從受信任的來源取得。廠商網站憑證經過驗證。下載的成品已藉由簽章以加密方式驗證。自有軟體會由您的運算環境以加密方式簽署和驗證。

常見的反模式：

- 信任信譽良好的廠商網站來取得軟體成品，但忽略憑證到期通知。未先確認憑證是否有效，即逕行下載。
- 驗證廠商網站憑證，但未以加密方式驗證從這些網站下載的成品。
- 僅仰賴摘要或雜湊值來驗證軟體完整性。雜湊值確定成品的原版未經修改，但未驗證其來源。
- 即使僅在您自己的部署中使用，也未簽署自有軟體、程式碼或程式庫。

建立此最佳實務的優勢：驗證您的工作負載所依賴之成品的完整性，有助於防止惡意軟體進入您的運算環境。簽署您的軟體有助於防範運算環境中發生未經授權執行的情況。藉由簽署和驗證程式碼來保護您的軟體供應鏈。

未建立此最佳實務時的風險暴露等級：中

實作指引

作業系統映像、容器映像和程式碼成品通常在散佈時會提供完整性檢查，例如透過摘要或雜湊值。這些檢查可讓用戶端運算自有的承載雜湊值並確認其與發佈的雜湊值相同，藉此驗證完整性。雖然這些檢查有助於驗證承載未遭到竄改，但不會驗證承載來自原始出處 (其來源)。驗證來源需要使用受信任的授權機構發出的憑證來數位簽署成品。

如果您在工作負載中使用下載的軟體或成品，請檢查提供者是否提供了用於驗證數位簽章的公有金鑰。以下一些範例說明 AWS 如何提供公有金鑰，以及如何驗證我們發佈的軟體：

- [EC2 Image Builder：驗證安裝下載的 AWS TOE 簽章](#)
- [AWS Systems Manager：驗證 SSM 客服人員的簽章](#)
- [Amazon CloudWatch：驗證 CloudWatch 客服人員套件的簽章](#)

將數位簽章驗證納入您用於取得和強化影像的程序，如 [SEC06-BP02 佈建來自強化影像的運算](#) 中所述。

您可以使用 [AWS Signer](#) 協助您管理簽章驗證，以及您自有軟體和成品的程式碼簽署生命週期。[AWS Lambda](#) 和 [Amazon Elastic Container Registry](#) 兩者都提供與 Signer 的整合，可用來驗證程式碼和映像的簽章。您可以使用「資源」區段中的範例，將 Signer 納入您的持續整合和交付 (CI/CD) 管道中，以便自動驗證簽章及自動簽署自有程式碼和映像。

資源

相關文件：

- [容器的加密簽署](#)
- [使用 協助保護容器映像建置管道的最佳實務 AWS Signer](#)
- [宣佈使用 AWS Signer 和 Amazon 進行容器映像簽署 EKS](#)
- [設定的程式碼簽署 AWS Lambda](#)
- [Lambda 程式碼簽署的最佳實務和進階模式](#)
- [使用 AWS Certificate Manager Private CA AWS Key Management Service 和非對稱金鑰進程式碼簽署](#)

相關範例：

- [使用 Amazon CodeCatalyst 和 自動化 Lambda 程式碼簽署 AWS Signer](#)
- [使用 簽署和驗證OCI偽影 AWS Signer](#)

相關工具：

- [AWS Lambda](#)
- [AWS Signer](#)
- [AWS Certificate Manager](#)
- [AWS Key Management Service](#)
- [AWS CodeArtifact](#)

SEC06-BP05 自動化運算保護

自動化運算保護操作以減少人工介入的需求。使用自動化掃描偵測運算資源內的潛在問題，並透過自動化的程式化回應或機群管理操作進行修復。將自動化納入您的 CI/CD 程序，以部署具有 up-to-date 相依性的可靠工作負載。

預期成果：自動化系統會執行運算資源的所有掃描和修補工作。您可以使用自動驗證來檢查軟體映像和相依性是否來自信任來源，以及是否遭到竄改。工作負載會自動檢查 up-to-date 相依性，並簽署以在運算環境中建立可信度 AWS。偵測到不合規資源時，系統會啟動自動補救措施。

常見的反模式：

- 遵循不可變的基礎設施實務，但未備妥解決方案來因應緊急修補或取代實際執行系統。

- 使用自動化方式修復設定錯誤的資源，但未設置手動覆寫機制。可能會發生需要調整需求的情況，且您可能需要暫停自動化程序，直到完成這些變更為止。

建立此最佳實務的優勢：自動化可降低未經授權存取和使用您的運算資源的風險。它有助於防止錯誤的組態進入實際執行環境，並且在發生組態錯誤時偵測到該錯誤並加以修復。自動化還可協助偵測未經授權存取和使用運算資源的情況，進而縮短您回應的時間。如此還能進一步縮小問題的整體影響範圍。

未建立此最佳實務時的曝險等級：中

實作指引

您可以套用「安全支柱」實務中所述的自動化方式，以保護您的運算資源。[SEC06-BP01 執行漏洞管理](#)說明如何在 CI/CD 管道和 中使用 [Amazon Inspector](#)，以持續掃描執行期環境是否有已知的常見漏洞和暴險（CVEs）。您可以透過自動化執行手冊，使用 [AWS Systems Manager](#) 套用修補程式或從全新映像重新部署，讓您的運算機群隨時擁有最新的軟體和程式庫。使用這些技術可減少對手動程序和互動式存取運算資源的需求。請參閱 [SEC06-BP03 減少手動管理和互動式存取](#)以進一步了解。

自動化在部署值得信任的工作負載方面也扮演了重要角色，如 [SEC06-BP02 佈建從強化影像運算](#)和 [SEC06-BP04 驗證軟體完整性](#)所述。您可以使用 [EC2 Image Builder](#)、[AWS CodeArtifact](#)、[AWS Signer](#)和 [Amazon Elastic Container Registry（ECR）](#) 等服務來下載、驗證、建構和存放強化且核准的映像和程式碼相依性。除了 Inspector 之外，這些都可以在您的 CI/CD 程序中扮演角色，因此您的工作負載只有在確認其相依性是 up-to-date 來自可信任來源時，才能進行生產。您的工作負載也會經過簽署，因此 AWS 運算環境，例如 [AWS Lambda](#)和 [Amazon Elastic Kubernetes Service（EKS）](#) 可以在允許執行之前驗證其尚未遭到竄改。

除了這些預防性控制之外，您還可以在偵測控制中針對運算資源使用自動化。例如，[AWS Security Hub](#)提供 [NIST 800-53 修訂版 5 標準](#)，其中包含檢查，例如 [【EC2.8】EC2執行個體，應使用執行個體中繼資料服務第 2 版（IMDSv2）](#)。IMDSv2 使用工作階段身分驗證技術，封鎖包含 X-Forwarded-For HTTP 標頭的請求，以及 1 TTL 的網路，以停止來自外部來源的流量，以擷取 EC2 執行個體的相關資訊。Security Hub 中的此檢查可以偵測 EC2 執行個體何時使用 IMDSv1 並啟動自動修復。進一步了解 [SEC04-BP04 中的自動偵測和修復 啟動不合規資源的修復](#)。

實作步驟

1. AMIs 使用 [EC2 Image Builder 自動化建立安全、合規和強化](#)。您可以產生包含來自 Center for Internet Security（CIS）Benchmarks 或 Security Technical Implementation Guide（STIG）標準之控制項的映像，這些標準來自基礎映像 AWS 和 APN 合作夥伴映像。

2. 自動化組態管理。藉由使用組態管理服務或工具，在您的運算資源中自動強制執行和驗證安全組態。
 - a. 使用 [AWS Config](#) 自動化組態管理
 - b. 使用 [AWS Security Hub](#) 自動化安全和合規狀態管理
3. 自動化修補或取代 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體。AWS Systems Manager Patch Manager 會使用安全相關和其他類型的更新，自動修補受管執行個體的程序。您可以使用修補程式管理員以套用適用於作業系統和應用程式的修補程式。
 - a. [AWS Systems Manager 修補程式管理員](#)
4. 自動掃描運算資源是否有常見漏洞和暴露 (CVEs)，並在建置管道中嵌入安全掃描解決方案。
 - a. [Amazon Inspector](#)
 - b. [ECR 映像掃描](#)
5. 考慮使用 Amazon GuardDuty 進行自動惡意軟體和威脅偵測，以保護運算資源。GuardDuty 也可以在您的 AWS 環境中調用 [AWS Lambda](#) 函數時識別潛在問題。
 - a. [Amazon GuardDuty](#)
6. 考慮 AWS 合作夥伴解決方案。AWS 合作夥伴提供與內部部署環境中現有控制項等效、相同或整合的業界領先產品。這些產品可補充現有的 AWS 服務，讓您在雲端和內部部署環境中部署全方位的安全架構，以及擁有更流暢的體驗。
 - a. [基礎設施安全性](#)

資源

相關的最佳實務：

- [SEC01-BP06 自動化標準安全控制的部署](#)

相關文件：

- [取得IMDSv1基礎設施 AWS 的完整優勢IMDSv2並停用](#)

相關影片：

- [Amazon EC2執行個體中繼資料服務的安全最佳實務](#)

資料保護

在建構任何工作負載之前，應先制訂有影響安全的基礎實務。例如，資料分類可基於敏感性等級將資料分類，加密則能對未經授權的存取將資料呈現為無法辨識，以保護資料。這些方法之所以重要，因為能支援例如防止處理不當，或遵循法規義務等目標。

在 AWS 中，進行資料保護時有多種不同的方法可供使用。以下一節介紹如何使用這些方法。

主題

- [資料分類](#)
- [保護靜態資料](#)
- [保護傳輸中的資料](#)

資料分類

資料分類可讓您根據關鍵性和敏感度將組織資料分類，協助判定適當的保護和保留控制。

最佳實務

- [SEC07-BP01 了解您的資料分類機制](#)
- [SEC07-BP02 根據資料敏感性實施資料保護控制](#)
- [SEC07-BP03 自動識別和分類](#)
- [SEC07-BP04 定義可擴展的資料生命週期管理](#)

SEC07-BP01 了解您的資料分類機制

了解您的工作負載要處理的資料分類、其處理需求、關聯的業務流程、資料儲存在何處，以及誰是資料擁有者。您的資料分類和處理機制應考慮工作負載的適用法律和合規需求，以及需要何種資料控制。了解資料是資料分類之旅的第一步。

預期成果：您的工作負載中存在的資料類型已得到充分了解並加以記錄。設置了適當的控制，可根據資料分類來保護敏感資料。這些控制左右著下列考量：允許誰存取資料及存取的目的為何、資料儲存在何處、該資料的加密政策及如何管理加密金鑰、資料的生命週期及其保留需求、適當的銷毀程序、設置了哪些備份和復原程序，以及存取權稽核。

常見的反模式：

- 未制定正式的資料分類政策來定義資料敏感程度及其處理需求
- 未充分了解工作負載內資料的敏感程度，也未在架構和營運文件中擷取這些資訊
- 未能根據您的資料分類和處理政策中所述的資料敏感程度和需求，對資料實施適當的控制
- 未能向政策負責人提供有關資料分類和處理需求的意見回饋。

建立此最佳實務的優勢：此實務可消除有關適當處理工作負載內資料的不確定性。實施正式政策來定義組織中資料的敏感程度及其所需防護措施，有助於符合法律規範和其他網路安全鑑定與認證。工作負載負責人清楚知道敏感資料儲存在何處以及設置了哪些保護控制，因而能夠放心。將這些資訊納入文件中，可協助新的團隊成員更充分了解並在任職期間及早採取控制。這些實務還可針對每一種資料類型實施適當的控制，進而有助於降低成本。

未建立此最佳實務時的曝險等級：高

實作指引

在設計工作負載時，您可能會考量採取直接了當的方式保護敏感資料。例如，在多租用戶應用程式中，直接將每一個租用戶的資料視為敏感資料並採取防護措施，讓租用戶無法存取其他租用戶的資料。同樣地，您可能會直接設計存取控制，只讓管理員修改資料，而其他使用者只擁有讀取層級存取權，或完全無存取權。

藉由在政策中定義並擷取這些資料敏感程度，以及其資料保護需求，您就能正式確定哪些資料要放在您的工作負載中。然後，您可以確定是否設置了正確的控制、是否可稽核控制，以及在發現資料遭不當處理的情況時，要採取何種適當的回應。

為了有助於識別敏感資料存放在工作負載中的位置，請考慮使用資料目錄。資料目錄是對應您組織中資料的資料庫，其位置、敏感度層級和既有控制機制都會保護該資料。此外，請考慮使用可用的[資源標籤](#)。例如，您可以對受保護醫療資訊 (PHI) 套用標籤索引鍵為 Classification 且標籤值為 PHI 的標籤，以及另一個標籤索引鍵為 Sensitivity 且標籤值為 High 的標籤。然後，您可以使用 [AWS Config](#) 等服務來監控這些資源是否發生變更，並且在發生修改後導致資源不符合保護需求 (例如變更加密設定) 的情況時發出提醒。您可以使用[標籤政策](#) (AWS Organizations 的功能) 擷取標籤索引鍵的標準定義和可接受的值。不建議在標籤索引鍵或值中包含私人或敏感資料。

實作步驟

1. 了解組織的資料分類機制和保護需求。
2. 識別工作負載處理的敏感資料類型。
3. 在資料目錄中擷取資料，該目錄提供單一檢視讓您得知資料在組織中的位置，以及該資料的敏感度層級。

4. 考慮使用資源和資料層級標記 (如可用) 來標記資料的敏感程度和其他操作中繼資料，以協助監控和回應事件。
 - a. AWS Organizations 標籤政策可用來強制實施標記標準。

資源

相關的最佳實務：

- [SUS04-BP01 實作資料分類政策](#)

相關文件：

- [資料分類白皮書](#)
- [標記 AWS 資源的最佳實務](#)

相關範例：

- [AWS Organizations 標籤政策語法和範例](#)

相關工具

- [AWS Tag Editor](#)

SEC07-BP02 根據資料敏感性實施資料保護控制

實施資料保護控制，為您分類政策中定義的每一個資料類別提供適當的控制層級。此實務可讓您保護敏感資料防止遭到未經授權的存取和使用，同時讓資料保持可用且實用。

預期成果：您設置了分類政策，在組織中定義不同程度的資料敏感性。您針對每一種敏感程度發布了清楚的指引，以界定核准的儲存和處理服務與位置，以及其所需的組態。您根據所需的保護層級及其關聯成本，針對每一種敏感程度實施控制。您設置了監控和提醒，以偵測資料是否出現在未經授權的位置、在未經授權的環境中經過處理、遭到未經授權的人員存取，或是相關服務的組態是否變得不合規。

常見的反模式：

- 對所有資料實施相同層級的保護控制。這可能會導致對低敏感性資料過度佈建安全控制，或對高敏感性資料的保護不足。
- 在定義資料保護控制時，未邀集安全、合規和業務團隊的利害關係人參與此過程。
- 忽略實施和維護資料保護控制伴隨的營運支出和成本。
- 未定期審查資料保護控制，而未能持續遵循分類政策。
- 沒有靜態資料和傳輸中資料位置的完整庫存。

建立此最佳實務的優勢：藉由依照資料分類層級實施您的控制，您的組織就能在需要時投入更高層級的控制。這可能包括增加保障安全、監控、衡量、修復和報告方面的資源。在適度採取較少控制的情況下，您可以改善員工、客戶或成員使用的資料存取性和完整性。此方法為您的組織帶來了最大的資料使用彈性，同時遵守資料保護要求。

未建立此最佳實務時的曝險等級：高

實作指引

根據資料敏感程度實施資料保護控制的方式包含幾個重要的步驟。首先，識別工作負載架構內不同的資料敏感程度 (例如公開、內部、機密和受限)，並評估您儲存和處理這些資料的位置。接著，根據資料敏感程度定義其隔離界限。建議您使用[服務控制政策 \(SCP\)](#) 將資料分隔到不同的 AWS 帳戶中，以限制每一種資料敏感程度允許的服務和動作。這樣一來，您就可以建立強大的隔離界限，並強制執行最低權限原則。

定義隔離界限之後，根據資料敏感程度實施適當的保護控制。請參閱[保護靜態資料](#)和[保護傳輸中的資料](#)的最佳實務，以實作加密、存取控制和稽核等相關控制。考慮採用記號化或匿名化等技術來降低資料的敏感程度。採用集中式系統進行記號化和去記號化，以簡化對整個企業套用一致的資料政策的程序。

持續監控和測試所實作控制的有效性。隨著組織的資料態勢和威脅發展，定期審查和更新資料分類機制、風險評估和保護控制。實作的資料保護控制務必遵循相關產業法規、標準和法律要求。此外，提供安全意識和培訓，協助員工了解資料分類機制及他們在處理和保護敏感資料方面的責任。

實作步驟

1. 識別工作負載內資料的分類和敏感程度。
2. 為每一種敏感程度定義隔離界限，並確定執行策略。
3. 評估您定義的控制是否確實有效控管您的資料分類政策規定的存取、加密、稽核、保留和其他方面。
4. 評估能適時降低資料敏感程度的選項，例如使用記號化或匿名化。

5. 使用自動測試和監控所設定資源的方式來驗證您的控制。

資源

相關的最佳實務：

- [PERF03-BP01 使用最能滿足資料存取和儲存需求的專用資料存放區](#)
- [COST04-BP05 強制執行資料保留政策](#)

相關文件：

- [資料分類白皮書](#)
- [安全、身分及合規最佳實務](#)
- [AWS KMS 最佳實務](#)
- [AWS 服務的加密最佳實務和功能](#)

相關範例：

- [建置無伺服器記號化解決方案為敏感資料提供遮罩](#)
- [如何使用記號化的方式來提高資料安全並縮小稽核範圍](#)

相關工具：

- [AWS Key Management Service \(AWS KMS\)](#)
- [AWS CloudHSM](#)
- [AWS Organizations](#)

SEC07-BP03 自動識別和分類

將資料的識別和分類自動化，可協助您實作正確的控制。使用自動化增強手動判斷，可降低人為錯誤和暴露的風險。

預期成果：您可根據您的分類和處理政策來確認是否有適當的控制。自動化工具和服務可協助您識別和分類資料的敏感程度。自動化還可協助您持續監控環境，以偵測並提醒未經授權的資料儲存或處理行為，以便快速採取更正動作。

常見的反模式：

- 僅仰賴手動程序來識別和分類資料，這個過程可能容易出錯且相當耗時。這樣做可能導致資料分類效率不彰且不一致，尤其隨著資料量增加會每況愈下。
- 未設置追蹤和管理整個組織中資料資產的機制。
- 即使資料在組織內移動和發展，組織仍然忽略持續監控和分類資料的需要。

建立此最佳實務的優勢：採取自動識別和分類資料的方式，能夠更一致且準確地實施資料保護控制，進而降低人為錯誤的風險。自動化還可讓您深入洞悉敏感資料存取和移動的情形，進而協助您偵測未經授權的處理，並採取更正動作。

未建立此最佳實務時的曝險等級：中

實作指引

在工作負載的初始設計階段常會採用人為判斷來分類資料，儘管如此，仍請考慮設置系統來自動識別和分類測試資料，以此做為預防性控制。例如，您可提供工具或服務讓開發人員用來掃描代表性的資料，以確定其敏感性。在 AWS 內，您可以將資料集上傳到 [Amazon S3](#)，並使用 [Amazon Macie](#)、[Amazon Comprehend](#) 或 [Amazon Comprehend Medical](#) 掃描這些資料集。同樣地，請考慮在單元和整合測試的過程中掃描資料，以偵測不該出現敏感資料的位置。在此階段發出有關敏感資料的提醒，就能在部署到實際執行環境之前，讓防護措施的落差浮現。其他諸如 [AWS Glue](#)、[Amazon SNS](#) 和 [Amazon CloudWatch](#) 中的敏感資料偵測等功能，也可用來偵測 PII 和採取緩解措施。對於任何自動化工具或服務，務必了解其如何定義敏感資料，並利用其他人為或自動化解決方案加強它，以視需要消除任何落差。

持續監控您的環境，以其做為偵測控制，藉以偵測敏感資料是否以不合規的方式儲存。這樣做有助於偵測出在未適當去識別化或修訂的情況下，將敏感資料發送到日誌檔案或複製到資料分析環境中的情形。您可以使用 Amazon Macie 持續監控儲存在 Amazon S3 中的資料，以偵測其中是否存在敏感資料。

實作步驟

1. 請檢閱 [SEC07-BP01](#) 中所述，了解您組織內的資料分類結構。
 - a. 只要了解您組織的資料分類結構，您就可以建立準確的流程用於自動化適合您公司政策的識別和分類。
2. 對您的環境執行初始掃描，以進行自動識別和分類。
 - a. 初始完整掃描資料有助於全面了解敏感資料在您環境中的位置。若一開始不需要或因成本考量而無法事先完成完整掃描，請評估資料取樣技術是否適合用來實現您的成果。例如，您可以設定

