使用者指南

AWS 安全代理程式

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS 安全代理程式: 使用者指南

Copyright © 2026 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、任從何貶低或使 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，這些擁有者可能隸屬於 Amazon，或與 Amazon 有合作關係，或由 Amazon 贊助。

Table of Contents

簡介 1
什麼是 AWS Security Agent？ 1
關鍵功能 1
優勢 2
AWS Security Agent 功能 3
AWS Security Agent 的運作方式 3
概觀 4
了解資源階層和生命週期 7
在您的組織中共用的內容 7
每個客服人員空間的範圍 7
GitHub 儲存庫如何符合階層 9
安全功能之間的主要差異 9
了解資源關係 10
開始使用 12
註冊 AWS 12
選擇您的路徑 12
快速入門：執行滲透測試 13
先決條件 13
步驟 1：在 AWS 主控台中設定 AWS 安全代理程式 14
步驟 2：啟用滲透測試並驗證您的網域 14
步驟 3：連接原始程式碼 （選用） 15
步驟 4：建立並執行滲透測試 15
步驟 5：檢閱滲透測試問題清單 16
快速入門：執行程式碼檢閱 17
先決條件 13
步驟 1：在 AWS 主控台中設定 AWS Security Agent 14
步驟 2：啟用和設定程式碼檢閱 18
步驟 3：建立和執行程式碼檢閱 19
步驟 4：檢閱程式碼檢閱問題清單 20
快速入門：執行威脅模型 20
步驟 1：在 AWS 主控台中設定 AWS Security Agent 14
步驟 2：連接原始程式碼 21
步驟 3：建立並執行威脅模型 22
步驟 4：檢閱威脅 22

適用於管理員 （主控台） 24
範例組態 24
設定和建立客服人員空間 24
設定 AWS 安全代理程式 24
建立 代理程式空間 29
連線整合 31
整合如何與 Agent Spaces 搭配使用 31
將 AWS 安全代理程式連線至 GitHub 儲存庫 32
移除 GitHub 整合 38
將 AWS 安全代理程式連線至 GitLab 儲存庫 39
將 AWS 安全代理程式連線至 GitLab 自我管理 43
移除 GitLab 整合 46
將 AWS 安全代理程式連線至 GitHub Enterprise Server 46
移除 GitHub Enterprise Server 整合 50
尋找您的 Atlassian 安裝 ID 51
將 AWS Security Agent 連線至 Bitbucket 儲存庫 51
移除 Bitbucket 整合 55
將 AWS 安全代理程式連線至 Confluence 56
移除 Confluence 整合 59
連線至私有託管來源控制 60
將代理程式連線至私有 VPC 資源 65
設定 功能 68
啟用滲透測試 68
啟用 GitHub 儲存庫的提取請求程式碼檢閱 73
啟用程式碼檢閱 73
啟用威脅建模 79
讓使用者能夠開始修復滲透測試和程式碼檢閱問題清單 82
從 S3 儲存貯體提供代理程式資源 84
啟用用於滲透測試的應用程式網域 84
用於滲透測試的受管目標網域 86
管理安全需求 88
管理安全需求 88
AWS 受管套件 93
從文件產生安全需求 94
準備文件以產生需求 96
管理使用者和存取 97

授予使用者存取 AWS Security Agent Web 應用程式的權限 97
為 AWS 安全代理程式建立 IAM 角色 99
客戶自管金鑰 106
疑難排解 127
拒絕存取：選取不正確的 GitHub 帳戶類型或指定的組織名稱不正確 127
拒絕存取：在組織中安裝 GitHub App 的許可不足 128
客服人員無法在滲透測試期間連線至端點 128
取得其他協助 129
適用於使用者和開發人員 (Web 應用程式和 IDE) 130
登入 AWS Security Agent Web 應用程式 130
存取方法 130
使用 IAM Identity Center (SSO) 登入 130
使用管理員存取登入 (IAM) 132
建立設計檢閱 133
先決條件 13
步驟 1：開始建立設計檢閱 133
步驟 2：為您的設計檢閱命名 133
步驟 3：上傳設計檔案 134
步驟 4：啟動設計檢閱 134
後續步驟 28
檢閱設計檢閱的問題清單 135
建立威脅模型 139
先決條件 13
AWS Security Agent 如何執行威脅模型 140
存取威脅模型頁面 140
建立威脅模型 140
執行威脅模型 142
監控威脅模型執行 143
編輯威脅模型 144
刪除威脅模型 144
後續步驟 28
檢閱來自威脅模型的威脅 145
檢閱提取請求中的程式碼安全性問題清單 149
程式碼檢閱在提取請求中的運作方式 150
了解程式碼檢閱結果 150
回應安全性問題清單 151

篩選程式碼檢閱問題清單 151
後續步驟 28
建立程式碼檢閱 153
先決條件 13
存取程式碼檢閱頁面 154
建立程式碼檢閱 154
執行程式碼檢閱 159
監控程式碼檢閱執行 160
後續步驟 28
檢閱程式碼檢閱的問題清單 161
修正程式碼檢閱問題清單 167
使用 S3 執行差異碼掃描 169
差異掃描的運作方式 170
先決條件 13
步驟 1：產生 diff 檔案 170
步驟 2：將差異上傳至 S3 171
步驟 3：啟動差異碼檢閱任務 171
步驟 4：監控掃描 172
步驟 5：檢閱問題清單 173
配額和限制 173
後續步驟 28
從 IDE 執行程式碼安全掃描 174
IDE 整合的運作方式 174
先決條件 13
安裝 MCP 伺服器 175
首次設定 176
執行完整的安全性掃描 177
執行差異掃描 178
執行威脅模型檢閱 178
檢閱調查結果 179
套用自動修正 179
自動掃描建議 (Kiro) 180
可用的掃描類型 180
環境變數 181
疑難排解 181
配額和限制 173

後續步驟 28
建立滲透測試 182
先決條件 13
開始建立滲透測試 182
為您的滲透測試命名 183
設定滲透測試範圍 183
設定 IAM 角色 186
自動程式碼修復 168
設定 VPC 資源 （選用） 187
設定身分驗證憑證 （選用） 188
連接其他資源 （選用） 190
建立滲透測試 193
檢閱滲透測試的問題清單 194
提供用於滲透測試的身分驗證憑證 203
修復滲透測試問題清單 208
安全性和參考 210
安全 210
安全考量 210
AWS 受管政策 216
資料保護 218
身分與存取管理 221
事件回應 233
法規遵循驗證 234
恢復能力 235
基礎設施安全性 236
介面 VPC 端點 236
組態與漏洞分析 240
預防跨服務混淆代理人 241
安全最佳實務 241
IAM 動作和資源遷移 244
使用 CloudTrail 進行記錄 248
Service Quotas 250
操作配額 250
組態配額 251
文件歷史紀錄 251
..... ccliv

簡介

了解 AWS Security Agent 的功能、運作方式，以及其資源如何結合在一起。

什麼是 AWS Security Agent？

AWS Security Agent 是前端代理程式，可在整個開發生命週期中主動保護您的應用程式。它會根據您的組織需求進行自動化安全性審查、建置威脅模型以識別您的應用程式如何遭到攻擊，以及隨需提供內容感知滲透測試。透過持續驗證從設計到部署的安全性，AWS Security Agent 有助於防止所有環境中的早期漏洞。

安全團隊會在 AWS 主控台中定義組織安全需求一次：核准的授權程式庫、記錄標準和資料存取政策。AWS Security Agent 會在開發過程中自動強制執行這些安全要求，根據您的標準評估架構文件和程式碼，並在偵測到違規時提供特定指引。這可為所有團隊的設計和程式碼檢閱提供一致的安全性強制執行。

為了進行部署驗證，AWS Security Agent 會將滲透測試從定期瓶頸轉換為隨需功能。安全團隊提供目標 URLs、身分驗證詳細資訊、原始程式碼和應用程式文件。AWS Security Agent 開發對應用程式的深入理解，並執行複雜的攻擊鏈來探索和驗證漏洞，讓團隊在需要時進行測試。

關鍵功能

AWS Security Agent 提供涵蓋整個開發生命週期的完整安全功能。

設計安全審查

AWS Security Agent 會在撰寫任何程式碼之前，提供設計文件的即時安全意見回饋，並評估組織安全需求的合規性，藉此轉移剩餘的安全性。安全團隊會透過 Web 應用程式上傳文件，並接收修補指引，以排定問題清單的優先順序，從而加速耗時的手動檢閱以專注於分析。透過主動將您的安全標準嵌入到每個設計審查中，您可以減少後期架構重做，並與多個開發團隊保持同步。

威脅建模

AWS Security Agent 會建置應用程式的威脅模型，以識別其可能遭到攻擊的方式。提供技術設計文件（範圍文件）來定義代理程式的重點，將原始程式碼做為內容，讓代理程式了解您現有的系統，或兩者。AWS Security Agent 會產生系統概觀，描述您應用程式的架構、元件、信任界限、資料流程和安全狀態，以及一組依 STRIDE 類別分類的威脅（詐騙、竄改、複寫、資訊揭露、拒絕服務、提升權限），以及嚴重性等級和可行的建議。威脅模型是可重複使用的組態，您可以在程式碼和設計演進時重新執行，在整個開發生命週期中啟用反覆式安全評估。