Amazon Macie 跨 S3 儲存貯體執行廣泛的自動化敏感資料探索操作。此功能使用的取樣技術會以符合成本效益的方式初步分析敏感資料的所在位置。然後，您可以使用敏感資料探索工作來深入分析 S3 儲存貯體。也可以將其他資料存放區匯出至 S3，以便讓 Macie 進行掃描。

- b. 針對掃描中識別的資料儲存資源，依照 [SEC07-BP02](#) 中所定義來建立存取控制。
3. 設定環境的持續掃描。
 - a. Macie 的自動化敏感資料探索功能可用來持續掃描您的環境。若有任何經授權儲存敏感資料的已知 S3 儲存貯體，則可使用 Macie 中的允許清單將其排除在外。
 4. 將識別和分類納入您的建置和測試程序中。
 - a. 識別開發人員可在工作負載開發過程中用來掃描資料以判斷敏感性的工具。在整合測試的過程中使用這些工具，以便在敏感資料意外出現時發出提醒，並防止進一步部署。
 5. 在未經授權的位置發現敏感資料時，實作系統或執行手冊來採取行動。
 - a. 使用自動修復來限制對資料的存取。例如，若您使用屬性型存取控制 (ABAC)，就可以將此資料移至限制存取的 S3 儲存貯體或標記物件。此外，請考慮在偵測到資料時，為其加上遮罩。
 - b. 提醒您的資料防護和事件回應團隊調查事件的根本原因。他們在調查過程中學到的任何經驗，都有助於防止未來的事件發生。

資源

相關文件：

- [AWS Glue：偵測和處理敏感資料](#)
- [在 Amazon SNS 中使用受管資料識別碼](#)
- [Amazon CloudWatch Logs：使用遮罩協助保護敏感日誌資料](#)

相關範例：

- [使用 Macie 為 Amazon RDS 資料庫啟用資料分類](#)
- [使用 Macie 偵測 DynamoDB 中的敏感資料](#)

相關工具：

- [Amazon Macie](#)
- [Amazon Comprehend](#)
- [Amazon Comprehend Medical](#)

- [AWS Glue](#)

SEC07-BP04 定義可擴展的資料生命週期管理

了解您的資料生命週期需求，因為這些需求與您不同層級的資料分類和處理相關。這可能包括資料一開始進入環境時的處理方式、資料轉換的方式，以及銷毀資料的規則。請將保留期、存取、稽核和追蹤來源等因素納入考量。

預期成果：您的資料分類會盡可能接近擷取點和時間。當資料分類需要遮罩、記號化或其他降低敏感程度的處理時，您會在盡可能最接近擷取點和時間的條件下執行這些動作。

當資料不再適合保存時，您會遵循政策根據資料的分類將其刪除。

常見的反模式：

- 實作一體適用的方法來管理資料生命週期，而未考量不同的敏感程度和存取需求。
- 僅從資料為可用資料或備份資料的角度來考量生命週期管理，而非兩者均考量。
- 假設已輸入工作負載的資料有效，但未確定其價值或來源。
- 仰賴資料耐久性來替代資料備份和保護。
- 保留資料的時間超過其實用性和所需的保留期。

建立此最佳實務的優勢：定義明確且可擴展的資料生命週期管理策略有助於保持合規、提高資料安全性、最佳化儲存成本，以及在維持適當控制之下實現有效率的資料存取和共用。

未建立此最佳實務時的風險暴露等級：高

實作指引

工作負載內的資料通常是動態的。資料進入工作負載環境時採取的形式，可能與資料儲存或使用在商業邏輯、報告、分析或機器學習上的形式有所不同。此外，資料的價值可能隨時間而改變。有些資料本質上是暫時性的，會隨著時間失去其價值。請考量在您的資料分類機制與相關控制下，這些資料變更對評估的影響。可能的話，盡量使用自動化生命週期機制 (如 [Amazon S3 生命週期政策](#) 和 [Amazon Data Lifecycle Manager](#)) 來設定資料保留、封存和到期程序。對於儲存在 DynamoDB 中的資料，您可以使用 [存留時間 \(TTL\)](#) 功能來定義每個項目的過期時間戳記。

區分可供使用的資料與儲存為備份的資料。考慮使用 [AWS Backup](#) 自動備份 AWS 服務中的資料。[Amazon EBS 快照](#) 提供了複製 EBS 磁碟區並使用 S3 功能來儲存它的方式，包括生命週期、資料保護和存取保護機制。其中兩種機制為 [S3 Object Lock](#) 和 [AWS Backup Vault Lock](#)，皆可提高您備份

的安全性，並且讓您更有效地掌控備份。進行分明的職責和備份存取權劃分管理。在帳戶層級隔離備份，以便在事件發生期間與受影響的環境保持分離。

生命週期管理的另一方面，是記錄資料在工作負載中進度的歷史記錄，稱為資料來源追蹤。如此您就能確信自己知道資料來自何處、執行的任何轉換、哪些擁有者或處理程序做出這些變更，以及時間點。這份歷史記錄有助於在可能發生安全事件的期間進行問題的疑難排解和調查。例如，您可以在 [Amazon DynamoDB](#) 資料表中記錄有關轉換的中繼資料。在資料湖內，您可以針對每一個資料管道階段，將轉換後資料的副本保留在不同的 S3 儲存貯體中。將結構描述和時間戳記資訊儲存在 [AWS Glue Data Catalog](#) 中。無論您採用何種解決方案，請務必考量最終使用者的需求，以確定報告資料來源所需的適當工具。這樣做將協助您確定追蹤來源的最佳方式。

實作步驟

1. 分析工作負載的資料類型、敏感程度及存取需求，以分類資料並定義適當的生命週期管理策略。
2. 設計並實施符合法律、法規和組織要求的資料保留政策及自動銷毀程序。
3. 建立流程和自動化功能，以隨著工作負載需求和法規發展，持續監控、稽核和調整資料生命週期管理策略、控制及政策。
 - a. 使用 [AWS Config](#) 偵測尚未開啟自動生命週期管理的資源

資源

相關的最佳實務：

- [COST04-BP05 強制執行資料保留政策](#)
- [SUS04-BP03 使用政策來管理資料集的生命週期](#)

相關文件：

- [資料分類白皮書](#)
- [AWS 勒索軟體防禦藍圖](#)
- [DevOps 指引：利用資料來源追蹤改善可追溯性](#)

相關範例：

- [如何在 AWS 中於整個生命週期保護敏感資料](#)
- [使用 AWS Glue、Amazon Neptune 和 Spline 為資料湖建置資料歷程](#)

相關工具：

- [AWS Backup](#)
- [Amazon Data Lifecycle Manager](#)
- [AWS Identity and Access Management Access Analyzer](#)

保護靜態資料

靜態資料代表您在工作負載中的任何期間，保留在非揮發性儲存體中的任何資料。其中包括：長期存放資料的區塊儲存體、物件儲存體、資料庫、封存、IoT 裝置和任何其他儲存媒介。實作加密和適當的存取控制，保護靜態資料能將未經授權存取的風險降低。

加密和記號化是兩個重要但截然不同的資料保護方案。

記號化是一種過程，可讓您定義權杖來表示敏感資訊 (例如，用權杖來表示客戶的信用卡號)。記號本身必須毫無意義，而且不可衍生自記號化的資料，因此，密碼編譯摘要無法作為記號使用。透過仔細地規劃記號化的方法，您可以為內容提供額外的保護，並確保滿足合規要求。例如，如果您善用記號而非使用信用卡號碼，則可以縮小信用卡處理系統的合規範圍。

加密是一種轉換內容的方式，若沒有將內容解密回純文字所需的私密金鑰，就無法讀取。記號化和加密都可以用來適當地保護資訊。此外，遮罩這種技術也允許將資料片段修訂成為剩餘的資料不視為敏感的狀態。例如，PCI-DSS 允許超出合規範圍邊界，保留卡號的後四碼以編製索引。

稽核加密金鑰的使用：請確定您了解並稽核加密金鑰的使用，以驗證金鑰的存取控制機制是否正確實作。例如，任何使用 AWS KMS 金鑰的 AWS 服務，都會將每次的使用記錄在 AWS CloudTrail 中。然後，您可以使用 Amazon CloudWatch Logs Insights 等工具來查詢 AWS CloudTrail，以確保金鑰的所有使用都是有效的。

最佳實務

- [SEC08-BP01 實作安全金鑰管理](#)
- [SEC08-BP02 強制靜態加密](#)
- [SEC08-BP03 自動化靜態資料保護](#)
- [SEC08-BP04 強制存取控制](#)

SEC08-BP01 實作安全金鑰管理

安全金鑰管理包括儲存、輪換、存取控制及監控保護工作負載的靜態資料所需的金鑰資料。

預期成果：您擁有可擴展、可重複且自動化的金鑰管理機制。此機制會對金鑰資料強制執行最低權限存取，並且在金鑰可用性、機密性和完整性之間取得正確的平衡。您會監控金鑰的存取情況，若需要輪換金鑰資料，您可以使用自動化程序進行輪換。您不允許真人操作員存取金鑰資料。

常見的反模式：

- 真人存取未加密的金鑰資料。
- 建立自訂加密演算法。
- 存取金鑰資料的許可過於廣泛。

建立此最佳實務的優勢：透過為工作負載建立安全的金鑰管理機制，就可以協助保護您的內容，防止未經授權的存取。此外，您可能需要依法加密您的資料。有效的金鑰管理解決方案能夠提供符合這些法規的技術機制，以保護金鑰資料。

未建立此最佳實務時的曝險等級：高

實作指引

靜態資料加密是基本的安全控制。為了實作此控制，您的工作負載須採取某種機制，以安全儲存和管理用於加密靜態資料的金鑰資料。

AWS 提供 AWS Key Management Service (AWS KMS) 來為 AWS KMS 金鑰提供耐用、安全和備援儲存。[許多 AWS 服務會與 AWS KMS 整合](#)，以支援資料加密。AWS KMS 使用 FIPS 140-2 3 級驗證的硬體安全模組來保護您的金鑰。沒有任何機制可將 AWS KMS 金鑰匯出為純文字。

使用多帳戶策略部署工作負載時，您應將 AWS KMS 金鑰與使用金鑰的工作負載保存在相同的帳戶中。[這個分散式模型](#)會將管理 AWS KMS 金鑰的責任歸在您的團隊身上。在其他使用案例中，您的組織可能會選擇將 AWS KMS 金鑰儲存到集中式帳戶中。此集中式結構須實施其他政策來實現跨帳戶存取權，才能讓工作負載帳戶存取儲存在集中式帳戶中的金鑰，但此結構可能較適合跨多個 AWS 帳戶共用單一金鑰的使用案例。

無論金鑰資料儲存在何處，您都應透過使用[金鑰政策](#)和 IAM 政策嚴格控管金鑰存取權。金鑰政策是控制 AWS KMS 金鑰存取權的主要方式。此外，AWS KMS 金鑰授權可提供 AWS 服務的存取權，以代表您加密和解密資料。檢閱 [AWS KMS 金鑰存取控制指引](#)。

您應監控加密金鑰的使用情況，以偵測不尋常的存取模式。使用儲存在 AWS KMS 中 AWS 管理的金鑰和客戶管理的金鑰執行的操作可記錄在 AWS CloudTrail 中，並且應定期審查。務必特別注意監控金鑰銷毀事件。為了減少意外或惡意銷毀金鑰資料的情況，金鑰銷毀事件並不會立即刪除金鑰資料。嘗試

刪除 AWS KMS 中的金鑰會受到等待期的約束 (預設為 30 天，最短 7 天)，讓管理員有時間檢閱這些動作，並在必要時撤回請求。

大多數 AWS 服務會以顯而易見的方式使用 AWS KMS，您唯一要做的就是決定要使用 AWS 管理或客戶管理的金鑰。如果您的工作負載要求直接使用 AWS KMS 來加密或解密資料，則您應使用封套加密來保護您的資料。此 [AWS Encryption SDK](#) 可為您的應用程式提供用戶端加密基本類型，以實作封套加密並與 AWS KMS 整合。

實作步驟

1. 確定金鑰的適當金鑰管理選項 (AWS 受管或客戶自管)。
 - a. 為了方便使用，AWS 為大多數服務提供了 AWS 擁有和 AWS 管理的金鑰，其提供靜態加密功能，而不需要管理金鑰資料或金鑰政策。
 - b. 使用客戶管理的金鑰時，請考慮使用預設金鑰存放區，以便在敏捷性、安全性、資料主權與可用性之間達到最佳平衡。其他使用案例可能會要求使用自訂金鑰存放區搭配 [AWS CloudHSM](#) 或外部金鑰存放區。
2. 檢閱您用於工作負載的服務清單，以了解 AWS KMS 與服務整合的方式。例如，EC2 執行個體可以使用加密的 EBS 磁碟區，因此要確認從這些磁碟區建立的 Amazon EBS 快照同樣是使用客戶管理的金鑰加密，並減少意外洩漏未加密的快照資料。
 - a. [AWS 服務使用 AWS KMS 的方式](#)
 - b. 如需 AWS 服務提供之加密選項的詳細資訊，請參閱該服務之使用者指南或開發人員指南中的靜態加密主題。
3. 實作 AWS KMS：AWS KMS 可讓您輕鬆建立和管理金鑰，並控制多種 AWS 服務和應用程式中的加密使用方式。
 - a. [入門：AWS Key Management Service \(AWS KMS\)](#)
 - b. 檢閱 [AWS KMS 金鑰存取控制的最佳實務](#)。
4. 考慮使用 AWS Encryption SDK：當您的應用程式需要在用戶端對資料進行加密時，可使用整合 AWS KMS 的 AWS Encryption SDK。
 - a. [AWS Encryption SDK](#)
5. 啟用 [IAM Access Analyzer](#) 以自動檢閱並在發現有過度廣泛的 AWS KMS 金鑰政策時發出通知。
 - a. 考慮使用 [自訂政策檢查](#)，以確認資源政策更新未將公開存取權授予 KMS 金鑰。
6. 啟用 [Security Hub](#) 以在金鑰政策設定錯誤、有排定要刪除的金鑰，或有未啟用自動輪替的金鑰時收到通知。
7. 確定適合 AWS KMS 金鑰的日誌記錄層級。由於 AWS KMS 的呼叫 (包括唯讀事件) 會加以記錄，因此與 AWS KMS 關聯的 CloudTrail 日誌可能會變得很龐大。

- a. 有些組織偏好將 AWS KMS 日誌記錄活動分隔為單獨的追蹤。如需詳細資訊，請參閱《AWS KMS 開發人員指南》的 [Logging AWS KMS API calls with CloudTrail](#)。

資源

相關文件：

- [AWS Key Management Service](#)
- [AWS 加密服務和工具](#)
- [使用加密保護 Amazon S3 資料](#)
- [信封加密](#)
- [數位主權承諾](#)
- [揭密 AWS KMS 金鑰操作、攜帶自有金鑰、自訂金鑰存放區，以及密文可攜性](#)
- [AWS Key Management Service 密碼編譯詳細資訊](#)

相關影片：

- [在 AWS 中加密如何運作](#)
- [保護 AWS 上的區塊儲存安全](#)
- [AWS 資料保護：使用鎖定、金鑰、簽章和憑證](#)

相關範例：

- [使用 AWS KMS 實作進階存取控制機制](#)

SEC08-BP02 強制靜態加密

加密私有靜態資料以維持機密性，並提供一層額外的保護，以防止意外揭露或洩露資料。加密可保護資料，防止資料在未先解密的情況下遭到讀取或存取。清查和控制未加密的資料，以降低資料暴露伴隨的風險。

預期成果：您建立了預設加密靜態私有資料的機制。這些機制有助於維持資料的機密性，並提供一層額外的保護，以防有意或不慎揭露或洩露資料。您維護未加密資料的庫存，並了解為了保護資料而採取的控制措施。

常見的反模式：

- 未使用預設加密組態。
- 對解密金鑰提供過於寬鬆的存取權。
- 未監控加密和解密金鑰的使用。
- 在未加密的情況下儲存資料。
- 對所有資料使用相同的加密金鑰，無論資料使用方式、類型和分類。

未建立此最佳實務時的曝險等級：高

實作指引

在工作負載中將加密金鑰對應到資料分類。當對資料使用單一或極少數的加密金鑰時，此方法有助於防止過於寬鬆的存取權 (請參閱 [SEC07-BP01 了解您的資料分類機制](#))。

AWS Key Management Service (AWS KMS) 與許多 AWS 服務整合，更方便您加密靜態資料。例如，在 Amazon Elastic Compute Cloud (Amazon EC2) 中，您可以在帳戶上設定[預設加密](#)，以便自動加密新的 EBS 磁碟區。當使用 AWS KMS 時，考慮需要嚴格限制資料的程度。AWS 會代表您管理及使用預設和服務控制的 AWS KMS 金鑰。對於需要對基礎加密金鑰的精細存取權之敏感資料，可考慮客戶自管金鑰 (CMK)。您可全權控制 CMK，包括透過使用金鑰政策進行輪換和存取管理。

此外，像是 Amazon Simple Storage Service ([Amazon S3](#)) 等服務現在預設會加密所有新物件。此實作提供了更強大的安全性，而且不會影響效能。

其他服務像是 [Amazon Elastic Compute Cloud](#) (Amazon EC2) 或 [Amazon Elastic File System](#) (Amazon EFS)，都可支援預設加密的設定。您也可以使用 [AWS Config 規則](#) 自動檢查您是否對 [Amazon Elastic Block Store \(Amazon EBS\) 磁碟區](#)、[Amazon Relational Database Service \(Amazon RDS\) 執行個體](#) 和 [Amazon S3 儲存貯體](#)，以及組織內的其他服務使用加密。

AWS 也提供用戶端加密選項，允許您在上傳到雲端之前加密資料。AWS Encryption SDK 提供使用[封套加密](#)來加密資料的方法。您提供包裝金鑰，而 AWS Encryption SDK 會為它加密的每個資料物件產生唯一的資料金鑰。如果您需要受管的單一租用戶硬體安全模組 (HSM)，可考慮 AWS CloudHSM。AWS CloudHSM 可讓您在 FIPS 140-2 3 級驗證的 HSM 上產生、匯入和管理加密金鑰。AWS CloudHSM 的一些使用案例包括保護用於核發憑證認證機構 (CA) 的私有金鑰，以及為 Oracle 資料庫開啟透明資料加密 (TDE)。AWS CloudHSM 用戶端 SDK 提供軟體，可讓您在將資料上傳到 AWS 之前，使用儲存在 AWS CloudHSM 內的金鑰加密資料用戶端。Amazon DynamoDB Encryption Client 還允許您在上傳到 DynamoDB 資料表之前，加密和簽署項目。

實作步驟

- 為新的 [Amazon EBS 磁碟區設定預設加密](#)：使用 AWS 提供的預設金鑰或您自行建立的金鑰，指定您希望以加密形式建立所有新的 Amazon EBS 磁碟區。
- 設定加密的 Amazon Machine Images (AMI)：複製已設定加密的現有 AMI 會自動加密根磁碟區和快照。
- 設定 [Amazon RDS 加密](#)：透過使用加密選項，為您的 Amazon RDS 資料庫叢集和靜態快照設定加密。
- 使用政策限制對適當主體的存取，為每個資料分類建立和設定 AWS KMS 金鑰：例如，建立一個 AWS KMS 金鑰用於加密生產資料，另一個金鑰用於加密開發或測試資料。您還可以提供金鑰來存取其他 AWS 帳戶。考慮針對開發和生產環境擁有不同的帳戶。如果您的生產環境需要解密開發帳戶中的成品，您可以編輯用來加密開發成品的 CMK 金鑰，使生產帳戶能夠解密這些成品。生產環境接著可以擷取解密的資料以用於生產。
- 在其他 AWS 服務中設定加密：對於您使用的其他 AWS 服務，請檢閱該服務的[安全文件](#)，以確定該服務的加密選項。

資源

相關文件：

- [AWS 加密工具](#)
- [AWS Encryption SDK](#)
- [AWS KMS 加密詳細資訊白皮書](#)
- [AWS Key Management Service](#)
- [AWS 加密服務和工具](#)
- [Amazon EBS 加密](#)
- [Amazon EBS 磁碟區的預設加密](#)
- [加密 Amazon RDS 資源](#)
- [如何啟用 Amazon S3 儲存貯體的預設加密？](#)
- [使用加密保護 Amazon S3 資料](#)

相關影片：

- [在 AWS 中加密如何運作](#)

- [保護 AWS 上的區塊儲存安全](#)

SEC08-BP03 自動化靜態資料保護

使用自動化來驗證和強制執行靜態資料控制。使用自動掃描來偵測資料儲存解決方案的錯誤組態，並盡可能透過自動化的程式化回應執行補救措施。將自動化納入 CI/CD 程序中，以在部署到實際執行環境之前先偵測是否有資料儲存組態錯誤的情形。

預期成果：自動化系統會掃描和監控資料儲存位置，找出是否有控制組態錯誤、未經授權存取及意外使用的情況。偵測到設定錯誤的儲存位置就會啟動自動化補救措施。自動化程序會建立資料備份，並將不可變的副本儲存在原始環境之外。

常見的反模式：

- 未考慮在受支援的情況下，啟用預設加密設定的選項。
- 制定自動備份和復原策略時，未考慮安全事件還有操作事件。
- 未強制執行儲存服務的公開存取設定。
- 未監控和稽核保護靜態資料的控制。

建立此最佳實務的優勢：自動化有助於防止發生資料儲存位置設定錯誤的風險。這有助於防止錯誤組態進入您的實際執行環境。此最佳實務也有助於偵測並修正錯誤組態 (如發生)。

未建立此最佳實務時的曝險等級：中

實作指引

自動化是貫穿保護靜態資料的實務的主題。[SEC01-BP06 自動部署標準安全控制](#)說明了如何使用基礎設施即程式碼 (IaC) 範本 (例如使用 [AWS CloudFormation](#)) 擷取資源的組態。這些範本已遞交至版本控制系統中，且用於在 AWS 上透過 CI/CD 管道部署資源。這些技術同樣適用於自動化資料儲存解決方案的組態，例如 Amazon S3 儲存貯體上的加密設定。

您可以在 CI/CD 管道中使用 [AWS CloudFormation Guard](#) 內的規則檢查您在 IaC 範本中定義的設定是否有組態錯誤。您可以透過 [AWS Config](#) 監控尚未在 CloudFormation 或其他 IaC 工具中提供的設定，以檢查是否有組態錯誤。Config 針對錯誤組態所產生的提醒可以自動修復，如 [SEC04-BP04 針對不合規資源實施補救措施](#) 中所述。

將自動化納入您的許可管理策略中，也是整體自動化資料防護措施的一環。[SEC03-BP02 授予最低權限存取權](#)和 [SEC03-BP04 持續減少許可](#)中說明了如何設定最低權限存取權政策，這些政策會受到

[AWS Identity and Access Management Access Analyzer](#) 的持續監控，以在能夠減少許可時產生調查結果。除了自動化監控許可之外，您還可以設定 [Amazon GuardDuty](#) 來監看 [EBS 磁碟區](#) (藉由 EC2 執行個體)、[S3 儲存貯體](#) 和支援的 [Amazon Relational Database Service 資料庫](#) 中是否存在異常資料存取行為。

自動化也會在偵測到敏感資料儲存於未經授權的位置時，發揮重要的作用。[SEC07-BP03 自動識別和分類](#) 說明了 [Amazon Macie](#) 如何監控您的 S3 儲存貯體是否有非預期的敏感資料，並產生可啟動自動化回應的提醒。

遵循 [REL09 備份資料](#) 中的實務，制定自動化資料備份和復原策略。資料備份和復原對於操作事件，以及從安全事件中復原來說都相當重要。

實作步驟

1. 在 IaC 範本中擷取資料儲存組態。使用自動化檢查在 CI/CD 管道中偵測組態錯誤。
 - a. 您可以將 [AWS CloudFormation](#) 用於您的 IaC 範本，並且將 [AWS CloudFormation Guard](#) 用於檢查範本是否有組態錯誤。
 - b. 使用 [AWS Config](#) 在主動評估模式下執行規則。在建立資源之前，使用此設定做為 CI/CD 管道中的步驟來檢查資源是否合規。
2. 監控資源是否有資料儲存組態錯誤。
 - a. 設定 [AWS Config](#) 來監控資料儲存資源是否有控制組態方面的變更，並在偵測到組態錯誤時產生提醒，以調用修復動作。
 - b. 如需有關自動化補救措施的詳細指引，請參閱 [SEC04-BP04 針對不合規資源實施補救措施](#)。
3. 透過自動化持續監控並減少資料存取許可。
 - a. [IAM Access Analyzer](#) 可持續執行，並在有可能減少許可時產生提醒。
4. 監控並提醒異常資料存取行為。
 - a. [GuardDuty](#) 會監看 EBS 磁碟區、S3 儲存貯體及 RDS 資料庫等資料儲存資源中，是否存在已知的威脅特徵和偏離基準存取行為。
5. 監控並在敏感資料儲存於非預期位置時發出提醒。
 - a. 使用 [Amazon Macie](#) 持續掃描您的 S3 儲存貯體是否有敏感資料。
6. 自動保護和加密資料備份。
 - a. [AWS Backup](#) 是受管服務，會在 AWS 上建立各種資料來源的加密和安全備份。[彈性災難復原](#) 可讓您透過以秒為單位測量的復原點目標 (RPO)，複製完整的伺服器工作負載並維持持續的資料保護。您可以設定讓兩種服務搭配運作，以自動建立資料備份並將其複製到容錯移轉位置。這有助於在受到操作或安全事件影響時，保持資料的可用性。

資源

相關的最佳實務：

- [SEC01-BP06 自動部署標準安全控制](#)
- [SEC03-BP02 授予最低權限存取權](#)
- [SEC03-BP04 持續減少許可](#)
- [SEC04-BP04 針對不合規資源實施補救措施](#)
- [SEC07-BP03 自動識別和分類](#)
- [REL09-BP02 保護和加密備份](#)
- [REL09-BP03 自動執行資料備份](#)

相關文件：

- [AWS 方案指引：自動加密現有和新的 Amazon EBS 磁碟區](#)
- [在 AWS 上使用 NIST 網路安全架構 \(CSF\) 進行勒索軟體風險管理](#)

相關範例：

- [如何使用 AWS Config 主動式規則和 AWS CloudFormation 勾點來防止建立不合規的雲端資源](#)
- [使用 AWS Backup 自動化並集中管理 Amazon S3 的資料保護](#)
- [AWS re:Invent 2023 - 使用 Amazon EBS 快照實作主動式資料保護](#)
- [AWS re:Invent 2022 - 利用現代化資料保護建置並自動化以強化恢復能力](#)

相關工具：

- [AWS CloudFormation Guard](#)
- [AWS CloudFormation Guard 規則登錄檔](#)
- [IAM Access Analyzer](#)
- [Amazon Macie](#)
- [AWS Backup](#)
- [彈性災難復原](#)

SEC08-BP04 強制存取控制

要協助保護您的靜態資料，使用隔離和版本控制等機制來強制存取控制。套用最低權限和條件存取控制。避免授予您資料的公開存取權。

預期成果：確認只有授權使用者才能在必要時存取資料。您透過定期備份和版本控制來保護您的資料，以防有意或不慎修改或刪除資料。您將重要資料與其他資料分離，以保護其機密性和資料完整性。

常見的反模式：

- 將具有不同敏感度需求或分類的資料儲存在一起。
- 對解密金鑰使用過於寬鬆的許可。
- 資料分類不當。
- 未保留重要資料的詳細備份。
- 對生產資料提供持續存取權。
- 未稽核資料存取或定期審查許可。

未建立此最佳實務時的曝險等級：高

實作指引

保護靜態資料對於維護資料完整性、機密性和符合法規要求至關重要。您可以實作多個控制項來協助達成此目標，包括存取控制、隔離、條件式存取和版本控制。

您可以使用最低權限原則強制執行存取控制，該原則僅為使用者和服務提供執行其任務所需的許可權。這包括加密金鑰的存取權。檢閱您的 [AWS Key Management Service \(AWS KMS\) 政策](#)，確認您授予的存取權層級適當，並且適用相關條件。

您可以針對每個層級使用各自不同的 AWS 帳戶，根據不同的分類層級分隔資料，並使用 [AWS Organizations](#) 管理這些帳戶。此隔離有助於預防未經授權的存取，並將資料揭露風險降到最低。

定期審查 S3 儲存貯體政策中授予的存取權層級。除非絕對必要，否則避免使用可公開讀取或可寫入的儲存貯體。考慮使用 [AWS Config](#) 偵測公開可用的儲存貯體，以及使用 Amazon CloudFront 從 Amazon S3 提供內容。確認不允許公開存取的儲存貯體，已正確設定為防止公開存取。

針對存放在 Amazon S3 中的關鍵資料實作版本控制和物件鎖定機制。[Amazon S3 版本控制](#) 會保留舊版物件，以從意外刪除或覆寫中復原資料。[Amazon S3 物件鎖定](#) 為物件提供強制性存取控制功能，此功能可預防遭到刪除或覆寫，即使根使用者也一樣，直到鎖定到期為止。此外，[Amazon S3 Glacier Vault Lock](#) 為存放在 Amazon S3 Glacier 中的封存提供類似的功能。

實作步驟

1. 使用最低權限原則強制執行存取控制：
 - 檢閱授予使用者和服務的存取權限，並確認他們只擁有執行任務所需的許可權。
 - 檢查 [AWS Key Management Service \(AWS KMS\) 政策](#) 以檢閱加密金鑰的存取權。
2. 根據不同的分類層級分隔資料：
 - 針對每個資料分類層級使用不同的 AWS 帳戶。
 - 使用 [AWS Organizations](#) 管理這些帳戶。
3. 檢閱 Amazon S3 儲存貯體和物件許可權：
 - 定期審查 S3 儲存貯體政策中授予的存取權層級。
 - 除非絕對必要，否則避免使用可公開讀取或可寫入的儲存貯體。
 - 考慮使用 [AWS Config](#) 偵測公開可用的儲存貯體。
 - 使用 Amazon CloudFront 提供來自 Amazon S3 的內容。
 - 確認不允許公開存取的儲存貯體，已正確設定為防止公開存取。
 - 您可以對資料庫和使用 IAM 身分驗證的其他任何資料來源，套用相同的審查程序，例如 SQS 或第三方資料存放區。
4. 使用 AWS IAM Access Analyzer：
 - 您可以設定讓 [AWS IAM Access Analyzer](#) 分析 Amazon S3 儲存貯體，並在 S3 政策將存取權授予外部實體時產生調查結果。
5. 實作版本控制和物件鎖定機制：
 - 使用 [Amazon S3 版本控制](#) 保留物件的舊版本，如此即可將意外刪除或覆寫的資料復原。
 - 使用 [Amazon S3 物件鎖定](#) 為物件提供強制存取控制，如此可避免遭到刪除或覆寫，即使根使用者也一樣，直到鎖定功能到期為止。
 - 將 [Amazon S3 Glacier Vault Lock](#) 用於存放在 Amazon S3 Glacier 中的封存。
6. 使用 Amazon S3 庫存清單
 - 使用 [Amazon S3 庫存清單](#) 稽核及報告 S3 物件的複寫和加密狀態。
7. 檢閱 Amazon EBS 和 AMI 共用權限：
 - 檢閱 [Amazon EBS](#) 和 [AMI 共用](#) 權限，確認您的映像和磁碟區未與在工作負載外部的 AWS 帳戶共用。
8. 定期審查 AWS Resource Access Manager Shares：
 - 您可以使用 [AWS Resource Access Manager](#) 共用 Amazon VPC 內的資源，例如 AWS Network Firewall 政策、Amazon Route 53 Resolver 規則和子網路。

- 定期稽核共用的資源，並停止共用不再需要共用的資源。

資源

相關的最佳實務：

- [SEC03-BP01 定義存取需求](#)
- [SEC03-BP02 授予最低權限存取權](#)

相關文件：

- [AWS KMS 加密詳細資訊白皮書](#)
- [管理對 Amazon S3 資源的存取權限的簡介](#)
- [管理 AWS KMS 資源存取的概觀](#)
- [AWS Config 規則](#)
- [Amazon S3 + Amazon CloudFront：雲端中的最佳拍檔](#)
- [使用版本控制](#)
- [使用 Amazon S3 Object Lock 鎖定物件](#)
- [共用 Amazon EBS 快照](#)
- [共用 AMI](#)
- [在 Amazon S3 上託管單頁應用程式](#)
- [AWS 全域條件索引鍵](#)
- [在 AWS 上建置資料周邊](#)

相關影片：

- [保護 AWS 上的區塊儲存安全](#)

保護傳輸中的資料

傳輸中的資料是指從一個系統傳送到另一個系統的任何資料。這包括工作負載內資源之間的通訊，以及其他服務與最終使用者之間的通訊。為傳輸中的資料提供適度的保護，意味著您保護工作負載中資料的機密性和完整性。

保護來自 VPC 之間或內部部署位置的資料：您可以使用 [AWS PrivateLink](#)，在 Amazon Virtual Private Cloud (Amazon VPC) 之間建立安全且私有的網路連線，或建立 AWS 中所託管服務的內部部署連線。您可以存取 AWS 服務、第三方服務，以及其他 AWS 帳戶中的服務，就像這些服務是在您的私有網路上一樣。使用 AWS PrivateLink，您可以跨具有重疊 IP CIDR 的帳戶存取服務，而不需要網際網路閘道或 NAT。您也不必設定防火牆規則、路徑定義或路由表。流量停留在 Amazon 骨幹網路上，而且不會周遊網際網路，因此您的資料受到保護。您可以遵守產業特定的合規法規，例如 HIPAA 和歐盟/美國隱私保護盾。AWS PrivateLink 無縫使用第三方解決方案，建立簡化的全球網路，可讓您加速遷移到雲端並利用可用的 AWS 服務。

最佳實務

- [SEC09-BP01 實作安全金鑰和憑證管理](#)
- [SEC09-BP02 強制傳輸中加密](#)
- [SEC09-BP03 驗證網路通訊](#)

SEC09-BP01 實作安全金鑰和憑證管理

Transport Layer Security (TLS) 憑證可用來保護網路通訊，和建立網際網路跟私有網路中的網站、資源和工作負載的身分。

預期成果：可以在公開金鑰基礎設施 (PKI) 中佈建、部署、儲存和更新憑證的安全憑證管理系統。安全金鑰與憑證管理機制可以防止憑證私有金鑰資料外洩，也能定期自動更新憑證。它也能與其他服務整合，為工作負載內的機器資源提供安全的網路通訊和身分識別。金鑰資料絕不可供真人身分存取。

常見的反模式：

- 在憑證部署或更新程序期間執行手動步驟。
- 設計私有 CA 時，忽略憑證授權單位 (CA) 階層。
- 針對公有資源使用自我簽署憑證。

建立此最佳實務的優勢：

- 透過自動化部署和更新來簡化憑證管理
- 鼓勵使用 TLS 憑證加密傳輸中的資料
- 增加憑證授權機構所採取憑證動作的安全性和可稽核性
- 在 CA 階層中不同層次的管理責任組織

未建立此最佳實務時的曝險等級：高

實作指引

現代工作負載可透過使用 TLS 等 PKI 協定，來廣泛使用加密網路通訊。PKI 憑證管理可能很複雜，但自動化憑證佈建、部署和更新可以減少憑證管理相關障礙。

AWS 提供兩種管理一般用途 PKI 憑證的服務：[AWS Certificate Manager](#) 和 [AWS Private Certificate Authority \(AWS Private CA\)](#)。ACM 是客戶在公有和私有 AWS 工作負載中，用來佈建、管理和部署憑證的主要服務。ACM 則使用 AWS Private CA 發行私有憑證，並與許多其他 AWS 受管服務[整合](#)，以提供安全的工作負載 TLS 憑證。ACM 也可以從 [Amazon Trust Services](#) 發出公有信任憑證。ACM 的公有憑證可用於面向公眾的工作負載，因為新型瀏覽器和作業系統預設信任這些憑證。

AWS Private CA 可讓您建立自己的根憑證授權單位或下層憑證授權單位，並透過 API 發行 TLS 憑證。在您控制和管理 TLS 連線用戶端信任鏈時，您可以使用這些憑證類型。除了 TLS 使用案例之外，AWS Private CA 還可以用來發行憑證給 Kubernetes Pod、重要裝置產品證明、程式碼簽署，以及其他具有[自訂範本](#)的使用案例。您也可以使用 [IAM Roles Anywhere](#) 提供暫時 IAM 憑證給具有您私有 CA 發行 X.509 憑證的內部部署工作負載。

除了 ACM 和 AWS Private CA 之外，[AWS IoT Core](#) 也為 IoT 裝置提供特別支援，用來佈建、管理和部署 PKI 憑證。AWS IoT Core 提供了專門的機制，可將 [IoT 裝置大規模加入](#)您的公有金鑰基礎設施。

有些 AWS 服務如 [Amazon API Gateway](#) 和 [Elastic Load Balancing](#)，會提供使用憑證來保護應用程式連線的自有功能。例如，API Gateway 和 Application Load Balancer (ALB) 都支援交互 TLS (mTLS)，其便是使用您使用 AWS Management Console、CLI 或 API 建立和匯出的用戶端憑證。

建立私有 CA 階層時的考量

在需要建立私有 CA 時，請務必特別留意，預先正確設計 CA 階層。建立私有 CA 階層時，最佳實務是將 CA 階層的每個層級部署到個別 AWS 帳戶。這個刻意的步驟會減少 CA 階層中每個層級的界面面積，讓您更容易察覺 CloudTrail 日誌資料的異常狀況，並在出現未經授權的帳戶存取動作時降低存取或影響範圍。根 CA 應位於自己的個別帳戶中，且只能用來發行一個或多個中繼 CA 憑證。

接著，請在不同於根 CA 帳戶的其他帳戶中建立一個或多個中繼 CA，為終端使用者、裝置或其他工作負載發行憑證。最後，請將憑證從根 CA 發行至中繼 CA，這個動作會將憑證發行給您的終端使用者或裝置。如需深入了解 CA 部署規劃和 CA 階層設計，包括恢復能力、跨區域複寫、在組織中共用 CA 等規劃，請參閱[規劃您的 AWS Private CA 部署](#)。

實作步驟

1. 確定使用案例所需的相關 AWS 服務：

- 許多使用案例都可以使用 [AWS Certificate Manager](#) 來利用現有的 AWS 公有金鑰基礎設施。ACM 可以用於為 Web 伺服器、負載平衡器或公開信任憑證的其他用途部署 TLS 憑證。
 - 當您需要建立自己的私有憑證授權機構階層或需要存取可匯出的憑證時，考慮 [AWS Private CA](#)。然後，ACM 可用於使用 AWS Private CA 發行 [許多類型的終端實體憑證](#)。
 - 針對必須為嵌入式物聯網 (IoT) 裝置大規模佈建憑證的使用案例，請考慮 [AWS IoT Core](#)。
 - 請考慮在 [Amazon API Gateway](#) 或 [Application Load Balancer](#) 等服務中使用原生 mTLS 功能。
2. 盡可能實作自動憑證續約：
 - 使用 [ACM 受管續約](#) ACM 發行的憑證以及整合式 AWS 受管服務。
 3. 建立日誌記錄和稽核記錄：
 - 啟用 [CloudTrail 日誌](#) 以追蹤對持有憑證授權機構之帳戶的存取權。請考慮在 CloudTrail 中設定日誌檔案完整性驗證，以驗證日誌資料的真實性。
 - 定期產出並檢閱 [稽核報告](#)，其中列出您的私有 CA 發行或撤銷的憑證。這些報告可以匯出到 S3 儲存貯體。
 - 部署私有 CA 時，您也需要建立 S3 儲存貯體來儲存憑證撤銷清單 (CRL)。如需有關根據工作負載需求設定此 S3 儲存貯體的指引，請參閱 [規劃憑證撤銷清單 \(CRL\)](#)。

資源

相關的最佳實務：

- [SEC02-BP02 使用臨時憑證](#)
- [SEC08-BP01 實作安全金鑰管理](#)
- [SEC09-BP03 驗證網路通訊](#)

相關文件：

- [如何在 AWS 中託管和管理整個私有憑證基礎設施](#)
- [如何保護用於汽車和製造領域的企業規模 ACM 私有 CA 階層](#)
- [私有 CA 最佳實務](#)
- [如何使用 AWS RAM 共用您的 ACM 私有 CA 跨帳戶](#)

相關影片：

- [啟動 AWS Certificate Manager 私有 CA \(研討會\)](#)

相關範例：

- [私有 CA 研討會](#)
- [IOT Device Management 研討會](#) (包括裝置佈建)

相關工具：

- [要使用 AWS Private CA 的 Kubernetes cert-manager 外掛程式](#)

SEC09-BP02 強制傳輸中加密

根據您組織的政策、法規義務和標準強制已定義的加密需求，以符合組織、法律和合規上的要求。只有在虛擬私有雲端 (VPC) 以外傳輸敏感資料時才使用加密協定。加密有助於保持資料完整性，甚至當資料傳輸於不受信任的網路。

預期成果：您將資源與網際網路之間的網路流量加密，以減少對資料的未經授權存取。您可以根據您的安全要求，加密內部 AWS 環境中的網路流量。您使用安全的 TLS 協定和密碼套件，為所有傳輸中的資料加密。

常見的反模式：

- 使用 SSL、TLS 和其他套件元件已棄用的版本 (例如，SSL v3.0、1024 位元 RSA 金鑰和 RC4 密碼)。
- 允許未加密的 (HTTP) 流量來往面向公眾的資源。
- 未監控 X.509 憑證並在到期前更換。
- 對 TLS 使用自我簽署的 X.509 憑證。