程式碼安全性審查

AWS Security Agent 會透過兩種互補方法主動保護應用程式。首先，您可以在 Web 應用程式中建立程式碼檢閱，以完整掃描 GitHub、GitLab、Bitbucket 或 GitHub Enterprise Server 儲存庫或 S3 儲存貯體的完整原始程式碼，識別安全漏洞，並驗證整個程式碼庫中組織需求的合規性。其次，您可以啟用連線儲存庫的自動提取請求分析，其中 AWS Security Agent 會檢閱程式碼變更，並直接發佈安全調查結果做為提取請求或合併請求評論。在這兩種情況下，開發人員都會收到修補指引，而 AWS Security Agent 可以自動產生提取請求，或使用已識別漏洞的程式碼修正來合併請求。這會將安全專業知識嵌入所有儲存庫，以減少開發管道中的安全相關延遲，並擴展所有程式碼庫的評估。

滲透測試

AWS Security Agent 透過部署專門的 AI 代理器來提供隨需滲透測試，以透過量身打造的多步驟攻擊案例來探索、驗證、報告和修復安全漏洞。代理程式會透過分析原始程式碼和文件來了解您應用程式的內容，以識別和利用自動化安全掃描工具找不到的漏洞。它會記錄具有影響分析、可重現的攻擊路徑的問題清單，並使用ready-to-implement的程式碼修正來建立提取請求，將定期評估轉換為持續驗證，該驗證會跨所有應用程式擴展，而不是僅限於關鍵的應用程式。

優勢

隨需可存取性

AWS Security Agent 可讓您立即存取安全專業知識。開發團隊可以視需要執行設計審查、威脅模型、程式碼分析和滲透測試，以符合現代開發週期的速度，並在每個階段啟用主動安全性。

自動驗證組織需求

在 AWS 主控台中定義組織的安全需求一次。AWS Security Agent 會在每一次設計和程式碼安全審查期間自動驗證所有應用程式的這些需求，確保團隊滿足您的特定需求，而不是一般檢查清單。

擴展安全專業知識

AWS Security Agent 透過自動化組織安全要求的強制執行來擴展安全審查，以符合開發速度。集中設定需求、使用調查結果分析進行全面審查，並透過 AWS 主控台管理整個組織的滲透測試範圍。

已確認漏洞的可行修正

AWS Security Agent 透過以證據為基礎的入侵來驗證滲透測試期間找到的安全性問題清單、提供可重現的入侵路徑、全面的影響分析，並以開發人員易用的語言建立ready-to-implement的提取請求。這有助於團隊專注於合法的高影響安全風險，而不會浪費時間處理誤報。

多雲端和混合雲端支援

AWS Security Agent 可跨 AWS、內部部署、混合、多雲端和 SaaS 環境運作，無論您的基礎設施或平台選擇為何，都能確保一致的安全指導和測試。

AWS Security Agent 功能

AWS Security Agent 在整個開發生命週期中提供四個核心安全功能：

- 設計安全審查 — 審查設計和架構文件，以識別安全風險。透過 Web 應用程式上傳文件，服務會根據您的組織安全需求來分析文件。AWS Security Agent 會評估是否符合您定義的安全需求，例如核准的授權程式庫、記錄標準和資料存取政策。這可協助您在開發程序的早期發現不安全的設計和政策違規。
- 威脅建模 — 建立應用程式的威脅模型，以識別如何遭到攻擊。提供設計文件做為範圍文件，以定義分析的焦點、將原始程式碼做為代理程式的內容，以了解您現有的系統，或同時定義兩者。AWS Security Agent 會產生系統概觀，描述您應用程式的架構、元件、信任界限、資料流程和安全狀態，以及一組依 STRIDE 類別分類的威脅（詐騙、竄改、複寫、資訊揭露、拒絕服務、提升權限），其中包含嚴重性等級和可行的建議。每個威脅都會連結回原始程式碼中的證據。
- 程式碼安全審查 — 分析儲存庫和 S3 來源中的程式碼，以識別跨語言、架構和架構的安全漏洞和違反組織安全要求。在 Web 應用程式中建立程式碼檢閱，以執行完整原始碼的完整掃描，或啟用提取請求註解，以自動檢閱 GitHub 中的程式碼變更。AWS Security Agent 提供特定的修補指導，並可透過已識別漏洞的程式碼修正來自動產生提取請求。這可確保在所有開發團隊中一致強制執行您的安全政策。
- 隨需滲透測試 — 透過量身打造的多步驟攻擊案例，探索、驗證、報告和修復即時 Web 應用程式和 APIs 中的安全漏洞。透過指定測試範圍、身分驗證詳細資訊和資源，設定服務透過 Web 應用程式建立 pentest。AWS Security Agent 會從提供的原始程式碼和文件中開發應用程式內容，並執行複雜的攻擊鏈，以識別靜態分析和傳統工具遺漏的可利用漏洞。它還提供ready-to-implement程式碼修正，並直接在您的程式碼儲存庫中建立提取請求，讓您更快地解決漏洞。

AWS Security Agent 的運作方式

AWS Security Agent 可跨三個界面運作，在整個開發生命週期中提供主動的應用程式安全性。安全組態是在 AWS 管理主控台中管理的，安全性審查是在 Security Agent Web 應用程式中執行的，而自動化程式碼和安全調查結果修復直接發生在 GitHub 等程式碼儲存庫平台中。

概觀

AWS Security Agent 包含三個主要元件：

- AWS 管理主控台 - 設定客服人員空間、定義安全需求、設定滲透測試、連接程式碼儲存庫和管理使用者存取
- Security Agent Web Application - 執行設計檢閱、建立和執行威脅模型、執行滲透測試、建立和執行程式碼檢閱，以及檢閱指派的 Agent Spaces 內的安全調查結果
- 程式碼儲存庫整合 - 接收提取請求和滲透測試修復提取請求的自動程式碼檢閱。目前，AWS Security Agent 支援連線至 GitHub。

您可以在三個角色之一中使用 AWS Security Agent，您可以透過使用的界面來識別這些角色：

Role	您工作的地方，以及您的工作
管理員	適用於 AWS 管理主控台：設定 Agent Spaces、定義安全需求、設定功能和管理使用者存取。
使用者	適用於 Security Agent Web Application — 執行設計檢閱、威脅模型、滲透測試和程式碼檢閱，並檢閱產生的調查結果。
開發人員	適用於 GitHub 或 IDE （例如 Kiro 或 Claude Code) — 接收提取請求的自動安全調查結果，並在不離開開發環境的情況下執行程式碼掃描。

我們全文使用這些角色名稱來指出執行每個任務的人員。單一人員可以擁有多個角色。

主控台組態

管理員透過 AWS 管理主控台設定 AWS 安全代理程式。

Agent Spaces - 當您在 AWS 管理主控台中建立第一個 Agent Space 時，AWS Security Agent 會為您的帳戶建立 Web 應用程式。您建立的每個客服人員空間都代表您要保護的不同應用程式或專案。在 Web 應用程式中，使用者會在執行安全評估時選取要使用的客服人員空間。

安全需求 - 定義 AWS Security Agent 在設計和程式碼檢閱期間評估的安全需求，整理成安全需求套件。根據產業標準啟用 AWS 受管套件、為組織的政策建立自訂套件，或上傳您的安全文件來產生需求。要求適用於所有 代理程式空間。

滲透測試組態 - 設定每個代理程式空間的滲透測試功能，方法如下：

- 驗證目標網域以透過 DNS 或 HTTP 驗證進行測試
- 設定 VPC 存取以測試私有應用程式
- 設定 CloudWatch 記錄以進行滲透測試執行
- 為測試登入資料設定 AWS Secrets Manager 或 Lambda 函數
- 為其他應用程式內容指定 S3 儲存貯體

程式碼檢閱組態 - 設定每個客服人員空間的程式碼檢閱功能，方法如下：

- 連接 GitHub 儲存庫或包含原始碼的 S3 儲存貯體
- 選取程式碼檢閱設定（安全漏洞、自訂需求或兩者）
- 啟用提取請求註解，以在 GitHub 中自動檢閱程式碼變更
- 設定 CloudWatch 記錄以進行程式碼檢閱執行

威脅建模組態 - 透過下列方式為每個客服人員空間設定威脅建模功能：

- 連接原始碼儲存庫或包含原始碼的 S3 儲存貯體
- 新增範圍文件（功能設計文件）以專注於特定功能的分析
- 設定 CloudWatch 記錄以進行威脅模型執行
- 設定 AWS 資源存取的服務角色

整合 - 將 GitHub 儲存庫連接到每個客服人員空間，以啟用關鍵功能：

- 提供應用程式內容，以進行更準確的滲透測試
- 針對完整的儲存庫掃描和提取請求分析啟用程式碼檢閱
- 透過提取程式碼檢閱和滲透測試問題清單的請求來啟用自動程式碼修復

使用者存取 - 管理使用者存取 Security Agent Web Application 的方式。如果您已啟用 IAM Identity Center (SSO)，請在 AWS Security Agent 主控台中指派使用者，以提供 Web 應用程式的直接 SSO 存取。如果您使用的是僅限 IAM 存取，具有 AWS 主控台存取的使用者可以透過主控台中任何代理程式空間的管理員存取連結來啟動 Web 應用程式。

Web 應用程式活動

使用者存取 Security Agent Web 應用程式，在其指派的 Agent Spaces 內執行安全評估。

選取客服人員空間 - 登入 Web 應用程式時，使用者選擇要使用的客服人員空間。使用者只能查看和存取已指派給他們的 Agent Spaces。

設計檢閱 - 上傳設計文件和架構規格，以根據組織安全需求進行分析。使用修補指引檢閱問題清單。

威脅模型 - 透過提供原始程式碼、技術設計文件（範圍文件）或兩者來建立和執行威脅模型。範圍文件定義代理程式的分析重點；原始程式碼提供現有系統的內容。每次執行都會產生系統概觀，描述應用程式的架構、信任界限、資料流程和安全狀態，以及一組依 STRIDE 類別分類、具有嚴重性等級和可行性建議的威脅。

程式碼檢閱 - 建立並執行程式碼檢閱，以跨完整原始程式碼執行全面的靜態分析。選取 GitHub 儲存庫或 S3 來源、設定掃描設定，並使用修補指引檢閱詳細的安全性問題清單。AWS Security Agent 會識別安全漏洞，並驗證整個程式碼庫中組織自訂安全需求的合規性。

滲透測試 - 透過提供目標 URLs、身分驗證詳細資訊和文件來設定和執行滲透測試。AWS Security Agent 會執行自動測試，透過多步驟攻擊案例探索可利用的漏洞。

檢閱問題清單 - 檢查設計檢閱、威脅模型、程式碼檢閱和滲透測試的詳細安全性問題清單，包括影響分析、可重現的攻擊路徑和修補指導。

驗證修正 - 在實作修補以驗證漏洞已解決之後，重新執行安全評估。

GitHub 整合

AWS Security Agent 直接與 GitHub 整合，以在開發人員的工作流程中提供自動化安全意見回饋。

提取請求評論 - 在管理員安裝 AWS 安全代理程式 GitHub 應用程式並啟用程式碼檢閱與代理程式空間的程式碼檢閱評論之後，AWS 安全代理程式會自動分析連線儲存庫中的提取請求。開發人員會在提取請求評論中直接收到安全調查結果和修補指導，根據組織安全要求和常見漏洞驗證程式碼變更。

自動修復 - 當使用者為程式碼檢閱啟用自動程式碼修復時，AWS Security Agent 會針對已識別的漏洞產生修正，並將提取請求提交至相關聯的 GitHub 儲存庫。

滲透測試修復 - 當管理員在 主控台中啟用問題清單修復時，使用者可以從 Web 應用程式請求滲透測試問題清單的自動修復。AWS Security Agent 會開啟對相關聯 GitHub 儲存庫的提取請求，其中包含解決漏洞的程式碼修正。

了解資源階層和生命週期

AWS Security Agent 會在階層結構中組織安全測試資源，以決定跨組織共用的內容，以及每個應用程式的範圍。了解此結構可協助您有效地設定 AWS Security Agent，並知道在何處尋找和管理不同的資源。

在您的組織中共用的內容

AWS Security Agent 中的某些資源會在組織層級設定一次，並套用至您的所有應用程式和 Agent Spaces。這些租戶層級資源可提供一致性，並減少重複的組態工作。

資源	這是什麼	共用的原因
安全需求	定義 AWS Security Agent 在設計和程式碼檢閱期間驗證內容的組織安全標準	您的安全政策適用於所有應用程式。定義一次，AWS Security Agent 會在任何地方強制執行它們。
GitHub 整合	授權與 AWS Security Agent 連線的已註冊 GitHub 組織或使用者帳戶	註冊您的 GitHub 組織一次，然後視需要將特定儲存庫連接到任何 Agent Space。
IAM Identity Center 組態	控制使用者如何存取 AWS Security Agent 的 SSO 設定	集中式身分管理適用於組織中的所有 Agent Spaces。

> **Important**
>
> 安全需求的變更會影響所有 Agent Spaces 的所有未來設計審查和程式碼審查。現有的檢閱不會受到影響。

每個客服人員空間的範圍

每個客服人員空間代表您要保護的不同應用程式或專案。客服人員空間層級的資源範圍限定於該特定應用程式，可讓不同的團隊獨立使用自己的組態和評估。

資源	這是什麼	為什麼每個應用程式都有其範圍
滲透測試組態	測試應用程式內特定功能、API 端點或功能的組態	每個應用程式都有該應用程式特有的唯一目標、身分驗證方法和範圍界限。
設計檢閱	設計文件的個別架構安全評估	每個應用程式都有自己的架構和設計文件，這些文件會獨立評估。
威脅模型	建立系統概觀並識別來源碼、設計文件或兩者之威脅的威脅建模評估	每個應用程式都有自己的程式碼和設計，並獨立建立威脅模型。威脅模型是可重複使用的組態，您可以隨著程式碼和設計發展而重新執行。
整合	連接到此代理程式空間的來源和文件提供者 (GitHub、GitLab、Bitbucket、GitHub Enterprise Server 和 Confluence)	不同的應用程式依賴不同的來源和文件。在客服人員空間層級進行連線，可保持應用程式邊界的清晰。
程式碼檢閱設定	程式碼檢閱功能的組態，包括連線來源、掃描設定和 PR 註解啟用	每個應用程式都有自己的儲存庫和獨立設定的安全審查需求。
滲透測試修復設定	連線儲存庫可以接收滲透測試問題清單自動修正提取請求的組態	團隊會根據其應用程式的工作流程，控制 AWS Security Agent 可以提交程式碼變更的位置。
使用者指派	可存取此特定客服人員空間的使用者	團隊只會看到他們負責的應用程式的安全評估，保持工作井然有序且專注。

Tip

我們建議為每個應用程式或專案建立一個代理程式空間，以在團隊之間維持明確的界限，並有效地組織安全評估。

GitHub 儲存庫如何符合階層

GitHub 儲存庫透過將組織資源連接到特定應用程式的多步驟程序進行整合：

1. 在租戶層級註冊 - 為您的 GitHub 組織或使用者帳戶授權一次 AWS Security Agent GitHub 應用程式
2. 在客服人員空間層級連線 - 選取特定儲存庫以連線至每個客服人員空間
3. 設定每個儲存庫的用量 - 為每個連線的儲存庫啟用特定功能：
   - 程式碼檢閱 - 完整原始程式碼掃描和自動提取請求分析
   - 滲透測試內容 - 在滲透測試期間從原始程式碼了解應用程式
   - 自動程式碼修復 - 自動化提取請求，具有程式碼檢閱和滲透測試問題清單的漏洞修正

單一儲存庫可以連接到多個 Agent Spaces，並在每個 Spaces 中啟用不同的功能。

安全功能之間的主要差異

AWS Security Agent 中的每個安全功能會根據安全團隊如何使用它，遵循不同的工作流程模型。

滲透測試：具有獨立執行的可重複使用組態

滲透測試使用支援反覆安全性測試的configuration-and-run模型：

- 建立一次，執行多次 - 使用範圍界限、身分驗證和測試參數定義特定目標的組態 (API 端點、功能區域）
- 獨立執行 - 在您改善安全性時多次執行相同的組態。每個執行都是獨立的，並產生新的問題清單

當您開發和部署改善項目時，此模型支援持續的安全驗證。

設計審查：使用複製進行一次性評估

設計審查是不遵循可重複使用組態模型的獨立評估：

- 單一評估 - 每個設計檢閱都會根據組織的安全需求分析上傳的文件一次
- 無法重新執行 - 設計檢閱不可重複使用。您無法重新執行相同的檢閱
- 複製以進行更新 - 複製現有的設計檢閱，以使用預先載入的原始文件建立新的檢閱，讓您更新文件並執行新的分析

此模型支援point-in-time架構安全評估。

程式碼檢閱：具有隨需掃描和自動 PR 分析的可重複使用組態

程式碼檢閱提供兩種操作模式來保護原始程式碼：

- 完整程式碼檢閱 (Web 應用程式） - 建立程式碼檢閱組態，以選取 GitHub 儲存庫或 S3 來源，然後隨需執行全面掃描。每次執行都會對完整原始程式碼執行靜態分析，並使用修補指引產生問題清單。您可以重新執行與程式碼演進相同的程式碼檢閱組態。
- 提取請求註解 (GitHub) - 啟用連線 GitHub 儲存庫的自動分析。當提取請求標記為準備好進行檢閱時，AWS Security Agent 會自動檢閱提取請求，並直接在 GitHub 中將安全性調查結果發佈為註解。

兩種模式都使用您設定的程式碼檢閱設定（安全漏洞、自訂需求或兩者），並透過提取請求支援自動程式碼修復。

威脅模型：隨需執行的可重複使用組態

威脅模型使用支援架構反覆評估的configuration-and-run模型：

- 建立一次、執行多次 – 透過選取原始碼做為來源、上傳設計文件做為範圍文件，或兩者，來定義威脅模型。隨需執行，並在程式碼和設計演進時重新執行。
- 彈性輸入 – 僅對原始程式碼執行威脅模型、僅設計文件，或兩者皆執行。範圍文件定義代理程式的分析重點；原始程式碼提供有關現有系統的內容。
- 系統概觀和威脅 – 每次執行都會產生系統概觀，描述應用程式的架構、信任界限、資料流程和安全狀態，以及一組依 STRIDE 類別分類且具有嚴重性、證據和可行建議的威脅。

了解資源關係

階層會決定您設定和存取不同資源的位置：

在 AWS 管理主控台中：

- 設定租戶層級資源（安全需求、GitHub 整合、IAM Identity Center)
- 建立和管理客服人員空間
- 設定 Agent Space 設定（已連線的儲存庫、程式碼檢閱啟用、滲透測試修復）