未建立此最佳實務時的曝險等級：高

實作指引

AWS 服務提供使用 TLS 的 HTTPS 端點以進行通訊，在與 AWS API 通訊時提供傳輸中加密。不安全的 HTTP 通訊協定可以在虛擬私有雲端 (VPC) 中，透過使用安全群組來進行稽核和封鎖。在 Amazon CloudFront 或 [Application Load Balancer](#) 中，HTTP 請求也可以[自動重新導向至 HTTPS](#)。您可以使用 [Amazon Simple Storage Service \(Amazon S3\) 儲存貯體政策](#) 來限制透過 HTTP 上傳物件的能力，並有效地強制執行使用 HTTPS，將物件上傳至儲存貯體。您可以全權控制您的運算資源，以在各個服務中實作傳輸中加密。此外，還可以從外部網路或 [AWS Direct Connect](#) 使用 VPN 連線功能進入 VPC，

加速流量加密。確認您的用戶端使用至少 TLS 1.2 對 AWS APIs 進行呼叫，因為 [AWS TLS 的早期版本已於 2024 年 2 月棄用](#)。建議您使用 TLS 1.3。如果您有傳輸中加密的特殊要求，您可以在 AWS Marketplace 中找到可用的第三方解決方案。

實作步驟

- 強制傳輸中加密：您定義的加密要求應符合最新標準和最佳實務，並僅允許採用安全協定。例如，設定安全群組，僅允許 HTTPS 協定連至 Application Load Balancer 或 Amazon EC2 執行個體。
- 在邊緣服務中設定安全協定：[使用 Amazon CloudFront 設定 HTTPS](#)，並使用[適用於您的安全狀態和使用案例的安全設定檔](#)。
- 使用 [VPN 進行外部連線](#)：考慮使用 IPsec VPN，保護點對點或網路對網路連線，以協助提供資料隱私和完整性。
- 在負載平衡器中設定安全協定：選取安全政策，以提供要連接到接聽程式的用戶端所支援的最強固的密碼套件。[為 Application Load Balancer 建立 HTTPS 接聽程式](#)。
- 在 Amazon Redshift 中設定安全協定：將您的叢集設定為要求 [Secure Socket Layer \(SSL\) 或 Transport Layer Security \(TLS\) 連線](#)。
- 設定安全協定：檢閱 AWS 服務文件以確定傳輸中加密功能。
- 設定上傳至 Amazon S3 儲存貯體時的安全存取：使用 Amazon S3 儲存貯體政策控制對資料[強制安全存取](#)。
- 考慮使用 [AWS Certificate Manager](#)：ACM 可讓您佈建、管理和部署 TLS 憑證與 AWS 服務搭配使用。
- 考慮使用 [AWS Private Certificate Authority](#) 滿足私有 PKI 需求：AWS Private CA 可讓您建立私有憑證授權機構 (CA) 階層來發行可用於建立已加密 TLS 通道的終端實體 X.509 憑證。

資源

相關文件：

- [將 HTTPS 與 CloudFront 搭配使用](#)
- [使用 AWS Virtual Private Network 將 VPC 連線至遠端網路](#)
- [為 Application Load Balancer 建立 HTTPS 接聽程式](#)
- [教學課程：在 Amazon Linux 2 上設定 SSL/TLS](#)
- [使用 SSL/TLS 加密與資料庫執行個體的連線](#)
- [設定連線的安全選項](#)

SEC09-BP03 驗證網路通訊

使用支援身分驗證的協定 [如 Transport Layer Security (TLS) 或 IPsec] 來驗證通訊的身分。

設計工作負載，以在每當服務、應用程式或使用者之間進行通訊時，使用安全、經驗證的網路協定。使用支援驗證和授權的網路協定可提供更強大的網路流量控制能力，並減少未經授權存取所造成的影響。

預期成果：設計出工作負載，讓其有明確定義的服務間資料平面和控制平面流量流程。在技術允許的情況下，流量流程要使用經過驗證和加密的網路協定。

常見的反模式：

- 工作負載內有未經加密或驗證的流量流程。
- 在多個使用者或實體之間重複使用驗證憑證。
- 僅仰賴網路控制做為存取控制機制。
- 建立自訂驗證機制，而非仰賴產業標準的驗證機制。
- 服務元件或 VPC 中的其他資源之間有過於寬鬆的流量。

建立此最佳實務的優勢：

- 將未經授權存取所造成的影響範圍限制在工作負載的某個部分。
- 提供只會由已驗證實體執行動作的更高層級保證。
- 透過清楚地定義並強制執行預期的資料傳輸介面來改善服務的去耦。
- 透過請求歸因和明確定義的通訊介面，增強監控、日誌記錄和事件回應。
- 結合網路控制與驗證和授權控制，為您的工作負載提供深度防禦。

未建立此最佳實務時的曝險等級：低

實作指引

您工作負載的網路流量模式可分為兩個類別：

- 東西流量代表構成工作負載的服務之間的流量流程。
- 南北流量代表工作負載和取用者之間的流量流程。

雖然加密南北流量是常見的做法，但是使用經過驗證的協定來保護東西流量則較不常見。現代安全實務的建議是，單靠網路設計並無法讓兩個實體之間建立信任的關係。當兩個服務可能位於一個共通的網路邊界內時，最佳實務仍是對這些服務之間的通訊進行加密、驗證和授權。

舉例來說，無論請求來自哪個網路，AWS 服務 API 都會使用 [AWS 第 4 版簽署程序 \(SigV4\)](#) 簽署協定來驗證呼叫者。此驗證可確保 AWS API 可以驗證發出動作請求的身分，該身分接著可與政策結合來作出授權決策，確定是否應允許該動作。

[Amazon VPC Lattice](#) 和 [Amazon API Gateway](#) 等服務可讓您使用相同的 SigV4 簽署協定，為自己的工作負載中的東西流量新增驗證和授權功能。如果 AWS 環境以外的資源需要與要求進行 SigV4 型驗證和授權的服務進行通訊，您可以在非 AWS 資源上使用 [AWS Identity and Access Management \(IAM\) Roles Anywhere](#) 來取得暫時 AWS 憑證。使用這些憑證，便可透過 SigV4 簽署服務請求以授權存取。

用於驗證東西流量的另一種常見機制是 TLS 相互驗證 (mTLS)。許多物聯網 (IoT)、企業對企業應用程式和微型服務都使用 mTLS，透過使用用戶端和伺服器端 X.509 憑證來驗證 TLS 通訊兩端的身分。這些憑證可由 AWS Private Certificate Authority (AWS Private CA) 發行。您可以使用 [Amazon API Gateway](#) 等服務，為工作負載之間或工作負載內部的通訊提供 mTLS 驗證。[Application Load Balancer](#) 也支援內部或外部工作負載的 mTLS。雖然 mTLS 會為 TLS 通訊的兩端提供驗證資訊，但不提供授權機制。

最後，OAuth 2.0 和 OpenID Connect (OIDC) 是兩種常用於控制使用者對服務存取行為的協定，但現在也變成服務對服務流量的熱門協定。API Gateway 會提供 [JSON Web Token \(JWT\) 授權器](#)，可讓工作負載使用 OIDC 或 OAuth 2.0 身分提供者所發行的 JWT 來限制 API 路由的存取。OAuth2 的範圍可做為基本授權決策的來源，但仍需要在應用程式層實作授權檢查，而且單靠 OAuth2 範圍並無法支援更複雜的授權需求。

實作步驟

- 定義並記錄您的工作負載網路流程：實作深度防禦策略的第一步是定義工作負載的流量流程。
 - 建立可清楚定義構成工作負載的不同服務間資料傳輸方式的資料流程圖。此圖是透過已驗證的網路通道強制執行這些流程的第一步。
 - 在開發和測試階段檢測您的工作負載，以驗證資料流程圖是否準確反映工作負載在執行時期的行為。
 - 資料流程圖在執行威脅建模練習時也很有用，如 [SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施](#) 中所述。
- 建立網路控制：考慮用來建立與資料流程一致的網路控制的 AWS 功能。雖然網路邊界不應成為唯一的安全控制，但其可在深度防禦策略中提供一個保護層，以保護您的工作負載。

- 使用[安全群組](#)建立定義和限制資源之間的資料流程。
- 考慮使用 [AWS PrivateLink](#) 與支援 AWS PrivateLink 的 AWS 和第三方服務進行通訊。透過 AWS PrivateLink 介面端點傳送的資料會保留在 AWS 網路骨幹內，不會周遊公用網際網路。
- 在工作負載中跨服務實作驗證和授權：選擇最適合用來在您工作負載中提供經驗證加密流量的 AWS 服務集。
 - 考慮用來保護服務對服務通訊的 [Amazon VPC Lattice](#)。VPC Lattice 可以使用 [SigV4 驗證結合驗證政策](#)來控制服務對服務的存取。
 - 對於使用 mTLS 的服務對服務通訊，請考慮 [API Gateway](#)、[Application Load Balancer](#)。 [AWS Private CA](#) 可用於建立能夠發行憑證以與 mTLS 搭配使用的私有 CA 階層。
 - 與使用 OAuth 2.0 或 OIDC 的服務進行整合時，請考慮[使用 JWT 授權器的 API Gateway](#)。
 - 對於工作負載和 IoT 裝置之間的通訊，請考慮 [AWS IoT Core](#)，它提供了幾種網路流量加密和驗證選項。
- 監控未經授權的存取：持續監控是否有意外的通訊管道、嘗試存取受保護資源的未經授權主體，以及其他不當的存取模式。
 - 如果使用 VPC Lattice 來管理服務的存取，請考慮啟用和監控 [VPC Lattice 存取日誌](#)。這些存取日誌包括請求方實體的資訊、包括來源和目的地 VPC 在內的網路資訊，以及請求中繼資料。
 - 考慮啟用 [VPC 流程日誌](#)來擷取網路流程上的中繼資料，並定期審查是否有異常狀況。
 - 如需更多有關規劃、模擬和應對安全事件的指引，請參閱 [AWS 安全事件回應指南](#)和 AWS Well Architected Framework 安全支柱的[事件回應](#)章節。

資源

相關的最佳實務：

- [SEC03-BP07 分析公有和跨帳戶存取權](#)
- [SEC02-BP02 使用臨時憑證](#)
- [SEC01-BP07 使用威脅模型識別威脅並優先考慮緩解措施](#)

相關文件：

- [評估用來保護 Amazon API Gateway API 的存取控制方法](#)
- [為 REST API 設定相互 TLS 驗證](#)
- [如何使用 JWT 授權器保護 API Gateway HTTP 端點](#)
- [使用 AWS IoT Core 憑證提供者來授權 AWS 服務的直接呼叫](#)

- [AWS 安全事件回應指南](#)

相關影片：

- [AWS re:invent 2022：VPC Lattice 簡介](#)
- [AWS re:invent 2020：針對 AWS 上 HTTP API 的無伺服器 API 驗證](#)

相關範例：

- [Amazon VPC Lattice 研討會](#)
- [零信任第 1 集 - Phantom Service Perimeter 研討會](#)

事件回應

即使採用了成熟的預防和偵測控制，您的組織仍應實作機制，以回應並緩和和安全事故的潛在影響。您的準備工作會大大地影響團隊在事故發生時能否有效運作，以隔離、遏制問題並進行鑑識，以及將營運恢復到已知的良好狀態。在安全事故發生前先備妥工具和存取權，然後在演練日期間定期練習事故回應，有助於確保您能夠復原，同時盡量減少業務中斷。

主題

- [AWS 事件回應的各個層面](#)
- [雲端回應的設計目標](#)
- [準備](#)
- [作業](#)
- [事後處理](#)

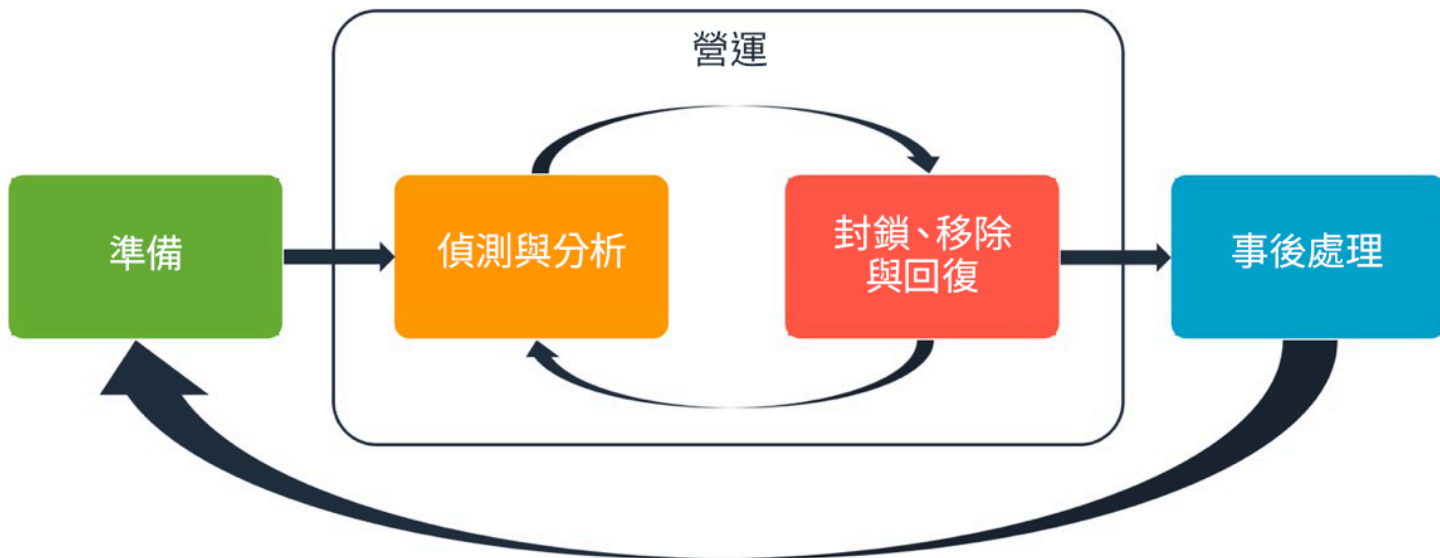
AWS 事件回應的各個層面

組織內的所有 AWS 使用者都應該對安全事件回應程序具有基本的了解，而安全人員應了解如何回應安全問題。教育、培訓和體驗是成功的雲端事件回應計畫必不可少的一環，最好能預先實施，以因應發生安全事件的情況。雲端中成功的事件回應計畫的基礎在於準備、操作和事後處理。

若要分別了解這三個層面，請參考下列說明：

- **準備**：藉由啟用偵測控制並確認具備存取必要工具和雲端服務的適當權限，讓事件回應團隊做好準備，以便偵測和回應 AWS 內的事件。此外，備妥必要的程序手冊 (包括手動和自動)，以確認能夠做出可靠且一致的回應。
- **操作**：依照 NIST 事件回應階段，對安全事件和可能的事件進行操作：偵測、分析、遏制、根除和復原。
- **事後處理**：反覆執行安全事件和模擬的結果，以改善回應的有效性，增加從回應與調查中獲得的價值，並進一步降低風險。您必須從事件中學習，並能夠確實實施後續改進。

下圖顯示這些方面的流程，與先前提到的 NIST 事件回應生命週期一致，但涵蓋了包含遏制、根除和復原的偵測及分析操作。



AWS 事件回應的各個層面

雲端回應的設計目標

雖然事件回應的一般程序和機制，例如 [NIST SP 800-61 Computer Security Incident Handling Guide](#) 中所定義的仍然可行，但我們鼓勵您評估在雲端環境中回應安全事件方面的下列特定設計目標：

- 建立回應目標：與利害關係人、法律顧問和組織領導階層合作，訂定回應事件的目標。共同目標包括遏制和緩解問題、復原受影響的資源、保留資料供鑑識使用、恢復已知的安全操作，以及最終從事件中學習經驗。
- 使用雲端回應：在雲端內事件發生和資料之處實作回應模式。
- 了解您擁有什麼及需要什麼：複製日誌、資源、快照和其他證據，並儲存在專門用於回應的集中式雲端帳戶中以便保留。運用標籤、中繼資料和機制，強制執行保留政策。您需要了解自己使用哪些服務，然後確定調查這些服務的需求。您也可以使用標籤來協助您了解自己的環境。
- 使用重新部署機制：如果安全異常可能歸因於組態錯誤，修復的方法可能就是透過使用適當的組態，重新部署資源以移除變異那麼簡單。如果發現可能是遭到入侵，則務必確認您的重新部署包含可成功緩解根本原因的措施。
- 盡可能自動化：當問題出現或事件重複發生時，請建立機制，以程式設計方式分類並回應常見的事件。對於自動化不足以因應的獨特、複雜或敏感事件，則採取真人回應。
- 選擇可擴展的解決方案：努力符合組織所採行雲端運算方法的可擴展性。實作能夠因應您的環境調整的偵測和回應機制，以便有效縮短偵測與回應之間的時間。
- 了解並改善程序：主動找出流程、工具或人員當中的落差，並實施計畫來彌補落差。模擬是找出落差並改善流程的安全方法。

這些設計目標可提醒您審核架構實作，以確認能夠進行事件回應和威脅偵測。當您規劃雲端實作時，請考慮回應事件，最好是採用鑑識上可靠的回應方法。在某些情況下，這表示您可能會針對這些回應任務設定多個組織、帳戶和工具。這些工具和功能應透過部署管道提供給事件回應人員使用。它們不應處於靜態，否則可能造成更大的風險。

準備

為因應事件做好準備，對於須及時並有效回應的事件來說至關重要。準備工作橫跨三個領域：

- 人員：讓員工做好因應安全事件的準備，其中包括找出事件回應的相關利害關係人，並進行事件回應和雲端技術的培訓。
- 流程：備妥處理安全事件的流程，其中包括記錄架構、制定完整的事件回應計畫，以及製作程序手冊以便一致回應安全事件。
- 技術：備妥因應安全事件的技術，其中包括設定存取權、彙總和監控必要的日誌、實作有效的提醒機制，以及開發回應和調查功能。

這三個領域對於有效回應事件來說同樣重要。缺少任一個領域，事件回應計畫便不完整或無法發揮效用。您需要讓人員、流程和技術三者緊密整合，才能做好因應事件的準備。

最佳實務

- [SEC10-BP01 識別關鍵人員和外部資源](#)
- [SEC10-BP02 制定事件管理計畫](#)
- [SEC10-BP03 準備鑑識功能](#)
- [SEC10-BP04 開發和測試安全事件回應程序手冊](#)
- [SEC10-BP05 預先佈建存取權](#)
- [SEC10-BP06 部署前工具](#)
- [SEC10-BP07 執行模擬](#)

SEC10-BP01 識別關鍵人員和外部資源

識別可以協助您的組織回應事件的內部和外部人員、資源及法律義務。

預期成果：您備有一份關鍵人員名單，包含其聯絡資訊，以及他們在回應安全事件時扮演的角色。您可定期審查並更新此資訊，以在內部和外部工具中反映人員變更。您在記錄此資訊時考量所有第三方服務供應商和廠商，包括安全合作夥伴、雲端供應商，以及軟體即服務 (SaaS) 應用程式。在安全事件

期間，人員會負起適當層級的責任、得知適當的關聯內容，並擁有適當的存取權能夠做出回應和進行復原。

常見的反模式：

- 回應安全事件時，未隨時備妥包含聯絡資訊、角色和職責的最新關鍵人員名單。
- 假設每個人都了解回應事件和從事件復原時的人員、相依關係、基礎設施和解決方案。
- 沒有能夠呈現主要基礎設施或應用程式設計的文件或知識儲存庫。
- 未制定適當的新員工上任流程，導致他們無法有效地參與安全事件回應工作，例如進行事件模擬。
- 在安全事件期間，當關鍵人員暫時聯絡不到或無法回應時，沒有向上呈報的管道。

建立此最佳實務的優勢：此實務可減少發生事件時，花在識別正確人員和其角色的權責劃分和回應時間。隨時備妥一份最新的關鍵人員及其角色的名單，您就能找到正確的人員進行權責劃分並從事件中復原，藉此在發生事件時盡量減少時間浪費。

未建立此最佳實務時的風險暴露等級：高

實作指引

識別組織中的關鍵人員：隨時備妥組織內應對特定事件所需的人員聯絡名單。發生人員變動 (例如組織變動、晉升和團隊變動) 時，定期審查並更新此資訊。這對於事件管理者、事件回應者和通訊負責人等關鍵角色尤其重要。

- 事件管理者：事件管理者在事件回應期間具有整體授權。
- 事件回應者：事件回應者負責調查和補救活動。這些人員可能根據事件類型而有所不同，但通常是負責受影響應用程式的開發人員和營運團隊。
- 通訊負責人：通訊負責人負責內部和外部溝通，特別是與公家機關、監管機構和客戶之間的溝通。
- 上線程序：定期培訓新進員工，並讓他們上線操作，讓他們具備必要的技能和知識，以有效地為事件回應工作做出貢獻。將模擬和實作練習納入以當成上線程序的一部分，以催促他們做好準備
- 主題專家 (SME)：如果是分散式和自主團隊，我們建議您組成 SME 團隊來負責關鍵任務工作負載。他們負責提供對事件所涉及關鍵工作負載的操作和資料分類的深入洞悉。