在 Security Agent Web 應用程式中：

- 建立和管理滲透測試組態和測試執行

- 建立和管理設計檢閱
- 針對連線的儲存庫和 S3 來源建立、管理和執行程式碼檢閱
- 針對原始碼、範圍文件或兩者建立、管理和執行威脅模型
- 檢視滲透測試、程式碼檢閱、設計檢閱和威脅模型的問題清單

在 GitHub 中：

- 將提取請求程式碼檢閱問題清單檢視為提取請求評論
- 接收程式碼檢閱和滲透測試問題清單的自動修補提取請求（在 Agent Space 中啟用時）

> **Note**
>
> 提取請求程式碼檢閱調查結果會顯示在 GitHub 中。完整程式碼檢閱、滲透測試和設計檢閱調查結果會顯示在 Security Agent Web Application 中。

開始使用

註冊、為您的目標尋找正確的起點，並快速開始執行您的第一個評估。

註冊 AWS 帳戶

若要開始使用 AWS，您需要 AWS 帳戶。如需建立 AWS 帳戶的資訊，請參閱《AWS 帳戶管理參考指南》中的 帳戶入門。 AWS

選擇您的路徑

尋找您想要使用 AWS Security Agent 執行作業的正確起點。使用下表將您的目標與任務配對，然後直接前往該頁面。如果您是初次使用 AWS Security Agent，the section called “什麼是 AWS Security Agent？”請先閱讀 以取得快速概觀。

我想要...	誰	從這裡開始
了解 AWS Security Agent 在設定之前執行的動作	每個人	the section called “什麼是 AWS Security Agent？”
設定服務並建立代理程式空間	管理員	the section called “設定 AWS 安全代理程式”
測試我的即時或部署應用程式是否有可利用的漏洞	使用者	the section called “快速入門：執行滲透測試”
掃描我的原始程式碼是否有漏洞和政策違規	使用者	the section called “快速入門：執行程式碼檢閱”
示範如何攻擊我的應用程式 (STRIDE)	使用者	the section called “快速入門：執行威脅模型”
在撰寫程式碼之前取得設計的安全意見回饋	使用者	the section called “建立設計檢閱”
掃描程式碼而不離開我的 IDE (Kiro 或 Claude Code)	開發人員	the section called “從 IDE 執行程式碼安全掃描”

我想要...	誰	從這裡開始
合併之前僅掃描我變更的行	開發人員	the section called “使用 S3 執行差異碼掃描”
取得提取請求和合併請求的自動安全註解	管理員	the section called “啟用程式碼檢閱”
檢閱問題清單並決定要先修正的項目	使用者	滲透測試、程式碼檢閱、威脅模型、設計檢閱

> **Note**
>
> AWS Security Agent 使用三個角色，您可以透過您工作的界面來識別：管理員在 AWS 管理主控台（設定和組態）中工作、使用者在 Web 應用程式（執行評估和檢閱調查結果）中工作，而開發人員在 GitHub 或 IDE 中工作，例如 Kiro 或 Claude Code。單一人員可以擁有多個角色。如需完整定義，請參閱 the section called “AWS Security Agent 的運作方式”。

快速入門：執行滲透測試

此快速入門會逐步引導您使用 AWS Security Agent 執行您的第一個滲透測試（滲透測試）。滲透測試會針對已驗證的目標網域執行您部署的應用程式，並傳回安全性調查結果，每個調查結果都有嚴重性評分和支援證據。它涵蓋可公開存取的應用程式；若要測試私有 VPC 中託管的應用程式，請參閱 the section called “啟用滲透測試”。

> **Note**
>
> 您需要存取 AWS 管理主控台來設定 AWS Security Agent 並定義測試範圍，以及存取 Web 應用程式來建立和執行滲透測試。

先決條件

開始之前，請確定您已：

- 存取具有設定 AWS Security Agent 許可的 AWS 管理主控台。

- 託管您要測試之應用程式的即時目標網域。
- 新增該網域的 DNS 記錄，或相同 AWS 帳戶中 Route 53 託管區域的功能，讓您可以驗證擁有權。
- （選用）您應用程式的原始程式碼儲存庫 (GitHub、GitLab 或 Bitbucket)，為代理程式提供更多內容。

步驟 1：在 AWS 主控台中設定 AWS 安全代理程式

如果您尚未設定 AWS Security Agent，請完成初始設定：

1. 在 AWS 管理主控台中導覽至 AWS 安全代理程式。
2. 選取設定 AWS 安全代理程式。
3. 建立 代理程式空間。代理程式空間可供多個使用者使用，且應針對您想要測試的每個應用程式指定。輸入名稱和描述。名稱應識別您要滲透測試的應用程式。
4. 在使用者存取組態下選取僅限 IAM 存取。
   - 此快速入門不包含使用 IAM Identity Center 啟用單一登入 (SSO)，這可讓使用者直接從 AWS 主控台存取 Web 應用程式。
   - 若要讓沒有 AWS 管理主控台的使用者存取啟動滲透測試，請啟用 IAM Identity Center 整合。如需詳細資訊，請參閱the section called “授予使用者存取 AWS Security Agent Web 應用程式的權限”。
5. 選擇設定 AWS 安全代理程式。

> **Note**
>
> 當您選擇設定時，AWS Security Agent 會建立您的 Agent Space，並建立 Web 應用程式，讓使用者可以執行滲透測試、程式碼檢閱、威脅模型和設計檢閱。

步驟 2：啟用滲透測試並驗證您的網域

> **Note**
>
> 在 AWS 主控台中，您可以定義可測試的內容範圍。使用者接著會從 Web 應用程式在該範圍內執行特定滲透測試。

1. 從左側邊欄中，選取客服人員空間，然後選取您在步驟 1 中建立的客服人員空間。
2. 選取滲透測試索引標籤，然後選擇設定滲透測試以開啟精靈。
3. 設定網域 - 輸入您要測試的目標網域，然後選取驗證方法、DNS TXT Record 或 HTTP Route。網域必須是即時的，並託管您要測試的應用程式。選擇下一步。
4. 驗證網域 - 驗證目標網域資料表中每個網域的擁有權。AWS Security Agent 只會針對已驗證的網域執行測試。如需詳細步驟和要複製的確切值，請參閱 the section called “啟用用於滲透測試的應用程式網域”。
   - 相同 AWS 帳戶中的 Route 53 網域 - 選取網域，然後選擇一鍵式驗證。AWS Security Agent 會建立 DNS 記錄並完成驗證。
   - DNS TXT 記錄（其他 DNS 提供者）- 在目標網域資料表中，複製記錄名稱和記錄值，將其新增為 DNS 提供者的 TXT 記錄，然後選取網域，然後選擇驗證。DNS 變更可能需要一些時間才能傳播，因此驗證可能不會立即完成。
   - HTTP 路由 - `.well-known/aws/securityagent-domain-verification.json` 在 Web 伺服器上於 建立檔案，以 格式新增驗證字符`{"tokens": ["<token>"]}`，然後選取網域，然後選擇驗證。
5. （選用）設定其他功能 - 新增選用資源，例如 CloudWatch 日誌群組和登入資料。服務存取區段已預先設定：AWS Security Agent 會建立具有所需許可的服務角色，除非您選擇現有的 IAM 角色。

步驟 3：連接原始程式碼（選用）

連接原始碼供應商可提供有關應用程式的 AWS Security Agent 內容，並改善滲透測試涵蓋範圍。此為選擇性步驟。

1. 在滲透測試索引標籤上，選擇頁面頂端的在連接來源碼提供者上新增滲透測試橫幅。
2. 註冊並連接您的供應商，然後選取可供滲透測試使用的儲存庫。

如需完整的整合流程，請參閱將 AWS 安全代理程式連線至 GitHub 儲存庫或供應商的連線主題。

步驟 4：建立並執行滲透測試

> **Note**
>
> 您可以在 AWS Security Agent Web 應用程式中建立和執行滲透測試。我們建議針對緊密反映生產的測試環境執行測試。

1. 啟動 Web 應用程式：選取 Web 應用程式索引標籤，然後選取管理員存取。
2. 在左側邊欄中，選擇滲透測試，然後選擇建立滲透測試。
3. 滲透測試詳細資訊 - 設定測試範圍和日誌來源：
   a. 輸入 Pentest 名稱。
   b. 在目標 URLs下，選取或輸入要測試的已驗證網域（例如 `https://example.com`)。只能測試已驗證的網域，且會自動涵蓋子網域。
   c. （選用）在可存取URLs 下，新增代理程式登入和導覽時必須到達的網域，但不應攻擊，例如身分提供者、CDNs 或目標網域以外的 APIs。AWS Security Agent 可以到達這些網域，但不會對其進行測試；只會攻擊您的目標網域。
   d. 檢閱聯網組態規則，以確認哪些端點在測試期間可以和不可以接收流量。
   e. 在許可下，選取服務角色，並選擇性地選取 CloudWatch 日誌群組。選擇下一步。
4. （選用）VPC 資源 - 如果您的目標位於無法公開連線的私有網路上，請設定 VPC。請參閱 the section called “啟用滲透測試”。
5. （選用）身分驗證資源 - 提供登入資料，以便代理程式可以到達已驗證的非公有路徑。我們建議新增這些項目，以擴大登入後的涵蓋範圍和表面漏洞。
6. （選用）其他組態 - 新增應用程式內容，例如檔案、GitHub 儲存庫或 S3 來源，以改善問題清單品質。您也可以啟用自動程式碼修復，並在此處設定其他執行選項。
7. 選擇建立並執行以立即開始測試，或選擇建立 pentest 以儲存並稍後執行測試。

步驟 5：檢閱滲透測試問題清單

1. 滲透測試可能需要幾個小時才能完成。大部分會在 16 小時內完成，取決於應用程式的大小和複雜性。
2. 執行會從測試開始之前驗證設定的預檢階段開始：它會設定記錄基礎設施、檢查您的目標端點是否可存取，以及準備測試環境。如果預檢檢查失敗（例如，無法解析的目標網域），則執行會在測試開始之前停止。開啟預檢索引標籤以查看哪個檢查失敗、解決它，然後開始新的執行。
3. 當執行完成時，開啟它並檢閱執行概觀、日誌和調查結果索引標籤。
4. 在調查結果索引標籤上，選取調查結果以檢視其描述、嚴重性、風險類型和支援證據。

如需詳細資訊，請參閱the section called “建立滲透測試”和the section called “檢閱滲透測試的問題清單”。

快速入門：執行程式碼檢閱

此快速入門會逐步引導您使用 AWS Security Agent 執行您的第一個程式碼檢閱。AWS Security Agent 會掃描您的原始程式碼儲存庫是否有安全漏洞，以及是否符合組織的安全需求。

> **Note**
>
> 您需要存取 AWS 管理主控台來設定 AWS 安全代理程式，以及存取 Web 應用程式來建立和執行程式碼檢閱。

先決條件

開始之前，請確定您已：

- 存取具有設定 AWS Security Agent 許可的 AWS 管理主控台。
- 原始程式碼儲存庫 (GitHub、GitLab 或 Bitbucket) 或包含您要檢閱之程式碼的 Amazon S3 來源。

步驟 1：在 AWS 主控台中設定 AWS Security Agent

如果您尚未設定 AWS Security Agent，請完成初始設定：

1. 在 AWS 管理主控台中導覽至 AWS 安全代理程式。
2. 選取設定 AWS 安全代理程式。
3. 建立 代理程式空間。代理程式空間可供多個使用者使用，且應針對您要測試的每個應用程式專用。輸入第一個客服人員空間的名稱和描述。此名稱對 Web 應用程式中的使用者顯示。名稱應識別您要檢閱其程式碼的應用程式。
4. 在使用者存取組態下選取僅限 IAM 存取。
   - 此快速入門不包含使用 IAM Identity Center 啟用單一登入 (SSO)。這可讓使用者從 AWS 主控台直接存取 AWS Security Agent Web 應用程式。
   - 若要讓沒有 AWS 管理主控台的使用者存取執行程式碼檢閱，請啟用 IAM Identity Center 整合。如需詳細資訊，請參閱the section called “授予使用者存取 AWS Security Agent Web 應用程式的權限”。
5. 選擇設定 AWS 安全代理程式。

> **Note**
>
> 當您選擇設定時，AWS Security Agent 會建立您的 Agent Space，並建立 Web 應用程式，讓使用者可以執行滲透測試、程式碼檢閱、威脅模型和設計檢閱。

步驟 2：啟用和設定程式碼檢閱

> **Note**
>
> 如果您已將 GitHub 儲存庫或 S3 儲存貯體連接至您的客服人員空間（例如，透過滲透測試設定），則程式碼檢閱已啟用。您可以略過此步驟並直接前往 Web 應用程式。

開啟程式碼檢閱設定精靈

1. 從左側邊欄中，選取客服人員空間，然後選取您的客服人員空間。
2. 從標頭或程式碼檢閱索引標籤中選取啟用程式碼檢閱。

連接整合、儲存庫和儲存貯體

1. （如果您還沒有 GitHub 整合）建立 GitHub 註冊。如果您已經有，請跳到下一個步驟。
   a. 在連線整合區段中，選擇新增，然後選擇建立新註冊。
   b. 選取 GitHub，然後選擇下一步。
   c. 選擇安裝並授權，然後在 GitHub 中完成安裝：
      i. 選取擁有您要檢閱之儲存庫的 GitHub 使用者或組織。
      ii. 選取所有儲存庫或僅選取儲存庫。
      iii. 選擇安裝和授權並完成 GitHub 身分驗證。
   d. 返回 AWS 管理主控台，輸入註冊名稱並確認帳戶類型與您安裝 GitHub 應用程式的位置相符。
   e. 選擇連線以儲存註冊。

   如需完整的 GitHub 整合流程，請參閱將 AWS 安全代理程式連線至 GitHub 儲存庫。
2. 連接 GitHub 儲存庫。在連線整合區段中，選擇新增，然後選取您的 GitHub 註冊。開啟兩個步驟的 Connect GitHub 精靈：
   a. 在 Connect GitHub 儲存庫上，選取要包含的儲存庫，然後選擇下一步。

b. 在管理功能上，為每個儲存庫切換下列項目：

- 程式碼檢閱評論 – 讓 AWS Security Agent 將安全性調查結果發佈為對儲存庫中提取請求的評論。
- 自動修復 – 讓 AWS Security Agent Web 應用程式請求的使用者提取請求來修正問題清單。

c. 選擇儲存以返回設定精靈。

3. （選用）連接 S3 來源。在 S3 儲存貯體區段中，選擇新增 S3 資源，並為包含原始碼的儲存貯體輸入 S3 URI，或選擇瀏覽以選擇一個。
4. 選取您的程式碼檢閱設定。預設的安全需求和漏洞調查結果，會分析程式碼是否符合您已啟用的安全需求和常見漏洞。
5. 選擇下一步。

選用組態

1. 設定選用的 CloudWatch 日誌群組和服務存取。預設服務角色已預先設定必要的許可。
2. 選擇儲存。

步驟 3：建立和執行程式碼檢閱

> **Note**
>
> 您只能在 AWS Security Agent Web 應用程式中建立和執行程式碼檢閱。

1. 選取 Web 應用程式索引標籤，然後管理員存取以啟動 AWS Security Agent Web 應用程式。
2. 在左側邊欄中，選擇程式碼檢閱。
3. 選擇建立程式碼檢閱。
4. 設定程式碼檢閱：

a. 輸入識別此檢閱範圍的標題（例如「billing-service-security-review」）。

b. 在來源下，選取您在步驟 2 中連線至客服人員空間的 GitHub 儲存庫，或輸入您要掃描的 S3 來源。

c. 從設定的角色中選取服務角色。

d. （選用）選取啟用自動程式碼修復，讓 AWS Security Agent 自動提交提取請求，並修正所有問題清單。

5. 選擇建立程式碼檢閱。
6. 在程式碼檢閱詳細資訊頁面上，選擇開始檢閱。

步驟 4：檢閱程式碼檢閱問題清單

1. 程式碼檢閱通常需要 30-60 分鐘，視您的程式碼庫大小而定。
2. 執行從預檢階段開始，在掃描開始之前驗證設定：它確認 AWS Security Agent 可以從儲存庫或 S3 來源提取您的原始程式碼，並設定掃描環境。如果預檢檢查失敗（例如，它無法存取您的來源），則執行會在靜態分析之前停止。開啟預檢索引標籤以查看哪些檢查失敗、解決它，然後開始新的執行。
3. 完成後，導覽至已完成的執行，然後選取問題清單索引標籤。
4. 在 list-detail 檢視中檢閱問題清單：
   a. 從左側面板選取問題清單以檢視其詳細資訊。
   b. 檢閱描述、程式碼位置、證據和建議的修正區段。
   c. 使用修復程式碼來產生具有修正的提取請求，或者如果您啟用該選項，則檢閱自動修復 PRs。

如需詳細資訊，請參閱the section called “建立程式碼檢閱”和the section called “檢閱程式碼檢閱的問題清單”。

快速入門：執行威脅模型

此快速入門會逐步引導您使用 AWS Security Agent 執行第一個威脅模型。威脅模型會分析應用程式的架構，並產生系統概觀 (AWS Security Agent 如何了解您的系統）和一組威脅（如何受到攻擊，每個威脅都有嚴重性等級、STRIDE 分類和建議）。您可以在設計文件 (範圍文件) 上執行威脅模型，以定義焦點、原始程式碼（來源)，以提供現有系統或兩者的內容。

> **Note**
>
> 您需要存取 AWS 管理主控台來設定 AWS Security Agent，以及存取 Web 應用程式來建立和執行威脅模型。

步驟 1：在 AWS 主控台中設定 AWS Security Agent

如果您尚未設定 AWS Security Agent，請完成初始設定：

1. 在 AWS 管理主控台中導覽至 AWS 安全代理程式。
2. 選取設定 AWS 安全代理程式。
3. 建立 代理程式空間。代理程式空間可供多個使用者使用，且應針對您想要保護的每個應用程式專用。輸入第一個客服人員空間的名稱和描述。名稱應識別您要威脅模型的應用程式。
4. 在使用者存取組態下選取僅限 IAM 存取。
   - 此快速入門不包含使用 IAM Identity Center 啟用單一登入 (SSO)。這可讓使用者從 AWS 主控台直接存取 AWS Security Agent Web 應用程式。
   - 如果您想要讓沒有 AWS 管理主控台存取權的使用者執行任務，例如啟動威脅模型，您應該啟用 IAM Identity Center 整合。
5. 選擇設定 AWS 安全代理程式。

> **Note**
>
> 當您選擇設定時，AWS Security Agent 會建立您的 Agent Space，並建立 Web 應用程式，讓使用者可以執行滲透測試、程式碼檢閱、威脅模型和設計檢閱。