資料表格式範例：

	Role	Name	Contact Information	Responsibilities
1	---	---	---	---

```
2 | Incident Manager | Jane Doe | jane.doe@example.com | Overall authority during  
response |  
3 | Incident Responder | John Smith | john.smith@example.com | Investigation and  
remediation |  
4 | Communications Lead | Emily Johnson | emily.johnson@example.com | Internal and  
external communications |  
5 | Communications Lead | Michael Brown | michael.brown@example.com | Insights on  
critical workloads |
```

請考慮使用 [AWS Systems Manager Incident Manager](#) 功能來擷取主要聯絡人、定義回應計畫、自動排定待命時間表，以及制定向上呈報計畫。自動排定待命時間表並輪替所有人員，藉此將工作負載的責任分散給其負責人。這樣有助於建立良好的實務，例如，發出相關的指標和日誌，以及定義對工作負載至關重要的警示閾值。

識別外部合作夥伴：企業使用由獨立軟體開發廠商 (ISV)、合作夥伴和子承包商建置的工具，為其客戶打造出獨特的解決方案。邀集上述多方的關鍵人員參與，他們可以協助回應事件並從中復原。我們建議您註冊適當層級的支援，以便透過支援案例迅速與 AWS 主題專家取得聯繫。請考慮針對工作負載的所有關鍵解決方案提供者做出類似的安排。有些安全事件會促使公開上市公司通知相關的公家機關和監管機構有關事件的情形和影響。維護並更新相關部門和負責人的聯絡資訊。

實作步驟

1. 設定事件管理解決方案。
 - a. 考慮在您的安全工具帳戶中部署 Incident Manager。
2. 在事件管理解決方案中定義聯絡人。
 - a. 針對每個聯絡人至少定義兩種類型的聯絡管道 (例如 SMS、電話或電子郵件)，以確保在發生事件時能夠與他們取得聯繫。
3. 定義回應計畫。
 - a. 識別發生事件時最合適參與的聯絡人。定義符合要參與之人員角色 (而非個別聯絡人) 的向上呈報計畫。考慮納入可負責通知外部實體的聯絡人，即使他們並未直接參與事件解決工作。

資源

相關的最佳實務：

- [OPS02-BP03 已為營運活動識別負責其效能的擁有者](#)

相關文件：

- [AWS 安全事件回應指南](#)

相關範例：

- [AWS 客戶程序手冊架構](#)
- [準備和回應 AWS 環境中的安全事件](#)

相關工具：

- [AWS Systems Manager Incident Manager](#)

相關影片：

- [Amazon 在開發期間採取的安全方法](#)

SEC10-BP02 制定事件管理計畫

為事件回應制定的第一份文件是事件回應計畫。事件回應計畫應是您事件回應計畫和策略的基礎。

建立此最佳實務的優勢：開發全面且明確定義的事件回應程序，是成功且可擴展的事件回應計畫的關鍵。當安全事件發生時，明確的步驟和工作流程可協助您及時因應。您可能已具備現有的事件回應程序。無論您目前的狀態為何，都必須定期更新、重複執行和測試事件回應程序。

未建立此最佳實務時的曝險等級：高

實作指引

事件管理計畫對於回應、減輕安全事件所造成潛在影響並從中復原而言至關重要。事件管理計畫是結構清晰的程序，可及時識別、修復和回應安全事件。

雲端有許多在內部部署環境中所見的相同營運角色和需求。您在建立事件管理計畫時，必須將與業務成果及合規要求最相符的回應及復原策略納入考量。例如，若您在 AWS 中操作的工作負載符合美國的 FedRAMP，那麼請遵循《[NIST SP 800-61 電腦安全處理指南](#)》中的建議。同樣的，當您操作儲存個人身分識別資訊 (PII) 的工作負載時，請考慮如何保護和回應與資料落地和使用相關的問題。

為 AWS 中的工作負載建置事件管理計畫時，請從 [AWS 共同責任模式](#) 開始，以便建置事件回應的深度防禦方法。在此模型中，AWS 會管理雲端的安全，但維護雲端的安全是您的責任。此表示您保有控制權，並對您選擇實作的安全控制項負責。[AWS 安全事件回應指南](#) 詳細說明在建置以雲端為中心的事件管理計畫時的重要概念和基礎指引。

有效的事件管理計畫必須經過持續的反覆測試，以與您的雲端維運目標保持同步。在您建立和制定事件管理計畫時，請考慮使用以下詳述的實作計畫。

實作步驟

1. 定義您組織內處理安全事件的角色和責任。這應涉及各部門的代表，包括：
 - 人力資源 (HR)
 - 執行團隊
 - 法務部門
 - 應用程式擁有者和開發人員 (主題專家或 SME)
2. 明確概述事件進行期間誰應負責、當責、受諮詢和被告知 (RACI)。建立 RACI 圖表以促進快速直接的溝通，並明確概述事件不同階段的領導。
3. 在事件期間讓應用程式擁有者和開發人員 (SME) 參與，因為他們可以提供有價值的資訊和內容，有助於衡量影響範圍。與這些 SME 建立關係，並在實際事件發生之前與他們演練事件回應情境。
4. 讓受信任的合作夥伴或外部專家參與調查或回應程序，因為他們可以提供額外的專業知識和觀點。
5. 讓您的事件管理計畫和角色符合任何管理組織的當地法規或合規要求。
6. 定期練習和測試您的事件回應計畫，並納入所有定義的角色和責任。這有助於簡化程序，並確認您對於安全事件能有協調且有效的回應。
7. 定期審查和更新角色、責任和 RACI 圖表，或隨著您的組織結構或需求進行變更。

了解 AWS 回應團隊和支援

- AWS 支援
 - [支援](#) 提供各種方案，可支援您運用各種工具與專業知識，協助您的 AWS 解決方案獲得成功並正常運作。如果您需要技術支援和更多資源以協助規劃、部署和最佳化 AWS 環境，您可以選取最符合您 AWS 使用案例的支援計畫。
 - 考慮以 AWS Management Console 中的 [支援中心](#) (需要登入) 做為中心聯絡窗口，以取得影響 AWS 資源的問題所需的支援。對支援的存取由 AWS Identity and Access Management 控制。如需有關取得支援功能存取權的詳細資訊，請參閱 [支援入門](#)。
- AWS 客戶事件回應團隊 (CIRT)
 - AWS 客戶事件回應團隊 (CIRT) 是一個專門的全天候全球 AWS 團隊，在客戶端的有效安全事件期間為客戶提供支援 - [AWS 共同責任模式](#)。

- 當 AWS CIRT 支援您時，他們會為 AWS 上的有效安全事件提供分類和復原方面的協助。他們可透過使用 AWS 服務日誌協助進行根本原因分析，並為您提供復原的建議。他們也可提供安全建議和最佳實務，以協助您避免事後發生安全事件。
- AWS 客戶可以透過 [支援 案例](#) 洽詢 AWS CIRT。
- DDoS 回應支援
 - AWS 提供 [AWS Shield](#)，其中包含受管的分散式拒絕服務 (DDoS) 保護服務，可為執行於 AWS 的 Web 應用程式提供保護。Shield 提供一律開啟的偵測和自動內嵌緩解措施，可將應用程式停機時間和延遲降至最低，讓您無需聯絡 支援 即可享有 DDoS 保護。Shield 有兩個層級：AWS Shield Standard 和 AWS Shield Advanced。若要了解這兩個層級的差異，請參閱 [Shield 功能文件](#)。
- AWS Managed Services (AMS)
 - [AWS Managed Services \(AMS\)](#) 可讓您持續管理 AWS 基礎設施，讓您專注在自己的應用程式上。透過實作維護基礎設施的最佳實務，AMS 有助於降低營運開銷和風險。AMS 可自動化常見的活動，例如，變更請求、監控、修補程式管理、安全性和備份服務，而且提供佈建、執行和支援基礎設施的完整生命週期服務。
 - AMS 負責部署安全偵測控制套件，並提供全年無休的第一線提醒應變措施。提醒啟動時，AMS 會依照一組標準的自動化和手動程序手冊來驗證回應的一致性。這些程序手冊會在上線期間與 AMS 客戶共享，讓他們能夠透過 AMS 來制定和協調應變措施。

制定事件回應計畫

事件回應計畫應是您事件回應計畫和策略的基礎。事件回應計畫應納入正式文件中。事件回應計畫通常包含下列章節：

- 事件回應團隊概觀：概述事件回應團隊的目標和職能。
- 角色和責任：列出事件回應利害關係人，並詳細說明他們在事件發生時的角色。
- 通訊計畫：詳細說明聯絡資訊，以及您在事件期間要如何進行通訊。
- 備份通訊方式：最佳實務是將頻外通訊做為事件通訊的備用方法。舉例來說，AWS Wickr 就是提供安全頻外通訊通道的應用程式。
- 事件回應的階段和應採取的行動：列舉事件回應的階段 (例如偵測、分析、消除、抑制及復原)，包括要在這些階段中採取的高階動作。
- 事件嚴重性和優先順序定義：詳細說明如何分類事件的嚴重性、如何排定事件的優先順序，以及嚴重性定義對於呈報程序有何影響。

儘管不同規模和產業的公司都會有這些章節，但每個組織的事件回應計畫都是獨一無二的。您必須建立最適合貴組織的事件回應計畫。

資源

相關的最佳實務：

- [SEC04 偵測](#)

相關文件：

- [AWS 安全事件回應指南](#)
- [NIST：電腦安全事件處理指南](#)

SEC10-BP03 準備鑑識功能

在安全事件發生之前，將開發鑑識功能納入考量，以協助安全事件調查。

未建立此最佳實務時的曝險等級：中

傳統內部部署鑑識的概念適用於 AWS。如需在 中開始建置鑑識能力的重要資訊 AWS 雲端，請參閱 [中的鑑識調查環境策略 AWS 雲端](#)。

設定取證的環境和 AWS 帳戶 結構後，請定義在四個階段有效執行取證合理方法所需的技術：

- 集合：收集相關 AWS 日誌，例如 AWS CloudTrail AWS Config、VPCFlow Logs 和主機層級日誌。收集可用時受影響 AWS 資源的快照、備份和記憶體傾印。
- 檢查：檢查透過擷取和評估相關資訊所收集的資料。
- 分析：分析收集的資料，以了解事件並從中得出結論。
- 報告：呈現分析階段所產生的資訊。

實作步驟

準備鑑識環境

[AWS Organizations](#) 可協助您在資源成長和擴展時集中管理和管理 AWS 環境 AWS。AWS 組織會合併您的，AWS 帳戶 以便您以單一單位管理它們。您可以使用組織單位（OUs）將帳戶分組在一起，以單一單位進行管理。

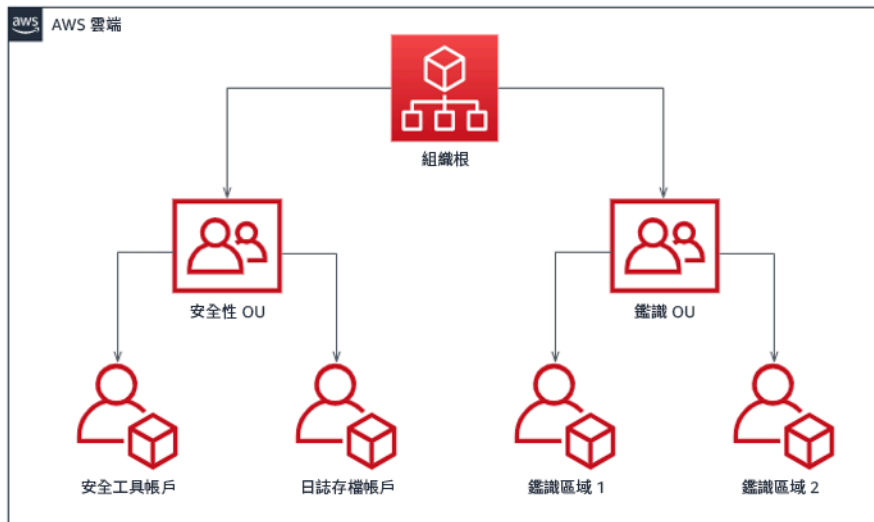
對於事件回應，擁有支援事件回應功能的 AWS 帳戶結構很有幫助，其中包括安全 OU 和鑑識 OU。在安全性 OU 中，您應該擁有下列項目的帳戶：

- 日誌封存：在許可 AWS 帳戶 有限的日誌封存中彙總日誌。
- 安全工具：在安全工具 中集中安全服務 AWS 帳戶。此帳戶會以安全性服務的委派系統管理員身分運作。

在鑑識 OU 中，您可以選擇為營運所在的每個區域實作一或多個鑑識帳戶，具體視哪個區域最適合您業務和營運模式而定。如果您為每個區域建立鑑識帳戶，則可以封鎖在該區域之外建立 AWS 資源，並降低資源複製到非預期區域的風險。例如，如果您只在美國東部 (維吉尼亞北部) 區域 (us-east-1) 和美國西部 (奧勒岡) (us-west-2) 進行營運，則鑑識 OU 中會有兩個帳戶：一個用於 us-east-1，另一個用於 us-west-2。

您可以為 AWS 帳戶 多個區域建立鑑識。在將 AWS 資源複製到該帳戶時，您應該小心謹慎，以確認您是否符合資料主權要求。佈建新帳戶需要一些時間，因此必須在事件之前建立和檢測鑑識帳戶，以便回應者能夠有效地使用這些帳戶進行回應。

下圖顯示範例帳戶結構，包括具有每個區域鑑識帳戶的鑑識 OU：



事件回應的每個區域帳戶結構

擷取備份和快照

設定重要系統和資料庫的備份，對於從安全事件中復原和鑑識用途非常重要。備份就緒後，您可以將系統還原到先前的安全狀態。在上 AWS，您可以拍攝各種資源的快照。快照可為您提供 point-in-time 這些資源的備份。有許多 AWS 服務可以支援您進行備份和復原。如需有關這些備份和復原之服務和方法的詳細資訊，請參閱 [備份和復原方案指引](#) 和 [使用備份從安全事件中復原](#)。

尤其是當涉及勒索軟體等情況時，務必確保備份是否有充足的保護。如需有關保護備份的指引，請參閱在 [AWS 中保護備份的 10 大安全最佳實務](#)。除了確保備份的安全之外，您還應該定期測試備份和還原程序，以確認您現有的技術和程序是否如預期般運作。

自動化鑑識

在安全事件期間，您的事件回應團隊必須能夠快速收集和分析證據，同時在事件周圍的期間內維持準確性（例如擷取與特定事件或資源相關的日誌，或收集 Amazon EC2 執行個體的記憶體傾印）。事件回應團隊手動收集相關證據既具挑戰性又耗時，尤其是範圍遍及大量執行個體和帳戶時。此外，手動收集可能容易出現人為錯誤。基於這些原因，您應盡可能開發和實作鑑識的自動化。

AWS 為鑑識提供了許多自動化資源，這些資源列於下列資源區段中。這些資源是我們已開發和客戶已實作的鑑識模式範例。雖然這些範例在一開始可能是有用的參考架構，但請根據環境、需求、工具和鑑識程序，考慮是否加以修改或建立新的鑑識自動化模式。

資源

相關文件：

- [AWS 安全事件回應指南 - 開發鑑識能力](#)
- [AWS 安全事件回應指南 - 鑑識資源](#)
- [中的鑑識調查環境策略 AWS 雲端](#)
- [如何在 中自動化鑑識磁碟收集 AWS](#)
- [AWS 規範指南 - 自動化事件回應和鑑識](#)

相關影片：

- [自動化事件回應和鑑識](#)

相關範例：

- [自動化事件回應和鑑識架構](#)
- [適用於 Amazon 的自動 Forensics Orchestrator EC2](#)

SEC10-BP04 開發和測試安全事件回應程序手冊

準備事件回應流程的關鍵部分是制定程序手冊。事件回應程序手冊提供方案指引，以及安全事件發生時應遵循的步驟。提供清晰的結構和步驟簡化了回應的複雜度並減少人為錯誤的可能性。

未建立此最佳實務時的曝險等級：中

實作指引

應針對事件案例建立程序手冊，例如：

- 預期事件：應針對您預期的事件建立程序手冊。這包括拒絕服務 (DoS)、勒索軟體和憑證入侵等威脅。
- 已知安全調查結果或提醒：應針對已知安全調查結果和提醒 (例如 Amazon GuardDuty 調查結果) 建立執行手冊。當您收到 GuardDuty 調查結果時，手冊應提供明確的步驟，以避免處理不當或忽略提醒。如需更多修復相關資訊和指引，請參閱[修復 GuardDuty 發現的安全問題](#)。

程序手冊應包含安全分析師應完成的技術步驟，以便充分調查和應對潛在的安全事件。

實作步驟

要納入程序手冊的項目包括：

- 程序手冊概觀：這份程序手冊可處理哪些風險或事件？程序手冊的目標是什麼？
- 先決條件：此事件案例需要哪些日誌、偵測機制和自動化工具？預期的通知是什麼？
- 溝通和向上呈報資訊：誰參與其中，其聯絡資訊為何？每個利害關係人的責任是什麼？
- 回應步驟：在事件回應的各個階段，應採取哪些戰術步驟？分析師應該執行哪些查詢？應該執行哪些程式碼以達到預期的成果？
 - 偵測：事件的偵測方式為何？
 - 分析：判斷影響範圍的方式為何？
 - 包含：隔離事件以限制範圍的方式為何？
 - 根除：將威脅從環境中移除的方式為何？
 - 復原：受影響的系統或資源重新投入生產環境的方式為何？
- 預期成果：執行查詢和程式碼後，程序手冊的預期結果是什麼？

資源

相關 Well-Architected 的最佳實務：

- [SEC10-BP02 - 制定事件管理計畫](#)

相關文件：

- [事件回應程序手冊的架構](#)
- [制定您自己的事件回應程序手冊](#)
- [事件回應程序手冊範例](#)
- [使用 Jupyter 程序手冊和 CloudTrail Lake 建置 AWS 事件回應執行手冊](#)

SEC10-BP05 預先佈建存取權

確認事件回應者具有在 AWS 中預先佈建的正確存取權限，以縮短調查直至復原所需的時間。

常見的反模式：

- 使用事件回應的根帳戶。
- 更改現有的帳戶。
- 當提供即時權限提升時直接操控 IAM 許可。

未建立此最佳實務時的曝險等級：中

實作指引

AWS 建議盡可能降低或避免對長期憑證的依賴，而是採用暫時憑證和即時權限提升機制。長期憑證容易發生安全性風險並會增加營運負擔。對於大多數管理任務，以及事件回應任務，我們建議您實作[聯合身分](#)以及[適用於管理存取權的暫時權限提升](#)。在此模型中，使用者會請求提升至較高層級的權限 (例如事件回應角色)；如果使用者符合提升的資格，則會將請求傳送至核准者。如果請求獲得核准，使用者就會收到一組暫時 [AWS 憑證](#)，使用者可使用此憑證來完成其任務。在這些憑證過期後，使用者就必須提交新的提升權限請求。

我們建議在大多數事件回應情境中，使用暫時權限提升。正確的做法是使用 [AWS Security Token Service](#) 和[工作階段政策](#)來界定存取權的範圍。

當發生聯合身分不可用的情況，例如：

- 與遭盜用身分提供者 (IdP) 相關的中斷。
- 設定錯誤或人為錯誤會導致聯合存取管理系統遭到破壞。
- 分散式拒絕服務 (DDoS) 事件或使系統無法使用之類的惡意活動。

在上述案例中，應會有已設定的緊急存取權，可協助調查和及時修復事件。我們建議您使用[具有適當許可的使用者](#)，來執行任務和存取 AWS 資源。僅將根憑證用於[需要根使用者存取權的任務](#)。若要確認事件回應者是否具有 AWS 和其他相關系統的正確存取權，我們建議預先佈建專屬帳戶。此類帳戶需要提升的存取權，且必須受到嚴格的控制和監控。必須以執行必要任務所需的最低權限來建置這些帳戶，而存取權層級應以事件管理計畫中建立的程序手冊為基礎。

使用專用和專屬的使用者及角色作為最佳實務。透過新增 IAM 政策而暫時提升權限的使用者和角色存取權，會同時使得使用者在事件發生期間的存取權不明確，又有無法將提升的權限撤銷的風險。

您必須盡可能移除相依性，來確認可在各種可能的失敗情境下獲得存取權。為了做到這一點，建立程序手冊來確認事件回應使用者的建立身分是專屬安全帳戶中的使用者，且不會透過任何現有的聯合或單一登入 (SSO) 解決方案來管理事件回應使用者。每個個別回應者必須具備其專屬的指定帳戶。帳戶組態必須強制執行[強式密碼政策](#)和多重要素驗證 (MFA)。如果事件回應程序手冊僅需要 AWS Management Console 的存取權，使用者就不應設定存取金鑰，且應明確禁止使用者建立存取金鑰。您可以使用 IAM 政策或服務控制政策 (SCP) 進行設定，如 [AWS Organizations SCP](#) 的 AWS 安全最佳實務中所述。除了在其他帳戶中擔任事件回應角色的能力外，使用者不應具備任何權限。