步驟 2：連接原始程式碼

> **Note**
>
> 如果您已將儲存庫或 S3 儲存貯體連接至您的 Agent Space （例如，透過程式碼檢閱或滲透測試設定），您可以略過此步驟並直接前往 Web 應用程式。您可以隨時在上傳的範圍文件上執行威脅模型，而無需連接任何原始程式碼。

連接儲存庫或 S3 儲存貯體，其中包含您希望代理程式用作了解現有系統內容的原始程式碼：

1. 從左側邊欄中，選取客服人員空間，然後選取您的客服人員空間。
2. 導覽至整合區段以連接您的原始碼提供者。
3. 或者，連接包含原始程式碼的 S3 儲存貯體。

如需完整的整合流程，請參閱將 AWS 安全代理程式連線至 GitHub 儲存庫。

步驟 3：建立並執行威脅模型

> **Note**
>
> 您可以在 AWS Security Agent Web 應用程式中建立和執行威脅模型。

1. 從 AWS 管理主控台中的 Web 應用程式索引標籤啟動 Web 應用程式。或者，如果您已設定 IAM Identity Center，請直接登入。
2. 在左側邊欄中，選擇威脅模型。
3. 選擇建立威脅模型。
4. 設定威脅模型：
   a. 輸入識別此威脅模型範圍的標題（例如「billing-service」或「checkout-api」）。
   b. 提供至少一個輸入：
      - 在範圍文件下，上傳設計文件 (DOC、DOCX、JPEG、MD、PDF、PNG 或 TXT)、輸入 S3 URIs，或選取 Confluence 頁面。
      - 在來源下，從連線的整合中選取儲存庫，或輸入包含來源碼的 S3 URIs。

      > **Tip**
      >
      > 提供來源和範圍文件，將威脅模型範圍限定為特定設計（例如，功能設計文件），同時將您的原始程式碼做為內容提供給代理程式，以了解您現有的系統。

   c. 從設定的角色中選取服務角色。
   d. （選用）選取 CloudWatch 日誌的日誌群組。
5. 選擇建立威脅模型。
6. 在威脅模型詳細資訊頁面上，選擇開始執行。

步驟 4：檢閱威脅

1. 執行會進行其分析任務。您可以在預檢索引標籤上監控進度，並在日誌索引標籤上檢視任務詳細資訊。
2. 完成後，執行狀態會變更為已完成。
3. 選取概觀索引標籤，以檢閱系統概觀和嚴重性分佈。

4. 選取威脅索引標籤以檢閱已識別的威脅：
   a. 從清單中選擇威脅，以在側邊面板中檢視其詳細資訊。
   b. 檢閱陳述式、嚴重性、STRIDE 類別、建議、證據和其他欄位。
   c. 當您處理或分類每個威脅時，將威脅狀態更新為已解決或已解除。

如需詳細資訊，請參閱the section called “建立威脅模型”和the section called “檢閱來自威脅模型的威脅”。

適用於管理員（主控台）

身為管理員，您可以在 AWS 管理主控台中設定 AWS Security Agent，並設定使用者透過 AWS Security Agent Web 應用程式存取的 Agent Spaces。每個客服人員空間代表具有特定許可和資源的不同環境。

建議您為要測試的每個應用程式建立唯一的 Agent Space。例如，如果您有兩個內部專案，即帳單應用程式和任務追蹤應用程式，您應該建立兩個單獨的客服人員空間。

範例組態

考慮管理員設定 代理程式空間，以評估內部帳單應用程式的安全性。管理員會：

- 驗證網域（例如 `beta.billing.example.com`)
- 連線至 GitHub 並啟用程式碼檢閱
- 透過指派適當的 VPC、子網路和安全群組進行滲透測試來設定網路存取

當使用者啟動滲透測試或設計審查時，他們可以從這些預先設定的資源中選擇，在您定義的護欄內工作，同時保持其特定評估需求的靈活性。

設定和建立客服人員空間

為您的組織設定 AWS 安全代理程式，並建立涵蓋每個應用程式資源和評估範圍的代理程式空間。

設定 AWS 安全代理程式

為您的組織設定 AWS 安全代理程式，方法是建立您的第一個代理程式空間，並建立使用者存取 Web 應用程式的方式。

此初始設定可讓管理員在 AWS 主控台中設定安全功能，並提供使用者透過 Security Agent Web Application 進行設計檢閱和滲透測試的存取權。在此一次性設定之後，您可以使用簡化的程序建立其他 Agent Spaces。

概觀

了解客服人員空間

Agent Space 是保護特定應用程式或專案安全的專用工作區。它包含該應用程式的所有安全性審查、選擇性連接的 GitHub 儲存庫、滲透測試組態、結果和調查結果。

Agent Spaces 可讓您將每個應用程式的安全評估分開，讓團隊專注於其特定應用程式，以協助您組織安全工作。我們建議您為每個應用程式或專案建立一個代理程式空間，以維持明確的界限。

安全需求是在組織層級定義，並適用於所有 Agent Spaces，而每個 Agent Space 會維護自己的設計文件、程式碼儲存庫、滲透測試組態、滲透測試結果和安全性調查結果。

客服人員空間中包含的內容

每個客服人員空間都包含：

- 選擇性地連接與應用程式或專案相關聯的 GitHub 儲存庫
- 應用程式的先前設計檢閱
- 專屬於應用程式的滲透測試組態和界限
- 滲透測試結果和安全性調查結果

您將在設定期間設定的內容

在此第一次設定期間，您將做出適用於所有客服人員空間的組織決策：

- 存取方法 - 選擇使用者如何透過 AWS 主控台存取 Security Agent Web Application (IAM Identity Center SSO 或僅限 IAM 存取）
- 許可 - 設定 Web 應用程式用來存取 AWS 服務的 IAM 角色
- 第一個客服人員空間 - 使用名稱和描述建立您的初始客服人員空間

> **Note**
>
> 完成此設定後，建立其他 Agent Spaces 會更簡單，只需提供名稱和描述即可。如需建立後續客服人員空間的詳細資訊the section called “建立 代理程式空間”，請參閱 。

先決條件

開始前，請確保您具備以下條件：

- 建立和管理 AWS Security Agent 資源的許可
- 了解組織的身分管理需求
- （選用）具有建立 IAM 角色和設定 IAM Identity Center（如果使用 SSO 存取）許可的 AWS 帳戶

- （選用）如果您想要使用自訂許可組態，現有的 IAM 角色

步驟 1：建立您的第一個客服人員空間

定義將在 Web 應用程式中顯示給使用者的第一個 Agent Space 的基本屬性。

1. 在 AWS Security Agent 主控台頁面上，選擇設定安全代理程式。
2. 在客服人員空間名稱欄位中，輸入客服人員空間的名稱。

> **Note**
>
> 客服人員空間名稱會向 Web 應用程式中的使用者顯示，並協助識別此空間代表的應用程式或專案。

3. （選用）在描述欄位中，提供描述以協助區分客服人員空間用途。

> **Tip**
>
> 描述有助於區分 Agent Space 的目的。我們建議描述此 Agent Space 將保護的特定應用程式或專案，例如「客戶入口網站 Web 應用程式」或「付款處理微服務」或「內部分析平台」。

步驟 2：選擇您的存取方法

選取使用者存取 Security Agent Web Application 的方式。您可以選擇透過 IAM Identity Center 啟用 SSO 存取，或透過 AWS 主控台提供存取。

> **Note**
>
> 如果您選擇僅 IAM 存取，且稍後想要使用 IAM Identity Center，您將需要刪除 AWS Security Agent 設定，然後再次完成設定程序。

1. 在存取方法區段中，選取下列其中一個選項：
   - IAM Identity Center (SSO) - 透過 IAM Identity Center 為您的團隊啟用 SSO 存取。對於需要集中式使用者管理，並希望使用者無需透過 AWS 主控台直接存取 Web 應用程式的團隊，建議使用此選項。

- 僅限 IAM 存取 - 透過 AWS 主控台中的管理員存取連結提供存取權。此選項更容易設定且適合偏好以主控台為基礎的存取或不需要 SSO 功能的團隊。

2. 如果您選擇了 IAM Identity Center (SSO)，請繼續下面的步驟 2a。
3. 如果您選取僅限 IAM 存取，請跳至步驟 3。

步驟 2a：設定 IAM Identity Center （僅限 SSO 存取）

只有在您選取 IAM Identity Center (SSO) 做為存取方法時，才能完成這些步驟。

1. 在連線至 IAM Identity Center 區段中，檢閱 AWS 區域。

> **Important**
>
> 您的應用程式必須在您啟用 IAM Identity Center 的相同區域中設定。顯示的 區域是您的客服人員空間將建立的位置。IAM Identity Center 必須在您建立客服人員空間的相同區域中啟用。

2. 在 IAM Identity Center 區段中，選擇下列其中一項：
   - 選擇建立帳戶執行個體以建立新的 IAM Identity Center 帳戶執行個體
   - 如果組織執行個體已存在於相同區域，AWS Security Agent 會自動連線到該執行個體

> **Note**
>
> AWS Security Agent 連線至 IAM Identity Center 後，您可以透過將 IAM Identity Center 中的使用者指派給應用程式來管理存取權。

3. 如果您要連接 Active Directory 或外部身分提供者，請檢閱資訊提醒。

> **Important**
>
> 如果您已在 Active Directory 或其他身分來源中管理使用者，並計劃將該身分來源連線至 IAM Identity Center，請先取消設定並前往 IAM Identity Center 主控台。選擇您要連線的身分來源，再設定 AWS Security Agent。稍後變更身分來源可能會移除現有的使用者指派。