在事件期間，必須將存取權授予其他內部或外部人員，來協助調查、修復和復原活動。在此案例中，使用先前提到的程序手冊機制，而且必須制定程序，以確認在事件完成後，立即將任何其他存取權撤回。

若要確認事件回應角色的使用是否受到適當的監控和稽核，則必須確保未在人員之間共用為此目的建立的 IAM 帳戶，且除非[特定任務所需](#)，否則不得使用 AWS 帳戶根使用者。如果需要根使用者 (例如，特定帳戶的 IAM 存取權不可用時)，請使用獨立的程序，其中有可用的程序手冊，來確認根使用者登入憑證和 MFA 字符是否可用。

若要為事件回應角色設定 IAM 政策，請考慮使用 [IAM Access Analyzer](#) 來根據 AWS CloudTrail 日誌產生政策。若要這麼做，請向管理員授予在非生產帳戶上事件回應角色的存取權，並透過程序手冊加以執行。完成後，您就可以建立政策來僅允許所採取的動作。接著就可以將此政策套用至所有帳戶中的所有事件回應角色。您可能希望為每個程序手冊建立個別 IAM 政策，來讓管理和稽核作業更輕鬆。範例程序手冊可能包含勒索軟體、資料洩漏、生產存取權遺失和其他情境的回應計畫。

使用事件回應帳戶來擔任[在其他 AWS 帳戶中專屬事件回應 IAM 角色](#)。必須將這些角色設定為僅供安全帳戶中的使用者擔任，而信任關係必須要求呼叫主體使用 MFA 進行驗證。這些角色必須使用嚴格控制範圍的 IAM 政策來控制存取權。確保所有對這些角色的 AssumeRole 請求都記錄在 CloudTrail 中並據以發出提醒，而使用這些角色採取的任何動作都會記錄下來。

強烈建議必須清楚地命名 IAM 帳戶和 IAM 角色，因此您可以輕鬆地在 CloudTrail 日誌中找到這些帳戶和角色。這類範例便是將 IAM 帳戶命名為 `<USER_ID>-BREAK-GLASS` 以及將 IAM 角色命名為 `BREAK-GLASS-ROLE`。

[CloudTrail](#) 會用來在 AWS 帳戶中記錄 API 活動，且應用來[設定對事件回應角色使用情形的提醒](#)。請參閱部落格貼文，其中會說明使用根金鑰如何設定提醒。您可以修改說明，以設定與事件回應 IAM 角色相關的 AssumeRole 事件的 [Amazon CloudWatch](#) 指標篩選條件至篩選條件：