步驟 3：設定許可 （選用）

設定 Security Agent Web Application 用來存取 AWS 服務、APIs 和帳戶的 IAM 角色。

1. 尋找許可組態 - 選用區段。
2. 如果區段已摺疊，請選擇許可組態區段來展開。
3. 選取以下其中一個選項：
   - 建立預設角色 - AWS Security Agent 會自動建立具有 Web 應用程式必要許可的新 IAM 角色
   - 使用另一個角色 - 從可用的選項中選取現有的 IAM 角色
4. 檢閱角色描述：

> **Note**
>
> 系統會為您的 Web 應用程式建立預設 IAM 角色，以存取其他 AWS 服務、APIs和帳戶。角色將獲授予安全評估操作所需的許可。

步驟 4：完成設定

設定所有必要的設定後，請完成設定以建立您的第一個客服人員空間，並建立安全客服人員 Web 應用程式。

1. 檢閱所有組態區段以確保準確性。
2. 選擇 Set up (設定)。
3. AWS Security Agent 會建立您的 Agent Space、建立 Web 應用程式，以及設定必要的 AWS 資源。

> **Note**
>
> 初始設定之後，您可以選擇設定功能，包括滲透測試界限、安全需求和 GitHub 整合。

後續步驟

設定 AWS Security Agent 之後：

- 定義組織的安全需求
- 設定滲透測試功能，包括代理程式空間的網域驗證和 VPC 存取
- 連接 GitHub 儲存庫以進行程式碼檢閱和滲透測試內容
- （如果使用 IAM Identity Center) 在 AWS Security Agent 主控台中將使用者指派給 Agent Space
- （如果使用僅限 IAM 存取）透過 AWS 主控台中的管理員存取連結啟動 Web 應用程式

- 為其他應用程式建立其他代理程式空間（請參閱 the section called “建立 代理程式空間”）

建立 代理程式空間

建立 Agent Spaces 以保護組織中的應用程式或專案。每個 Agent Space 都會提供一個工作區，用於該應用程式或專案特定的安全評估、調查結果和組態，並且可供多個使用者存取。

此程序假設您已完成初始 AWS Security Agent 設定。如果您尚未設定 AWS Security Agent，請參閱 the section called “設定 AWS 安全代理程式”。

概觀

了解客服人員空間

客服人員空間是保護特定應用程式或專案安全的專用工作區。它包含該應用程式的所有安全性審查、選擇性連線的程式碼儲存庫、滲透測試組態、結果和調查結果。

Agent Spaces 可讓您將每個應用程式的安全評估分開，讓團隊專注於其特定應用程式，以協助您組織安全工作。我們建議您為每個應用程式或專案建立一個代理程式空間，以維持明確的界限。

如需 Agent Spaces 及其如何符合組織安全結構的完整說明，請參閱 the section called “了解資源階層和生命週期”。

客服人員空間中包含的內容

每個客服人員空間都包含：

- 選擇性連線與應用程式或專案相關聯的程式碼儲存庫
- 應用程式或專案的先前設計審查
- 應用程式的專屬滲透測試組態和界限
- 滲透測試結果和安全性調查結果

建立 代理程式空間

為您要保護的應用程式或專案建立新的 Agent Space。

1. 在 AWS Security Agent 主控台中，導覽至 Agent Spaces 頁面。
2. 選擇建立客服人員空間。
3. 在客服人員空間名稱欄位中，輸入客服人員空間的名稱。

> **Note**
>
> 客服人員空間名稱會顯示在 Web 應用程式中的使用者，並協助識別此空間代表的應用程式或專案。

4. （選用）在描述欄位中，提供描述以協助區分客服人員空間用途。

> **Tip**
>
> 描述有助於區分客服人員空間的目的。我們建議您描述此 Agent Space 將保護的特定應用程式或專案，例如「客戶入口網站 Web 應用程式」或「付款處理微服務」或「內部分析平台」。

5. 選擇建立。

> **Note**
>
> AWS 安全代理程式會建立您的代理程式空間。您現在可以設定功能並連接此應用程式特定的資源。

後續步驟

建立客服人員空間之後：

- 連接原始程式碼儲存庫 (GitHub、GitLab、Bitbucket 或 GitHub Enterprise Server) 以進行程式碼檢閱和滲透測試內容
- Connect Confluence 用於設計審查和滲透測試的文件內容
- 啟用已連線儲存庫的程式碼檢閱功能（請參閱 the section called “啟用 GitHub 儲存庫的提取請求程式碼檢閱”）
- 設定滲透測試功能，包括網域驗證
- （如果使用 IAM Identity Center) 在客服人員空間頁面的 Web 應用程式區段下，將使用者指派給此客服人員空間。（請參閱 the section called “授予使用者存取 AWS Security Agent Web 應用程式的權限”）
- （如果使用僅限 IAM 存取）具有主控台存取的使用者可以透過此代理程式空間的管理員存取連結啟動 Web 應用程式

連線整合

連接和管理原始程式碼提供者 (GitHub、GitLab、Bitbucket 和 GitHub Enterprise Server) 和文件提供者 (Confluence)，以及您的 Agent Spaces 用於程式碼檢閱、滲透測試和威脅建模的私有網路連線。

整合如何與 Agent Spaces 搭配使用

AWS Security Agent 會連線至第三方原始碼和文件提供者，以便代理程式可以在安全評估期間分析您的程式碼和文件。在您連接特定供應商之前，了解整合與 Agent Spaces 和 功能的關係會很有幫助。

註冊一次，在任何地方重複使用

將供應商連線至 AWS Security Agent 涉及兩個不同層級的不同步驟：

1. 註冊整合 （帳戶層級）。您可以從 AWS 安全代理程式管理主控台的整合頁面註冊供應商一次。註冊代表與提供者帳戶的授權連線，例如 GitHub 組織、GitLab 群組、Bitbucket 工作區或 Confluence 網站。註冊是帳戶層級的資源。
2. 將資源連接至客服人員空間 （客服人員空間層級）。從 代理程式空間內，您可以從已註冊的整合 - 來源碼供應商的儲存庫或 Confluence 的頁面 - 連接資源，並設定代理程式使用它們的方式。此步驟專屬於每個客服人員空間。

您帳戶中的每個客服人員空間都會重複使用單一註冊。如果您在設定一個客服人員空間時註冊了供應商，則當您將資源連接到另一個客服人員空間時，即可使用相同的註冊 — 您不會再次註冊相同的供應商。

一個連線，跨功能共用

當您將資源連線至 代理程式空間時，該資源會跨代理程式的功能共用。您不會為每個功能分別連接儲存庫。

- 讀取存取權是透過連接 資源授予。連接儲存庫可讓 AWS Security Agent 讀取儲存庫以進行完整程式碼掃描 （程式碼檢閱）、滲透測試內容、威脅建模和設計檢閱。連接 Confluence 頁面可讓代理程式在設計檢閱、威脅建模和滲透測試中讀取文件內容。
- 每個儲存庫的寫入動作都是選擇加入。當您將原始碼儲存庫連線至 代理程式空間時，您可以另外為每個儲存庫啟用寫入動作：
  - 程式碼檢閱註解 - AWS Security Agent 會將檢閱調查結果張貼為提取請求 （或在 GitLab 中合併請求） 的註解。

- 程式碼修復 - AWS Security Agent 會開啟提取請求 （或合併請求），並修正探索到的漏洞。

  這些寫入動作不適用於公有儲存庫。唯讀分析仍然適用。

> **Note**
>
> Confluence 是文件提供者，而不是原始程式碼提供者。AWS Security Agent 會讀取連線 Confluence 頁面，以提供設計檢閱、威脅建模和滲透測試的內容。選取頁面會授予讀取存取權；沒有每一頁的功能切換。

支援的提供者

AWS Security Agent 支援下列供應商：

- 原始碼 - GitHub （雲端託管 GitHub 和雲端託管 GitHub Enterprise)、GitHub Enterprise Server （自我託管）、GitLab （雲端託管）、GitLab 自我管理 （自我託管） 和 Bitbucket Cloud。
- 文件 - Confluence Cloud。

對於無法透過公有網際網路連線的自我託管提供者，您可以透過私有連線路由代理程式的流量。如需詳細資訊，請參閱the section called “連線至私有託管來源控制”。

後續步驟

- 從整合頁面註冊供應商。請參閱供應商的連線主題，例如 the section called “將 AWS 安全代理程式連線至 GitHub 儲存庫”。
- 將資源連接至 代理程式空間並設定 功能。請參閱the section called “啟用程式碼檢閱”和the section called “啟用滲透測試”。

將 AWS 安全代理程式連線至 GitHub 儲存庫

將您的 AWS 安全代理程式連線至 GitHub 儲存庫，以啟用程式碼檢閱、威脅建模、滲透測試和自動化修復功能。AWS Security Agent 同時支援雲端託管的 GitHub 和雲端託管的 GitHub Enterprise。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 之間重複使用註冊並跨 功能共用。

GitHub 整合提供多種用途：

> **Note**
>
> 此頁面涵蓋雲端託管的 GitHub (github.com：//) 和雲端託管的 GitHub Enterprise。如需自我託管的 GitHub Enterprise Server，請參閱 the section called “將 AWS 安全代理程式連線至 GitHub Enterprise Server”。