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =  
  "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !=  
  "AwsServiceEvent" }
```

由於事件回應角色可能具備很高的存取權限，因此必須將這些提醒傳送給多個群組，並據此快速採取行動。

在事件期間，回應者可能需要存取未受 IAM 直接保護的系統。其中可能包含 Amazon Elastic Compute Cloud 執行個體、Amazon Relational Database Service 資料庫或軟體即服務 (SaaS) 平台。強烈建議使用此方法，而不是使用 SSH 或 RDP 等原生協定，[AWS Systems Manager Session Manager](#) 會用於對 Amazon EC2 執行個體的所有管理存取權。您可以使用安全且受稽核的 IAM 來控制此存取權。您也可以使用 [AWS Systems Manager Run Command 文件](#) 來自動化部分程序手冊，如此可減少使用者錯誤並縮短復原時間。如需資料庫和第三方工具的存取權，我們建議將存取憑證儲存在 AWS Secrets Manager 中，並將存取權授予事件回應者角色。

最後，應將事件回應 IAM 使用者帳戶的管理作業新增至[加入者、異動者和離職者程序](#)中，並定期審查和測試此管理作業，以確認僅允許預期的存取。

資源

相關文件：

- [管理對 AWS 環境的暫時提升存取權](#)
- [AWS 安全事件回應指南](#)
- [AWS Elastic Disaster Recovery](#)
- [AWS Systems Manager Incident Manager](#)
- [設定 IAM 使用者的帳戶密碼政策](#)
- [使用 AWS 中的多重要素驗證 \(MFA\)](#)
- [使用 MFA 設定跨帳戶存取權](#)
- [使用 IAM Access Analyzer 來產生 IAM 政策](#)
- [多帳戶環境中 AWS Organizations 服務控制政策的最佳實務](#)
- [如何在使用 AWS 帳戶的根存取金鑰時接收通知](#)
- [使用 IAM 受管政策來建立精細的工作階段許可](#)

- [緊急存取](#)

相關影片：

- [將 AWS 中的事件回應和鑑識自動化](#)
- [執行手冊、事件報告和事件回應的 DIY 指南](#)
- [準備和回應 AWS 環境中的安全事件](#)

相關範例：

- [實驗室：AWS 帳戶設定和根使用者](#)
- [實驗室：使用 AWS 主控台和 CLI 來回應事件](#)

SEC10-BP06 部署前工具

確認安全人員具有預先部署的適當工具，以縮短調查直至復原的時間。

未建立此最佳實務時的曝險等級：中

實作指引

若要自動執行安全回應和操作函數，您可以使用的 APIs 和工具集 AWS。您可以將身分管理、網路安全、資料保護和監控功能完全自動化，並使用現有的熱門軟體開發方法遞送這些功能。建置安全自動化時，您的系統可以監控、檢閱和啟動回應，而不是讓人員監控您的安全地位並手動回應事件。

如果您的事件回應團隊持續以相同方式回應提醒，可能會形成提醒疲勞的風險。隨著時間的推移，團隊可能會變得對收到提醒不敏感，而且在處理一般情況時可能會犯錯，或是錯過不尋常的提醒。自動化使用能夠處理重複和一般提醒的功能，讓人員處理敏感和獨特的事件，有助於避免發生提醒疲倦的情形。整合異常偵測系統，例如 Amazon GuardDuty、AWS CloudTrail Insights 和 Amazon CloudWatch 異常偵測，可以減輕常見閾值型警示的負擔。

您可以透過程式設計方式將程序中的步驟自動化，以改善手動程序。定義事件的補救模式之後，您可以將該模式分解為可行的邏輯，並撰寫程式碼來執行該邏輯。回應人員接著可以執行該程式碼來修復問題。隨著時間的推移，您可以將越來越多的步驟自動化，最終自動處理整個類別的常見事件。

在安全調查期間，您需要能夠檢閱相關日誌以記錄和了解該事件的完整範圍和時間表。產生提醒也需要日誌，以指出特定關注的動作已發生。選擇、啟用、儲存和設定查詢與擷取機制和設定提醒至關重要。此外，提供搜尋日誌資料之工具的有效方法是 [Amazon Detective](#)。

AWS 提供超過 200 個雲端服務和數千種功能。我們建議您檢閱可支援並簡化事件回應策略的服務。

除了日誌記錄之外，您還應該開發和實作[標記策略](#)。標記有助於提供有關 AWS 資源用途的背景。標記也可用於自動化。

實作步驟

選取並設定日誌以進行分析和提醒

請參閱下列有關設定事件回應日誌記錄的文件：

- [安全事件回應的日誌記錄策略](#)
- [SEC04-BP01 設定服務和應用程式日誌記錄](#)

啟用安全服務以支援偵測和回應

AWS 提供原生偵測、預防和回應功能，其他服務可用於架構自訂安全解決方案。如需安全事件回應最相關的服務清單，請參閱[雲端功能定義](#)。

制定和實作標記策略

取得有關業務使用案例和與 AWS 資源相關的內部利益相關者的內文資訊可能很困難。其中一種方法是標籤形式，將中繼資料指派給您的 AWS 資源，並包含使用者定義的金鑰和值。您可以建立標籤，依目的、擁有者、環境、處理的資料類型以及您選擇的其他條件來分類資源。

擁有一致的標記策略可以加快回應時間，並允許您快速識別和辨別 AWS 資源的上下文資訊，從而將花費在組織內容上的時間降至最低。標籤也可以作為啟動回應自動化的機制。如需有關要標記的內容的詳細資訊，請參閱[標記您的 AWS 資源](#)。您需要先定義要在整個組織中實作的標籤。之後，您將實作並強制執行此標記策略。如需實作和強制執行的詳細資訊，請參閱[使用 AWS 標籤政策和服務控制政策實作 AWS 資源標記策略 \(SCPs \)](#)。

資源

相關 Well-Architected 的最佳實務：

- [SEC04-BP01 設定服務和應用程式日誌記錄](#)
- [SEC04-BP02 在標準化的位置擷取日誌、調查結果和指標](#)

相關文件：

- [安全事件回應的日誌記錄策略](#)
- [事件回應雲端功能定義](#)

相關範例：

- [Amazon 和 Amazon Detective 的威脅偵測 GuardDuty 和回應](#)
- [Security Hub 研討會](#)
- [使用 Amazon Inspector 管理漏洞](#)

SEC10-BP07 執行模擬

在組織隨著時間成長和發展時，威脅態勢也會跟著演變，因此持續審查事件回應能力是很重要的。執行模擬 (也稱為比賽日) 是可用於執行此評估的一種方法。模擬會使用真實世界的安全事件案例，這些案例旨在模擬威脅參與者的策略、技術和程序 (TTP)，並且讓組織可藉由回應這些可能發生在現實中的模擬網路事件，來運用和評估其事件回應能力。

建立此最佳實務的優勢：模擬具有多種優勢：

- 驗證網路整備程度和培養事件回應人員的信心。
- 測試工具和工作流的正確性及效率。
- 根據您的事件回應計畫，精進溝通和呈報方法。
- 提供回應罕見媒介的機會。

未建立此最佳實務時的曝險等級：中

實作指引

主要的模擬類型有三種：

- **桌上模擬演練**：桌上模擬方法是基於討論的會議，涉及各種事件回應利害關係人的角色和責任練習，並使用已建立的溝通工具和程序手冊。模擬演練促進通常可在虛擬場地、實體場地或兩者的組合於一整天內完成。桌上模擬演練以討論為主軸，因此側重於程序、人員和協作。技術在討論中是不可或缺的一部分，但事件回應工具或指令碼的實際使用通常不是桌上模擬演練的一部分。
- **紫隊模擬演練**：紫隊模擬演練提高了事件回應人員 (藍隊) 和模擬威脅參與者 (紅隊) 之間的協作層級。藍隊由安全營運中心 (SOC) 的成員組成，但也可以包含在實際網路事件期間涉入的其他利害關係人。紅隊由滲透測試團隊或受過攻擊性安全培訓的主要利害關係人組成。紅隊在設計場景時會與模

擬演練協調員合作，使場景精確且可行。在紫隊模擬演練期間，主要重點是偵測機制、工具和支援事件回應工作的標準操作程序 (SOP)。

- 紅隊模擬演練：在紅隊模擬演練期間，攻方 (紅隊) 會進行模擬，以在預定範圍內達到某個目標或一組目標。守方 (藍隊) 不一定知道模擬演練的範圍和持續時間，這對他們應對實際事件的能力可呈現出更真實的評估。由於紅隊模擬演練可能是侵入性測試，請謹慎行事並施加控制，以確認該模擬演練不會對您的環境造成實際傷害。

考慮定期推行網路模擬。每種模擬演練類型都可以為參與者和整個組織提供特有的好處，因此您可以選擇從較不複雜的模擬類型 (例如桌上模擬演練) 開始著手，然後再進入更複雜的模擬類型 (紅隊模擬演練)。您應根據自身的安全成熟度、資源和所需的結果來選取模擬類型。由於複雜性和成本較高，有些客戶可能不會選擇執行紅隊模擬演練。

實作步驟

無論您選擇的模擬類型為何，模擬通常會執行下列實作步驟：

1. 定義核心演練元素：定義模擬情境與模擬的目標。這兩者都應獲得領導階層的允許。
2. 識別關鍵利害關係人：模擬演練至少需要模擬演練協調員和參與者。根據情境，可能會涉及法律、通訊或主管領導階層等其他利害關係人。
3. 建置和測試情境：如果特定元素不可行，則可能需要在情境建置期間加以重新定義。預計最終的情境會成為此階段的輸出。
4. 促進模擬：模擬的類型將決定使用的促進形式 (編撰的場景對比於高度技術性的模擬場景)。協調員應使其促進策略與模擬演練目標相對應，他們應盡可能吸引所有模擬演練參與者，以提供最大的效益。
5. 撰寫事後報告 (AAR)：識別進展順利的領域、可以改進的領域，以及潛在的差距。AAR 應衡量模擬的有效性以及團隊對於模擬事件的應變能力，以便在未來的模擬追蹤進展幅度。

資源

相關文件：

- [AWS 事件回應指南](#)

相關影片：

- [AWS GameDay - Security Edition](#)

- [執行有效的安全事件回應模擬](#)

作業

操作是進行事件回應的核心。這也是採取行動回應和補救安全事件的所在。操作包括以下五個階段：偵測、分析、遏制、根除和復原。下表中可找到這些階段和目標的說明。

階段	目標
偵測	識別潛在的安全事件。
分析	確定安全事件是否為事件，並評估事件的範圍。
遏制	盡量縮小並限制安全事件的範圍。
根除	移除與安全事件相關的未經授權資源或成品。實作造成安全事件的緩和措施。
復原	將系統還原至已知的安全狀態，並監控這些系統以確認威脅未再發生。

這些階段應做為您回應和操作安全事件時的指引，以便採取有效且可靠的方式來回應。您採取的實際行動會因事件而有所不同。舉例來說，對於涉及勒索軟體的事件與涉及公有 Amazon S3 儲存貯體的事件將採取不同的回應步驟。此外，這些階段不一定會依序發生。在遏制和根除之後，您可能需要返回分析，以了解採取的行動是否有效。

涵蓋人員、流程和技術的完整準備，是讓操作發揮效用的關鍵。因此，請依照[準備](#)一節的最佳實務，以便有效回應作用中安全事件。

若要進一步了解，請參閱 AWS Security Incident Response Guide 中的 [Operations](#) 一節。

事後處理

威脅態勢不斷變化，因此組織有效保護環境的能力也務必同樣保持動態。持續改進的關鍵在於反覆執行事件和模擬的結果，以便改善有效偵測、回應和調查可能發生的安全事件的能力、減少可能的漏洞、縮短回應時間，並且恢復安全操作。下列機制可協助您驗證組織是否具備最新功能和知識，以便在任何情況下有效回應。

最佳實務

- [SEC10-BP08 建立從事件中學習的架構](#)

SEC10-BP08 建立從事件中學習的架構

實作經驗教訓的架構和根本原因分析能力，不僅有助改善事件回應能力，還有助防止事件重複發生。透過學習每個事件，您可以協助避免重複相同的錯誤、披露或錯誤設定，不僅能夠改善安全狀態，還可以盡可能縮短因可預防情況而損失的時間。

未建立此最佳實務時的曝險等級：中

實作指引

實作經驗教訓是非常重要的，其可在高層級實現以下幾點：

- 什麼時候開設經驗教訓課程？
- 經驗教訓課程中包含哪些內容？
- 經驗教訓課程的進行方式？
- 這個課程的參與者以及參與方式？
- 如何識別待改善之處？
- 您將如何確保有效地追蹤和實作待改善之處？

此架構不應該針對或責怪個人，而應該專注於改善工具和流程。

實作步驟

除了前述所列的高層級結果之外，確保您提出正確問題以從流程中獲得最大價值 (即協助您找到可行改善之處的資訊) 非常重要。考慮這些問題，有助您發起經驗教訓的討論：

- 事件是什麼？
- 第一次識別事件的時間？
- 事件的識別方式？
- 哪些系統對活動發出提醒？
- 涉及哪些系統、服務和資料？
- 具體發生的事件？
- 哪些方面做得很好？

- 哪些方面做得不好？
- 哪個流程或程序失敗或未能擴展以回應事件？
- 在以下幾個領域有哪些可以改善之處：
 - 人物
 - 需要聯絡的對象實際上是否有空，並且聯絡人清單是最新的嗎？
 - 人們是否缺少有效回應和調查事件所需的培訓或能力？
 - 適當的資源是否已準備就緒且可供使用？
 - 流程
 - 是否遵循流程和程序？
 - 是否已記錄並提供這類事件的流程和程序？
 - 是否缺少必要的流程和程序？
 - 回應人員是否能夠即時存取所需的資訊以回應問題？
 - 技術
 - 現有的提醒系統是否能有效地識別活動，並據以發出提醒？
 - 如何減少 time-to-detection 50%？
 - 是否需要改善現有提醒，或是需要針對此類事件建立新的提醒？
 - 現有的工具是否允許對事件進行有效的調查 (搜尋/分析)？
 - 可以做什麼來協助加快這類事件的識別速度？
 - 可以做什麼來協助避免這類事件再次發生？
 - 負責改善計畫的人是誰，您將如何測試是否已實作此計畫？
 - 實作和測試其他監控或預防性控制和流程的時間表為何？

這份清單並不詳盡，但可作為起點，幫助您識別組織和企業的需求，以及如何分析這些需求，以便最有效地從事件中學習並持續改善安全狀態。最重要的是透過將經驗教訓納入事件回應流程，文件和利害關係人期望的標準部分。

資源

相關文件：

- [AWS 安全事件回應指南 - 建立從事件中學習的架構](#)
- [NCSC CAF 指引 - 學到的課程](#)

應用程式安全

應用程式安全 (AppSec) 介紹如何為開發工作負載的安全屬性進行設計、建置與測試的整體過程。您應該安排組織成員接受適當訓練，了解您的建置與發佈基礎結構的安全屬性，以及應用自動化來識別出安全問題。

採用應用程式安全測試做為軟體開發生命週期 (SDLC) 與發佈後程序中的常規部分，有助於確保您可建立一套結構化機制，專門識別、修正應用程式安全問題，以及防止這些問題進入您的生產環境。

您的應用程式開發方法應該在設計、建置與操作工作負載期間納入安全控制。過程當中，可以調整程序，達到持續減少缺陷和最低技術負債。例如，在設計階段中應用威脅建模有助於提早發現設計瑕疵，修正更加簡單，成本節省更多，而不需要等到日後才能進行緩解。

解決瑕疵的成本與複雜性通常比過去在 SDLC 中來得低。最簡單的解決問題方法就是別讓問題發生，因此從使用威脅模型開始，有助於您專注在設計階段的正確成果。隨著 AppSec 計畫逐漸成熟，您可以自動化方式提高測試量，改善意見回饋對建置人員的準確性 (保真度)，同時縮短安全審核所需要的時間。這些動作全都可以改善所建置軟體的品質，並且加快功能進入生產階段。

這些實作準則著重於四個領域：組織與文化、管道的安全性、管道中的安全性以及相依性管理。每個區域都會提供一組可實作原則，並針對設計、建置與操作工作負載的做法提供端對端檢視。

在 AWS 中，您可以使用許多方法來解決應用程式安全計劃問題。當中有一些方法依賴技術，而其他方法則著重在應用程式安全計劃的人員和組織層面。

最佳實務

- [SEC11-BP01 應用程式安全訓練](#)
- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)
- [SEC11-BP03 定期進行滲透測試](#)
- [SEC11-BP04 執行程式碼檢閱](#)
- [SEC11-BP05 集中化套件和相依性的服務](#)
- [SEC11-BP06 以程式設計方式部署軟體](#)
- [SEC11-BP07 定期評估管道的安全屬性](#)
- [SEC11-BP08 打造在工作負載團隊中納入安全所有權的計畫](#)

SEC11-BP01 應用程式安全訓練

為您的團隊提供安全開發和操作實務的培訓，協助他們建置安全的優質軟體。此實務可協助您的團隊在開發生命週期中提早預防、偵測和修復安全問題。請考慮提供涵蓋建立威脅模型、安全編碼實務，以及使用服務進行安全組態設定和操作的培訓。透過自助服務資源，提供培訓內容存取權給您的團隊，並定期收集他們的意見回饋以確保持續進步。

預期成果：您可以從一開始就將安全性納入考量，為團隊提供設計和建置軟體所需的知識和技能。經過有關建立威脅模型和安全開發實務的培訓後，您的團隊將對潛在安全風險，以及如何在軟體開發生命週期 (SDLC) 期間減輕這些風險有更深入的了解。這種主動式安全措施是團隊文化的一部分，能夠讓您及早識別和修復潛在的安全問題。因此，您的團隊可以更有效率地提供高品質、安全的軟體和功能，進而加速整體交付時間表。您在組織中擁有協作且具包容性的安全文化，並將安全擁有權與所有建置者共享。

常見的反模式：

- 您等到安全審查階段，才開始考慮系統的安全屬性。
- 您將所有的安全性決定工作全部留給中央安全團隊。
- 您未在 SDLC 溝通如何做出與整體安全期待或組織政策相關的決定。
- 您太晚執行安全審查程序。

建立此最佳實務的優勢：

- 可在開發生命週期初期更清楚了解組織對於安全的要求。
- 可以更快識別、修復安全問題，進而加快功能交付速度。
- 改善軟體和系統的品質。

未建立此最佳實務時的曝險等級：中

實作指引

若要建置安全且高品質的軟體，請為您的團隊提供有關安全開發和操作應用程式的常見實務培訓。此實務可協助您的團隊在開發生命週期中提早預防、偵測和修復安全問題，以加速交付時間表。

若要完成此實務，請考慮使用[建立威脅模型研討會](#)等的 AWS 資源，訓練您的團隊建立威脅模型。建立威脅模型可以讓您的團隊從一開始就了解潛在安全風險，並從安全考量來設計系統。此外，您可以提供

有關安全開發實務的 [AWS 培訓 and Certification](#)、產業或AWS合作夥伴培訓。如需更多設計、開發、保護和有效率地大規模操作的完整方法相關資訊，請參閱 [AWS DevOps 指引](#)。

明確定義和傳達組織的安全審查程序，並概述團隊、安全團隊和其他利害關係人的責任。發布展示如何達到您的安全要求的自助式指引、程式碼範例和範本。您可以使用 [AWS CloudFormation](#)、[AWS Cloud Development Kit \(AWS CDK\) \(AWS CDK\) 建構模組](#) 和 [Service Catalog](#) 等 AWS 服務，提供預先核准的安全組態，並減少自訂設定的需求。

定期收集團隊的安全審查程序與培訓體驗方面的意見回饋，並藉由該回饋持續改進。執行演練日或錯誤修復日活動，以識別和解決安全問題，同時增強團隊的技能。

實作步驟

1. 識別培訓需求：透過調查、程式碼檢閱或與團隊成員討論，評估團隊中有關安全開發實務的目前技能水平和知識差距。
2. 規劃培訓：根據確定的需求建立培訓計畫，培訓範圍涵蓋建立威脅模型、安全編碼實務、安全測試和安全部署實務等相關主題。使用 [建立威脅模型研討會](#)、[AWS 培訓 and Certification](#)，以及產業或AWS合作夥伴培訓計畫等資源。
3. 排程和交付培訓：為您的團隊安排定期培訓課程或研討會。這些可以由講師指導或自行安排進度，端視團隊的偏好設定和可用狀況而定。鼓勵實作練習和實際範例，以強化學習。
4. 定義安全審查程序：與您的安全團隊和其他利害關係人合作，明確定義應用程式的安全審查程序。記錄參與程序的每個團隊或個人的責任，包括您的開發團隊、安全團隊和其他利害關係人。
5. 建立自助式資源：製作自助式指引、程式碼範例和範本，示範如何做才能符合組織的安全要求。考慮採用 [CloudFormation](#)、[AWS CDK Constructs](#) 和 [Service Catalog](#) 等 AWS 服務，提供預先核准且安全的組態，並減少自訂設定的需求。
6. 溝通和社交：有效地將安全審查程序和可用的自助式資源傳達給您的團隊。進行培訓課程或研討會，讓團隊成員熟悉這些資源，並確認他們了解如何使用這些資源。
7. 收集意見回饋並改進：定期收集團隊對於安全審查程序與培訓體驗的意見回饋。利用此回饋識別需要改進的領域，並持續改良培訓教材、自助式資源和安全審查程序。
8. 執行安全防護演練：安排演練日或錯誤修復日活動，以識別和解決應用程式內的安全問題。這些練習不僅有助於發現潛在的漏洞，同時也為您的團隊提供實際的學習機會，增強他們在安全開發和操作方面的技能。
9. 持續學習和改進：鼓勵您的團隊掌握最新的安全開發實務、工具和技術。定期審查和更新您的培訓教材和資源，以反映不斷變化的資安情勢和最佳實務。

資源

相關的最佳實務：

- [SEC11-BP08 打造在工作負載團隊中納入安全所有權的計畫](#)

相關文件：

- [AWS 培訓 和認證](#)
- [如何思考雲端安全管控](#)
- [如何建立威脅模型](#)
- [加速訓練 – AWS 技能培養](#)
- [AWS DevOps Sagas](#)

相關影片：

- [預防性安全：考量與方法](#)

相關範例：

- [威脅建模相關的研討會](#)
- [開發人員的產業認知](#)

相關服務：

- [AWS CloudFormation](#)
- [AWS Cloud Development Kit \(AWS CDK\) \(AWS CDK\) 建構模組](#)
- [Service Catalog](#)

SEC11-BP02 自動化在整個開發和發布生命週期的測試

自動化在整個開發和發布生命週期的安全署性測試。自動化可以讓軟體在發布之前更容易一致且重複地識別潛在問題，因此能降低將供應軟體的安全問題風險。

預期成果：自動化測試的目標是提供以程式設計方式，及早偵測潛在問題，而且常常是遍及整個開發生命週期。啟用自動化迴歸測試時，您可以對變更之後的軟體重新執行功能性與非功能性的測試，確認

先前測試過的軟體運作仍如預期。如果定義安全單元測試來檢查常見的錯誤組態，例如損壞或缺失的驗證，您就能夠在開發過程中及早發現和修正這些問題。

測試自動化會使用專用測試個案來進行應用程式驗證，測試期間以應用程式需求和所需功能為基礎。自動化測試會將產生的測試輸出與其個別預期輸出進行比較，得到最後結果，進而加快整體的測試生命週期。包括像是迴歸測試與單位測試套組的測試方法最適合自動化應用。自動化安全屬性測試，建置人員就能接收自動化意見回饋，而不需要等待舉行安全審核。採用靜態或靜態程式碼分析的自動化測試，可以提高程式碼品質，並協助及早在開發生命週期中偵測出潛在的軟體問題。

常見的反模式：

- 未傳達測試個案與自動化測試的測試結果。
- 僅在即將發佈前執行自動化測試。
- 自動化有經常改變需求的測試個案。
- 無法提供如何解決安全測試結果的指引。

建立此最佳實務的優勢：

- 降低人員評估系統安全署性的依賴性。
- 可在跨多個工作串流之間找到一致結果，進而提高一致性。
- 降低造成安全問題被導入產品線上軟體的可能性。
- 因提早捕捉到軟體問題，而縮短偵測與矯正之間的範圍時段。
- 提高跨多個工作串流之系統或重複行為的能見度，其可用來促進整體組織改進。

未建立此最佳實務時的風險暴露等級：中

實作指引

隨著軟體逐漸建置，採用各種不同機制來測試軟體，確保您正根據應用程式的業務邏輯為主的功能性需求，以及著重應用程式可靠性、效能和安全性的非功能性需求，進行應用程式的測試作業。

靜態應用程式安全測試 (SAST) 分析原始程式碼是否有異常的安全模式，並提供可能存在缺陷程式碼的提示。SAST 依賴靜態輸入來測試某個範圍的已知安全問題，這些輸入包括文件 (需求規格、設計文件，以及設計規格) 和應用程式原始程式碼。靜態程式碼分析器可協助加快大量程式碼的分析作業。[NIST 品質群組](#)則提供[原始程式碼安全分析器](#)的比較，其中包括能用於[位元組程式碼掃描器](#)和[二進位程式碼掃描器](#)的開放原始碼工具。

應用動態分析安全測試 (DAST) 方法以補充您的靜態測試，這個方法會在應用程式執行期間進行測試，識別潛在的意外行為。動態測試可用來偵測出靜態分析無法偵測出的潛在問題。在程式碼儲存、建置和管道等階段進行測試，您就可以檢查進入程式碼當中的各種不同潛在問題類型。[Amazon Q Developer](#) 提供程式碼建議，包括在建置者的 IDE 中執行安全掃描。[Amazon CodeGuru 安全工具](#) 可以識別關鍵問題、安全問題，以及在應用程式開發期間難以發現的錯誤，並提供改善程式碼品質的建議。擷取軟體材料清單 (SBOM) 也可讓您擷取正式紀錄，其中包含建置軟體時所使用的各種元件的詳細資訊和關係。如此您便能知會漏洞管理，並快速識別軟體或元件相依項及供應鏈風險。

[開發人員安全研討會](#) 使用 AWS 開發人員工具 (如 [AWS CodeBuild](#)、[AWS CodeCommit](#) 和 [AWS CodePipeline](#)) 執行包括 SAST 和 DAST 測試方法的發布管道自動化。

隨著 SDLC 繼續進行，建立包括安全團隊定期進行應用程式審核的反覆程序。收集自這些安全審核的意見回饋應加以解決，並在發佈準備度審核時加以驗證。這些審核作業會建立堅實強大的應用程式安全狀態，並提供建置人員可解決潛在問題的可行動意見回饋。

實作步驟

- 實作一致的 IDE、程式碼審核，以及包括安全測試的 CI/CD 工具。
- 考慮 SDLC 中的哪個位置適合封鎖管道，而不只是通知建置者出現需要矯正的問題。
- [自動化安全協助程式 \(ASH\)](#) 是開放原始碼程式碼安全掃描工具的範例。
- 使用自動化工具 (例如，已與開發人員 IDE 完成整合的 [Amazon Q Developer](#)，以及可在遞交認可時掃描程式碼的 [Amazon CodeGuru 安全工具](#)) 執行測試或程式碼分析，協助建置者在適當時間取得意見回饋。
- 使用 AWS Lambda 進行建置時，您可以使用 [Amazon Inspector](#) 來掃描多個功能的應用程式程式碼。
- 如果將自動化測試納入 CI/CD 管道，您應該使用票證系統來追蹤通知，以及軟體問題的矯正。
- 如果是可能會產生調查結果的安全測試，連結矯正的指引將有助於建置人員改善程式碼品質。
- 定期分析自動化工具所找到的調查結果，以便找出下次自動化、建置者訓練或認知行銷活動的優先順序。
- 若要擷取 SBOM 做為 CI/CD 管道的一部分，請使用 [Amazon Inspector SBOM 產生器](#)，以 CycloneDX SBOM 格式為封存、容器映像、目錄、本機系統和編譯的 Go and Rust 二進位檔製作 SBOM。

資源

相關的最佳實務：

- [DevOps 指引：DL.CR.3 為程式碼任務建立明確的完成條件](#)

相關文件：

- [持續交付與持續部署](#)
- [AWS DevOps 能力合作夥伴](#)
- [AWS 安全能力合作夥伴 \(應用程式安全\)](#)
- [選擇 Well-Architected CI/CD 方法](#)
- [Amazon CodeGuru 安全工具中的機密偵測](#)
- [Amazon CodeGuru 安全工具偵測程式庫](#)
- [配合有效管控，加速在 AWS 的部署](#)
- [AWS 如何達到自動化安全、無人為介入的部署](#)
- [Amazon CodeGuru 安全工具如何協助您有效平衡安全性和速度](#)

相關影片：

- [無人為介入：Amazon 的自動化持續交付管道](#)
- [自動化跨帳戶 CI/CD 管道](#)
- [Amazon 的軟體開發程序](#)
- [Amazon 的測試軟體和系統](#)

相關範例：

- [開發人員的產業認知](#)
- [自動化安全協助程式 \(ASH\)](#)
- [AWS CodePipeline 管控 - Github](#)

SEC11-BP03 定期進行滲透測試

定期對您的軟體進行滲透測試。這項機制有助於識別自動化測試或手動程式碼審查時，未能偵測到的潛在軟體問題。此外還有助於了解偵測控制的效用。滲透測試應嘗試判斷軟體是否會透過非預期的方式執行，例如暴露原本應受保護的資料，或是授予超乎預期的較廣泛權限。

預期成果：滲透測試可用來為您的應用程式安全屬性進行偵測、修復和驗證。軟體開發生命週期 (SDLC) 期間應該進行定期與排程型滲透測試。從滲透測試找到的調查結果應事先解決，才能安排軟體發行。您應該分析從滲透測試得到的調查結果，並識別是否有任何問題可使用自動化找出。實施包括主動意見回饋機制的定期和可重複滲透測試程序，可協助建置人員得知指引，並改善軟體品質。

常見的反模式：

- 只對已知或普遍存在的安全問題進行滲透測試。
- 滲透測試應用程式 (不含相依第三方工具和程式庫)。
- 只對套件安全問題進行滲透測試，且不評估已實作的商業邏輯。

建立此最佳實務的優勢：

- 提高軟體在發行前的安全屬性信心。
- 可識別偏好應用程式模式，並藉以提高軟體品質的機會。
- 在開發生命週期初期進行的意見回饋循環流程，當中的自動化或額外訓練可以改善軟體的安全屬性。

未建立此最佳實務時的風險暴露等級：高

實作指引

滲透測試是一種結構化的安全測試練習，過程當中，您會執行計畫的安全性缺口情境，對安全控制進行偵測、修復與驗證。滲透測試從偵察活動開始，過程中會根據目前的應用程式設計與其相依性收集資料。已經建置並執行精選的安全特定測試情境清單。這些測試的主要目的在於找出您的應用程式中的安全問題，這些問題可能會被利用來非預期地存取環境，或未經授權存取資料。當您推出新功能，或是每當應用程式遭遇重大的功能變更或進行技術實作，您就應該進行滲透測試。

您應該識別開發生命週期中最適合進行滲透測試的階段。這項測試的執行時間應該盡量延到系統功能接近預定發行階段之時，而且要保留足夠修復任何問題的時間。

實作步驟

- 建立處理滲透測試範圍限制方式的結構化程序，前提是這個關於[威脅模型](#)的程序是維持內容的好方法。
- 識別開發週期中最適合進行滲透測試的時機。進行測試時應該是預期應用程式進行最少變更，而且有足夠時間進行修復。

- 訓練建置人員學會從滲透測試調查結果預期哪些內容，以及如何取得關於修復的資訊。
- 使用工具，透過自動化共通或可重複測試，加速滲透測試程序。
- 分析滲透測試調查結果來找出系統性安全問題，並使用這份資料，得知其他的自動化測試與持續進行的建置人員教育。

資源

相關的最佳實務：

- [SEC11-BP01 應用程式安全訓練](#)
- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)

相關文件：

- [AWS 滲透測試](#) 提供在 AWS 上進行滲透測試的詳細指引
- [配合有效管控，加速在 AWS 的部署](#)
- [AWS 安全能力合作夥伴](#)
- [現代化您在 AWS Fargate 上的滲透測試架構](#)
- [AWS Fault Injection Simulator](#)

相關範例：

- [自動化配合 AWS CodePipeline 的 API 測試 \(GitHub\)](#)
- [自動化安全協助程式 \(GitHub\)](#)

SEC11-BP04 執行程式碼檢閱

實作程式碼檢閱，協助驗證正在開發之軟體的品質和安全性。程式碼檢閱旨在讓原始程式碼作者以外的團隊成員檢閱程式碼是否潛藏任何問題、漏洞，以及是否遵守編碼標準和最佳實務。此處理程序有助於擷取原始開發人員可能忽略的錯誤、不一致和安全漏洞。使用自動化工具協助檢閱程式碼。

預期結果：您在開發期間納入程式碼檢閱流程，以提高撰寫的軟體品質。您可以利用在程式碼檢閱期間的學習，提升經驗不足之團隊成員的技能。您可以使用自動化工具和測試來識別自動化的機會，並支援程式碼檢閱程序。

常見的反模式：

- 您在部署之前不會檢閱程式碼。
- 撰寫程式碼和檢閱程式碼的是同一個人。
- 您未使用自動化程序和工具來協助或協調程式碼檢閱。
- 您未在建置者開始檢閱程式碼之前，先讓他們接受應用程式安全的訓練。

建立此最佳實務的優勢：

- 程式碼品質更高。
- 經由重複使用常用方法而使程式碼開發更具一致性。
- 減少在滲透測試與後期階段找出問題的數量。
- 團隊內部的知識轉移效能更高。

未建立此最佳實務時的曝險等級：中

實作指引

程式碼檢閱有助於在開發期間驗證軟體的品質和安全性。手動檢閱旨在讓原始程式碼作者以外的團隊成員檢閱程式碼是否潛藏任何問題、漏洞，以及是否遵守編碼標準和最佳實務。此處理程序有助於擷取原始開發人員可能忽略的錯誤、不一致和安全漏洞。

請考慮使用 [Amazon CodeGuru 安全工具](#) 協助自動執程式碼檢閱。CodeGuru 安全工具使用機器學習和自動推理來分析您的程式碼，並識別潛在的安全漏洞和編碼問題。將自動化程式碼檢閱與現有程式碼儲存庫整合，並持續整合/持續部署 (CI/CD) 管道。

實作步驟

1. 建立程式碼檢閱程序：

- 定義進行程式碼檢閱的時間點，例如在將程式碼合併到主要分支之前，或在部署到生產環境之前。
- 確定誰應該參與程式碼檢閱程序，例如團隊成員、資深開發人員和安全專家。
- 決定程式碼檢閱方法，包括使用的程序和工具。

2. 設定程式碼檢閱工具：

- 評估並選取符合您團隊需求的程式碼檢閱工具，例如 GitHub 提取請求或 CodeGuru 安全工具
- 將所選工具與您現有的程式碼儲存庫和 CI/CD 管道整合。

- 設定工具以強制執行程式碼檢閱要求，例如檢閱者人數下限與核准規則。
3. 定義程式碼檢閱檢查清單和指引：
 - 建立程式碼檢閱檢查清單，或概述應檢閱之內容的指引。考慮程式碼品質、安全漏洞、遵守編碼標準和效能等因素。
 - 與開發團隊共用檢查清單或指引，並確認每個人都了解預期情況。
 4. 訓練開發人員程式碼檢閱最佳實務：
 - 為您的團隊提供有關如何執行有效程式碼檢閱的訓練。
 - 教育您的團隊有關在檢閱過程中，需注意的應用程式安全原則和常見漏洞。
 - 鼓勵知識分享和配對程式設計工作階段，以提升經驗不足之團隊成員的技能。
 5. 實作程式碼檢閱程序：
 - 將程式碼檢閱步驟整合到您的開發工作流程中，例如建立提取請求和指派檢閱者。
 - 要求在合併或部署之前，針對程式碼變更內容進程式碼檢閱。
 - 在檢閱過程中，鼓勵公開溝通和建設性的意見回饋。
 6. 監控並改善：
 - 定期審查程式碼檢閱程序的有效性，並收集團隊的意見回饋。
 - 識別自動化或工具改進的機會，以簡化程式碼檢閱程序。
 - 根據學習和產業最佳實務，持續更新和精簡程式碼檢閱檢查清單或指引。
 7. 培養程式碼檢閱文化：
 - 強調程式碼檢閱的重要性，以維持程式碼品質和安全性。
 - 慶祝程式碼檢閱程序的成功和學習成果。
 - 營造一個可以讓開發人員放心地提供和接收意見回饋的協作和支援環境。

資源

相關的最佳實務：

- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)

相關文件：

- [DevOps 指引：DL.CR.2 針對程式碼變更執行對等檢閱](#)
- [關於 GitHub 中的提取請求](#)

相關範例：

- [使用 Amazon CodeGuru 安全工具自動執行程式碼檢閱](#)
- [使用 Amazon CodeGuru 安全工具 CLI，自動偵測 CI/CD 管道中的安全漏洞與錯誤](#)

相關影片：

- [使用 Amazon CodeGuru 安全工具持續改善程式碼品質](#)

SEC11-BP05 集中化套件和相依性的服務

提供可讓您的團隊取得軟體套件和其他相依性的集中化服務。這樣套件就能先接受驗證，再納入撰寫的軟體，並提供您組織中使用的軟體分析資料來源。

預期結果：除了您撰寫的程式碼之外，也可以從外部軟體套件建置您的工作負載。重複使用的實作功能變得更加簡單，例如 JSON 剖析器或加密程式庫。您可以集中這些套件和相依項的來源，如此一來您的安全團隊就能在使用這些項目前，先進行驗證。使用這個方法再加上手動與自動測試流程，即能提升對於您開發之軟體品質的信心。

常見的反模式：

- 您從網際網路的任意儲存庫中取出套件。
- 您未測試新套件，便將其提供給建置者。

建立此最佳實務的優勢：

- 更清楚了解哪些套件將用於建置中的軟體。
- 可以在了解過實際使用情況而需要更新套件時通知工作負載團隊。
- 降低在軟體中納入有問題套件的風險。

未建立此最佳實務時的風險暴露等級：中

實作指引

提供可讓建置人員輕鬆取得的套件和其他相依性集中化服務。集中化服務可依照邏輯方式進行集中，而非實作成單一龐大的系統。這個方法可讓您用符合建置人員需求的方式提供服務。您應該實作一種能在

發生更新或新需求萌生時，快速在儲存庫新增套件的方法。[AWS CodeArtifact](#) 或類似的 AWS 合作夥伴解決方案等 AWS 服務可提供發揮這種能力的方法。

實作步驟

- 實作依照邏輯方式集中，而且各種軟體開發所在環境均可使用的儲存庫服務。
- 將儲存庫的存取作業納入 AWS 帳戶 銷售程序。
- 建置自動化測試流程，在將套件發行至儲存庫之前先進行測試。
- 維護最常使用的套件、語言，以及變更程度最高團隊的規格表。
- 提供可讓建置人員團隊自動要求新套件與提供意見回饋的機制。
- 定期掃描儲存庫中的套件，識別最近所找到問題的潛在影響。

資源

相關的最佳實務：

- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)

相關文件：

- [DevOps 指引：DL.CS.2 在每次建置後，會在程式碼成品上簽名](#)
- [軟體成品的供應鏈層級 \(SLSA\)](#)

相關範例：

- [配合有效管控，加速在 AWS 的部署](#)
- [使用 CodeArtifact 套件來源控制工具組加強您的套件安全](#)
- [多重區域套件發行管道 \(GitHub\)](#)
- [使用 AWS CodePipeline 在 AWS CodeArtifact 上發行 Node.js 模組 \(GitHub\)](#)
- [AWS CDK CDK Java CodeArtifact 管道範例 \(GitHub\)](#)
- [使用 AWS CodeArtifact 分發私有 .NET NuGet 套件 \(GitHub\)](#)

相關影片：

- [預防性安全：考量與方法](#)

- [安全的 AWS 原則 \(re:Invent 2017\)](#)
- [當安全性、安全和緊迫性都很重要時：處理 Log4Shell](#)

SEC11-BP06 以程式設計方式部署軟體

盡可能以程式設計方式進行軟體部署。此方法可減少部署失敗或因人為疏失而發生非預期問題的機率。

預期結果：您測試的工作負載版本是您部署的版本，且每次都會一致地執行部署。您可以將工作負載的組態外部化，如此有助於在無變動的情況下部署到不同的環境。您可以使用軟體套件的加密簽章來驗證環境之間沒有任何變化。

常見的反模式：

- 手動部署軟體至生產環境。
- 手動執行因應不同環境需求的軟體變更。

建立此最佳實務的優勢：

- 提高軟體發佈程序的可信度。
- 降低變更失敗影響到業務功能的風險。
- 因變更風險降低而增加發佈規律。
- 部署其間意外事件的自動回復能力。
- 可以密碼編譯方式證明所測試的軟體就是實際部署的軟體。

未建立此最佳實務時的曝險等級：高

實作指引

要維護強大且可靠的應用程式基礎設施，請實作安全且自動化的部署實務。此實務與從生產環境移除持續的人工存取、使用 CI/CD 工具進行部署，以及將環境特定的組態資料外化相關。遵循此方法可以增強安全性、降低人為錯誤的風險，並可簡化部署程序。

您可以建置您的 AWS 帳戶結構，以從生產環境中移除持續的人力存取。此做法可將未經授權的變更或意外修改的風險降至最低，進而改善生產系統的完整性。您可以使用 [AWS CodeBuild](#) 和 [AWS CodePipeline](#) 等 CI/CD 工具來執行部署，而不是直接以人力存取。您可以使用這些服務來自動化建置、測試和部署處理程序，進而減少手動介入並提高一致性。

若要進一步增強安全性和可追蹤性，您可以在測試後簽署應用程式套件，並在部署期間驗證這些簽署。若要這麼做，請使用密碼編譯工具，例如 [AWS Signer](#) 或 [AWS Key Management Service \(AWS KMS\)](#)。您可以透過簽署和驗證套件，確保僅將已授權和已驗證的程式碼部署到您的環境。

此外，您的團隊可以建立工作負載架構，以從外部來源取得環境特定的組態資料，例如 [AWS Systems Manager Parameter Store](#)。此做法會將應用程式的程式碼與組態資料分開，這樣就能協助您獨立管理和更新組態，而無需修改應用程式的程式碼本身。

若要簡化基礎設施佈建和管理，請考慮使用基礎設施即程式碼 (IaC) 工具，例如 [AWS CloudFormation](#) 或 [AWS CDK](#)。您可以使用這些工具將基礎設施定義為程式碼，以提高跨不同環境部署的一致性和可重複性。

考慮採用 Canary 部署，驗證軟體是否已成功部署。Canary 部署需要在部署至整個生產環境之前，對執行個體或使用子集進行變更。然後，您可以監控變更造成的影響，並視需要復原，將常發生的問題風險降至最低。

請遵循[使用多個帳戶組織您的 AWS 環境](#)白皮書中概述的建議。此白皮書提供有關將環境 (例如開發、預備和生產) 分隔為不同 AWS 帳戶的指引，進一步增強安全性和隔離性。

實作步驟

1. 設定 AWS 帳戶 結構：

- 遵循[使用多個帳戶組織您的 AWS 環境](#)白皮書中的指引，針對不同的環境 (例如開發、預備和生產) 建立不同的 AWS 帳戶。
- 為每個帳戶設定適當的存取控制和許可權限，以限制人力對生產環境的直接存取。

2. 實作 CI/CD 管道：

- 使用 [AWS CodeBuild](#) 和 [AWS CodePipeline](#) 等服務來設定 CI/CD 管道。
- 設定管道以自動建置、測試和部署應用程式的程式碼至個別環境中。
- 將程式碼儲存庫與 CI/CD 管道整合，以便於進行版本控制和程式碼管理。

3. 簽署和驗證應用程式套件：

- 測試和驗證應用程式套件後，可使用 [AWS Signer](#) 或 [AWS Key Management Service \(AWS KMS\)](#) 簽署應用程式套件。
- 設定部署程序，以在將應用程式套件部署到目標環境之前驗證其簽章。

4. 外部化組態資料：

- 在 [AWS Systems Manager Parameter Store](#) 中儲存環境特定的組態資料。
- 修改應用程式的程式碼，以便在部署或執行時期從參數存放區擷取組態資料。

5. 基礎設施即程式碼 (IaC)：

- 使用 [AWS CloudFormation](#) 或 [AWS CDK](#) 等 IaC 工具，以定義基礎設施即程式碼，並加以管理。
- 建立 CloudFormation 範本或 CDK 指令碼，以佈建和設定應用程式的必要 AWS 資源。
- 將 IaC 與您的 CI/CD 管道整合，以自動部署基礎設施變更，以及應用程式的程式碼變更。

6. 實作 Canary 部署：

- 將部署程序設定為支援 Canary 部署，其會在您將變更部署到整個生產環境之前，先推廣到執行個體或使用者子集。
- 使用 [AWS CodeDeploy](#) 或 [AWS ECS](#) 等服務管理 Canary 部署，並監控變更造成的影響。
- 如果在 Canary 部署期間偵測到問題，請實作復原機制以還原至先前的穩定版本。

7. 監控和稽核：

- 設定監控和記錄機制，以追蹤部署、應用程式效能和基礎設施變更。
- 使用 [Amazon CloudWatch](#) 和 [AWS CloudTrail](#) 等服務來收集和分析日誌和指標。
- 實作稽核與合規檢查，確認是否遵循安全性最佳實務及法規要求。

8. 持續改善：

- 定期審查和更新您的部署實務，並納入對於先前部署的意見回饋，和從中學到的經驗。
- 盡可能自動化部署程序，以減少手動介入和潛在的人為錯誤。
- 與跨部門團隊合作 (例如，維運或資安)，以協調並持續改善部署實務。

遵循這些步驟，您可以在AWS您的環境中實作安全且自動化的部署實務，藉此增強安全性、降低人為錯誤的風險，並簡化部署程序。

資源

相關的最佳實務：

- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)
- [DL.CI.2 觸發程序會在原始碼修改後自動建置](#)

相關文件：

- [AWS CI/CD 研討會](#)
- [配合有效管控，加速在 AWS 的部署](#)
- [自動化安全、無人為介入的部署](#)

- [使用 AWS Certificate Manager Private CA 和 AWS Key Management Service 非對稱金鑰的程式碼簽署](#)
- [程式碼簽署，AWS Lambda 的信任和完整性控制](#)

相關影片：

- [無人為介入：Amazon 的自動化持續交付管道](#)

相關範例：

- [使用 AWS Fargate 的藍/綠部署](#)

SEC11-BP07 定期評估管道的安全屬性

採用 Well-Architected 安全原則保護您的流程，特別注意權限的區隔。定期評估管道基礎設施的安全屬性。有效管理管道的安全，就能讓您的軟體通過管道中重重的安全性考驗。

預期成果：用於建置與部署軟體的管道，皆遵循針對環境中其他任何工作負載所建議的相同做法。您在管道中實作的測試，無法由使用管道的團隊編輯。您只提供使用臨時憑證進行部署時所需的許可權給管道。您可以實作防護措施，以避免將管道部署到錯誤的環境。您可以將管道設定為發出狀態，如此才能驗證建置環境的完整性。

常見的反模式：

- 安全測試可能遭建置人員避開。
- 部署管道的權限過於廣泛。
- 管道未設定進行輸入驗證。
- 未定期審查與 CI/CD 基礎設施關聯的許可。
- 使用長期有效或硬式編碼的登入資料。

建立此最佳實務的優勢：

- 經由此類管道完成建置與部署的軟體完整性具備更高的可信度。
- 可在發現可疑活動時停止部署作業。

未建立此最佳實務時的風險暴露等級：高

實作指引

您的部署管道是軟體開發生命週期的重要元件，應遵循與環境中其他任何工作負載相同的安全原則和做法。這包括實作適當的存取控制、驗證輸入，以及定期審查與稽核與您的 CI/CD 基礎設施相關聯的許可權。

確認負責建置和部署應用程式的團隊，無法編輯或略過管道中實作的安全測試和檢查。這種將有疑慮的分隔的做法，有助於維護建置和部署程序的完整性。

剛開始，請考慮採用 [AWS 部署管道參考架構](#)。要在 AWS 上建置 CI/CD 管道，此參考架構提供了安全且可擴展的基礎。

此外，您可以使用 [AWS Identity and Access Management Access Analyzer](#) 等服務，為兩種管道許可產生最低權限 IAM 政策，並做為管道中用於驗證工作負載許可權的步驟。這有助於確認您的管道和工作負載僅擁有其特定功能所需的必要許可權，以降低未經授權的存取或動作風險。

實作步驟

- 從 [AWS 部署管道參考架構](#) 開始行動。
- 考慮使用 [AWS IAM Access Analyzer](#)，以程式設計方式產生管道的最低權限 IAM 政策。
- 整合您的管道與監控與提醒，以便您可在 AWS 受管服務 [Amazon EventBridge](#) 發生意外或異常活動時收到通知，這樣您就可以將資料路由至 [AWS Lambda](#) 或 [Amazon Simple Notification Service](#) (Amazon SNS) 等目的地。

資源

相關文件：

- [AWS 部署管道參考架構](#)
- [監控 AWS CodePipeline](#)
- [AWS CodePipeline 的安全最佳實務](#)

相關範例：

- [DevOps 監控儀表板](#) (GitHub)

SEC11-BP08 打造在工作負載團隊中納入安全所有權的計畫

打造一項計畫或一種機制，賦予建置人員團隊能對其本身所建軟體做出安全決策的能力。您的安全團隊仍需在審查過程中確認這些決策，但是讓建置人員團隊與生俱來擁有安全決策權，就能建置更快速、更安全的工作負載。這項機制也能推動所有權文化，積極影響您所建置系統的運作。

預期成果：您在團隊中已內嵌安全性擁有權和決策。您已針對如何考慮安全性培訓您的團隊，或已透過內嵌或關聯的資安人員來強化這些人的團隊。如此您的團隊才能在開發週期的早期，做出更優質的資安決策。

常見的反模式：

- 將所有的安全決策全部留給安全團隊。
- 未及早在開發程序初期解決安全需求。
- 未諮詢建置人員與安全部門人員在計畫運作方面的意見回饋。

建立此最佳實務的優勢：

- 縮短完成安全審核的時間。
- 減少必須在安全審核階段中偵測出的安全問題。
- 改善所編寫軟體的整體品質。
- 有機會識別並了解具備高度改善價值的系統性問題或區域。
- 減少因安全審核調查結果而必須進行的重新作業量。
- 改善對於安全功能的感覺。

未建立此最佳實務時的曝險等級：低

實作指引

從 [SEC11-BP01 應用程式安全訓練](#) 的指引開始。接著識別您認為最適合組織的計畫操作模式。其中兩種主要模式分別是訓練建置人員，以及在建置人員團隊當中納入安全部門人員。在您決定初步方法之後，您應該透過單一或小組型工作負載團隊進行先行試驗，證明該模式適合您的組織。建置人員與組織的安全部門所提供的領導支援，有助於計畫達成與成功實施。隨著這項計畫不斷建置，您一定要選擇可以用來顯示計畫價值的矩陣。了解 AWS 如何解決這個問題可以學到相當多知識。這項最佳實務非常強調組織層面變更與文化。您所使用的工具應該能支援建置人員與安全社群之間的協作。

實作步驟

- 從訓練建置人員處理應用程式安全開始。
- 建立專為教育建置人員的社群和上線計畫。
- 挑選計畫名稱。守門人、擁護者或倡導者是常見手法。
- 識別要應用的模式：訓練建置人員、納入安全部門工程師，或是安排親和性安全角色。
- 從安全部門、建置人員和可能的其他相關小組當中，識別專案贊助者。
- 計畫當中所涉多人的追蹤矩陣，審核所花時間，以及建置人員與安全團隊人員的意見回饋。使用這些矩陣來達成改善。

資源

相關的最佳實務：

- [SEC11-BP01 應用程式安全訓練](#)
- [SEC11-BP02 自動化在整個開發和發布生命週期的測試](#)

相關文件：

- [如何建立威脅模型](#)
- [如何思考雲端安全管控](#)
- [AWS 如何建置 Security Guardians 程式的這種分散安全所有權的機制](#)
- [如何建置 Security Guardians 程式以分散安全所有權](#)

相關影片：

- [預防性安全：考量與方法](#)
- [來自 AWS 和 Toyota Motor North America 的 AppSec 工具和文化秘訣](#)

結論

安全是一項持續的工作。發生事件時，應將其視為提高架構安全的機會。擁有強大的身分控制、自動化對安全事件的回應、在多個層級保護基礎設施，並且使用加密管理妥善分類的資料，可提供每個組織皆應實作的深度防禦。由於本文討論的程式設計函數和 AWS 功能和服務，因此這項工作更容易。

AWS 致力於協助您建置和操作保護資訊、系統和資產的架構，同時提供商業價值。

貢獻者

下列個人和組織為本文件作出了貢獻：

- Jay Michael , Amazon Web Services 首席安全主管解決方案架構師
- Kiaan Sumeet , Amazon Web Services 首席安全顧問
- Michael Fischer , Amazon Web Services 首席解決方案架構師
- Conor Colgan , Amazon Web Services 首席解決方案架構師
- Dave Walker , Amazon Web Services 安全與合規首席解決方案架構師
- Patrick Palmer , Amazon Web Services 安全與合規首席解決方案架構師
- Monka Vu Minh , Amazon Web Services 安全顧問
- Kurt Kumar , Amazon Web Services 安全顧問
- Fahima Khan , Amazon Web Services 安全解決方案架構師
- Mutaz Hajeer , Amazon Web Services 資深安全解決方案架構師
- Luis Pastor , Amazon Web Services 資深安全解決方案架構師
- Colin Igbokwe , Amazon Web Services 資深安全解決方案架構師
- Geoff Sweet , Amazon Web Services 資深安全解決方案架構師
- Anthony Harvey , Amazon Web Services 資深安全解決方案架構師
- Sowjanya Rajavaram , Amazon Web Services 資深安全解決方案架構師
- Krishna Prasad , Amazon Web Services 資深解決方案架構師
- Faisal Farooq , Amazon Web Services 資深解決方案架構師
- Arun Krishnaswamy , Amazon Web Services 資深解決方案架構師
- Dan Girard , Amazon Web Services 資深解決方案架構師
- Marc Luescher , Amazon Web Services 資深解決方案架構師
- Kyle Nicodemus , Amazon Web Services 資深技術客戶經理
- Irina Szabo , Amazon Web Services 資深技術客戶經理
- Arun Sivaraman , Amazon Web Services 解決方案架構師
- Stephen Novak , Amazon Web Services 技術客戶經理
- Jonathan Risbrook , Amazon Web Services 技術客戶經理
- Freddy Kasprzykowski , Amazon Web Services 全球金融服務實務經理
- Pat Gaw , Amazon Web Services 首席安全顧問

- Jason Garman , Amazon Web Services 首席安全解決方案架構師
- Mark Keating , Amazon Web Services 首席安全解決方案架構師
- Zach Miller , Amazon Web Services 首席安全解決方案架構師
- Maitreya Ranganath , Amazon Web Services 首席安全解決方案架構師
- Reef Dsouza , Amazon Web Services 首席解決方案架構師
- Brad Burnett , Amazon Web Services 安全解決方案架構師
- Matt Saner , Amazon Web Services 安全解決方案架構資深經理
- Priyank Ghedia , Amazon Web Services 資深安全解決方案架構師
- Arthur Mnev , Amazon Web Services 資深安全解決方案架構師
- Kyle Dickinson , Amazon Web Services 資深安全解決方案架構師
- Kevin Boland , Amazon Web Services 資深安全解決方案架構師
- Anna McAbee , Amazon Web Services 資深安全解決方案架構師
- Recep Meric Degirmenci , Amazon Web Services 資深安全解決方案架構師
- Daniel Salzedo , Amazon Web Services 資深安全技術產品經理
- Jake Izumi , Amazon Web Services 資深解決方案架構師
- Bert Bullough , Amazon Web Services 資深解決方案架構師
- Robert McCall , Amazon Web Services 解決方案架構師
- Angela Chao , Amazon Web Services AWS Enterprise Support ESL TAM
- Pratima Singh , Amazon Web Services 資深 ANZ 安全專家。解決方案架構師
- Amazon Web Services AWS Security , CISO 辦公室負責人 Darran Boyd
- Amazon Web Services 資深安全解決方案架構師 Byron Pogson

深入閱讀

如需其他協助，請參考以下資源：

- [AWS Well-Architected Framework 白皮書](#)
- [AWS 架構中心](#)

文件修訂

若要收到此白皮書更新的通知，請訂閱 RSS 摘要。

變更	描述	日期
已更新最佳實務指引	最佳實務已透過下列領域的新指引進行更新：SEC 2、SEC 3、SEC 4、SEC 6、SEC 7、SEC 8、SEC 9、SEC 10 和 SEC 11。跨支柱更新和精簡指引。	2024 年 11 月 6 日
已更新最佳實務指引	整個支柱進行了大規模最佳實務更新。重新排序和合併了多個最佳實務。SEC 1、4、5、6、7、8 和 9 中的重大變更。	2024 年 6 月 27 日
已更新最佳實務指引	最佳實務已更新，納入了以下方面的新指引： 安全操作工作負載 和 保護傳輸中的資料 。	2023 年 12 月 6 日
已更新最佳實務指引	對 事件回應 的指引和最佳實務進行了重大更新。 準備 中更新了多個最佳實務。對事件回應新增了兩個新領域： 操作 和 事後處理 。已新增最佳實務 SEC10-BP08 建立從事件中學習的架構 。	2023 年 10 月 3 日
已更新最佳實務指引	最佳實務已更新，納入了以下方面的新指引： 準備和模擬 。	2023 年 7 月 13 日
新框架的更新。	最佳實務已更新，納入了規範性指引，並增加了新的最佳	2023 年 4 月 10 日

	實務。已新增應用程式安全 (AppSec) 的新最佳實務領域。	
白皮書已更新	最佳實務更新了新的實作指引。	2022 年 12 月 15 日
白皮書已更新	已擴充最佳實務並新增了改善計畫。	2022 年 10 月 20 日
次要更新	已更新 IAM 資訊以反映目前最佳實務。	2022 年 6 月 28 日
次要更新	已新增額外的 AWS PrivateLink 資訊，並更正了中斷的連結。	2022 年 5 月 19 日
次要更新	新增了 AWS PrivateLink。	2022 年 5 月 6 日
次要更新	已移除非包容性語言。	2022 年 4 月 22 日
次要更新	已新增 VPC 網路存取分析器的相關資訊。	2022 年 2 月 2 日
次要更新	修正了中斷的連結。	2021 年 5 月 27 日
次要更新	整體的編輯變更。	2021 年 5 月 17 日
主要更新	已新增有關管控的章節、已將詳細資訊新增至不同的章簡，已在整體中加入新的功能和服務。	2021 年 5 月 7 日
次要更新	已更新連結。	2021 年 3 月 10 日
次要更新	修正了中斷的連結。	2020 年 7 月 15 日
新框架的更新	更新帳戶、身分和許可管理的指引。	2020 年 7 月 8 日
新框架的更新	更新以擴展每個領域的建議、新的最佳實務、服務和功能。	2020 年 4 月 30 日

白皮書已更新	更新以反映新的 AWS 服務和功能以及更新的參考。	2018 年 7 月 1 日
白皮書已更新	已更新「系統安全組態和維護」一節，以反映新的 AWS 服務和功能。	2017 年 5 月 1 日
初次出版	安全支柱 – AWS Well Architected Framework 已發布。	2016 年 11 月 1 日

注意

客戶有責任對本文件中的資訊進行自己的獨立評定。本文件：(a) 僅供參考，(b) 代表目前的 AWS 產品和實務，這些產品和實務可能隨時變更，恕不另行通知，以及 (c) 不會從 AWS 及其關聯機構、供應商或授權方建立任何承諾或保證。AWS 產品或服務「原樣」提供，無論明示還是暗示，均無任何保證、陳述或條件。AWS 對其客戶的責任和責任受 AWS 協議控制，本文件不屬於與其 AWS 客戶之間的任何協議，也未對其進行修改。

© 2023 Amazon Web Services, Inc. 或其附屬公司。保留所有權利。

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](#)。