- 程式碼檢閱 - 根據組織安全需求自動分析每個提取請求中的程式碼變更，並執行隨需完整儲存庫掃描
- 威脅建模 - 透過分析原始程式碼、資料流程和架構來提供應用程式理解
- 滲透測試內容 - 透過分析原始程式碼，為滲透測試提供應用程式理解
- 自動化修復 - 提交提取請求，並修正安全性評估期間發現的漏洞

將 GitHub 連線至 AWS Security Agent 需要為您的 GitHub 組織或使用者帳戶授權 AWS Security Agent GitHub 應用程式，然後在 AWS 主控台中註冊連線。

GitHub 整合的運作方式

提取請求分析會在 GitHub 內進行。在您授權 GitHub 應用程式、連接儲存庫，並在 AWS 管理主控台中啟用程式碼檢閱註解後，AWS Security Agent 會自動掃描每個新提取請求中的變更（僅對變更程式碼進行差異掃描），並將調查結果發佈為提取請求註解。

您可以在 AWS Security Agent Web 應用程式中，而不是在 GitHub 中建立並執行完整的程式碼檢閱，以掃描儲存庫的整個程式碼庫。

滲透測試和威脅建模會在 AWS Security Agent Web 應用程式內啟動。使用者選取連線的儲存庫以提供應用程式內容，並指定滲透測試的目標網域。如果您啟用自動修復，使用者可以向連線的儲存庫開啟提取請求，請求 AWS Security Agent 修正問題清單。

先決條件

開始前，請確保您具備以下條件：

- GitHub 組織管理員存取權或 GitHub 使用者帳戶擁有者存取權
- 在 AWS 管理主控台中設定客服人員空間整合的許可
- 了解您要連線哪些儲存庫以進行程式碼檢閱、威脅建模和滲透測試

Important

GitHub 應用程式只能安裝到 GitHub 帳戶或 GitHub 組織一次。如果您需要將相同的 GitHub 組織連線至 AWS Security Agent，您必須使用第一次註冊整合的相同 AWS 帳戶。

Note

如果您的 GitHub 企業組織已啟用 IP 允許清單，則必須接受 GitHub 應用程式上允許的 IP 地址。您也可以選擇自動新增 IP 位址至允許清單。如需詳細資訊，請參閱 GitHub 文件中的允許 GitHub 應用程式存取和啟用允許的 IP 地址。 GitHub
下列 IP 地址用於存取您的 GitHub 資源：

- 美國東部 (維吉尼亞北部) (us-east-1)
  - 34.228.181.128
  - 44.219.176.187
  - 54.226.244.221
- 美國西部 (奧勒岡) (us-west-2)
  - 34.212.16.133
  - 52.89.67.212
  - 54.187.135.61
- 亞太地區 (孟買) (ap-south-1)
  - 13.126.209.199
  - 13.234.6.24
  - 35.154.102.216
- 亞太地區 (新加坡) (ap-southeast-1)
  - 18.139.13.125
  - 47.130.240.215
  - 54.179.238.173
- 亞太地區 (雪梨) (ap-southeast-2)
  - 13.237.95.197
  - 13.238.84.102
  - 52.64.174.242

- 亞太區域 (東京) (ap-northeast-1)
  - 13.192.12.233
  - 35.74.181.230
  - 57.183.50.158
- 歐洲 (法蘭克福) (eu-central-1)
  - 18.158.110.140
  - 52.57.96.160
  - 52.59.55.56
- 歐洲 (愛爾蘭) (eu-west-1)
  - 34.251.85.24
  - 52.30.157.157
  - 52.51.192.222
- 南美洲 (聖保羅) (sa-east-1)
  - 54.94.247.213
  - 54.207.222.14
  - 54.232.201.242

授權並註冊 AWS Security Agent GitHub 應用程式

授權 AWS Security Agent GitHub 應用程式存取您的 GitHub 組織或使用者帳戶，然後在 AWS 主控台中註冊連線。

> **Important**
>
> 完成此程序中的所有步驟，無需關閉瀏覽器或瀏覽。如果註冊程序中斷，您可能需要解除安裝 GitHub 應用程式並重新開始。

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 GitHub。
4. 選擇下一步。

5. 選擇安裝並授權。

   系統會將您重新導向至 GitHub 以完成授權。確保您使用具有要連線之組織或使用者帳戶管理員存取權的帳戶登入 GitHub。

6. 在 GitHub 中，選取您要安裝 AWS Security Agent GitHub 應用程式的帳戶或組織。
7. 選取 AWS Security Agent 可以存取的儲存庫：
   - 所有儲存庫 - 授予組織或使用者帳戶中所有目前和未來儲存庫的存取權
   - 僅選取儲存庫 - 從下拉式清單中選擇特定儲存庫。您可以一次選取一個儲存庫。

   > **Note**
   >
   > 您可以隨時透過存取 GitHub 組織或使用者帳戶設定中的 GitHub 應用程式設定來修改儲存庫存取。

8. 選擇安裝並授權。
9. 系統會將您重新導向回 AWS 管理主控台以完成註冊。
10. 在註冊詳細資訊區段中，設定下列欄位：
    a. 註冊名稱 - 輸入此 GitHub 連線的描述性名稱。使用可識別 GitHub 組織或使用者帳戶的名稱，例如「Acme-Corp-Org」或「Production-Repos」。
    b. 帳戶類型 - 從下拉式清單中選取下列其中一項：
       - Organization - 如果您連接 GitHub 組織帳戶
       - 使用者 - 如果您連接個人 GitHub 使用者帳戶
    c. 組織名稱（只有在您選取組織時才會顯示）- 輸入 GitHub 中顯示的 GitHub 組織的確切名稱。
11. 選擇連線。
12. 您會看到確認訊息並返回整合頁面，其中會顯示新的 GitHub 連線及其註冊名稱。若要連接其他 GitHub 組織或使用者帳戶，請再次選擇新增整合來重複此程序。

故障診斷 GitHub 整合

如果您在 GitHub 整合程序期間遇到問題，請使用下列指引來解決常見問題。

無法完成註冊

如果您無法完成註冊程序（例如，您的瀏覽器已關閉、導覽離開註冊頁面，或遇到工作階段中斷），AWS Security Agent GitHub 應用程式可能安裝在 GitHub 組織中，但未在 AWS 主控台中註冊。

徵狀：

- 當您嘗試再次授權 GitHub 應用程式時，GitHub 會顯示「設定」而非「安裝」
- 您無法在 AWS 主控台中完成註冊
- 整合不會出現在您的整合清單中

解決方法：

1. 從 GitHub 組織或使用者帳戶解除安裝 AWS Security Agent GitHub 應用程式。
2. 返回 AWS Security Agent 主控台，並從頭開始再次啟動整合程序。

多個 AWS 帳戶嘗試整合相同的 GitHub 組織

GitHub 應用程式只能安裝到 GitHub 帳戶或 GitHub 組織一次。如果您需要使用已與不同 AWS Security Agent 帳戶整合之 GitHub 組織的儲存庫，您必須使用第一次註冊整合的 AWS 帳戶。

解決方法：

- 識別哪些 AWS Security Agent 帳戶已註冊 GitHub 整合
- 使用該 AWS 帳戶建立 Agent Spaces 並連接儲存庫
- 如果您需要將整合移至不同的 AWS 帳戶，請先從原始 AWS 帳戶解除安裝 GitHub 應用程式，然後將其與新帳戶整合

後續步驟

將 GitHub 連線至 AWS Security Agent 之後：

- 導覽至您要使用這些儲存庫的客服人員空間
- 選擇啟用程式碼檢閱或設定滲透測試，將特定儲存庫連線至您的客服人員空間，並設定其用量
- 啟用自動修復，以允許 AWS Security Agent 提交具有漏洞修正的提取請求
- 在 GitHub 組織設定中檢閱 GitHub 應用程式許可和儲存庫存取權

移除 GitHub 整合

當您不再需要 AWS Security Agent 從特定 GitHub 組織或使用者帳戶存取儲存庫時，請移除 GitHub 整合。您必須先從 GitHub 解除安裝 AWS Security Agent GitHub 應用程式，才能移除 AWS 主控台中的整合。

移除的先決條件

在移除 GitHub 整合之前，請確定您有：

- 檢查哪些 Agent Spaces 具有從此整合連線的儲存庫
- 了解影響：移除此整合將中斷儲存庫連線，以便使用此整合中的儲存庫進行程式碼檢閱、滲透測試內容和所有客服人員空間的滲透測試修復
- GitHub 組織管理員存取權或使用者帳戶擁有者存取權，以解除安裝 GitHub 應用程式

步驟 1：從 GitHub 解除安裝 AWS 安全代理程式 GitHub 應用程式

首先，從您的 GitHub 組織或使用者帳戶解除安裝 AWS Security Agent GitHub 應用程式。

對於 GitHub Organizations：

1. 前往 https：//github.com 並開啟您的組織頁面。
2. 在左側邊欄中，選擇設定。
3. 在程式碼、規劃和自動化下，選擇已安裝的 GitHub 應用程式。
4. 在清單中找到 AWS Security Agent 應用程式。
5. 選擇 AWS Security Agent 應用程式旁的設定。
6. 捲動至組態頁面底部，然後選擇解除安裝。
7. 出現提示時確認解除安裝。

對於 GitHub 使用者帳戶：

1. 前往 https：//github.com。
2. 選擇您的設定檔圖片。
3. 選擇設定。
4. 在左側邊欄中，選取應用程式。
5. 開啟已安裝的 GitHub 應用程式索引標籤。

6. 在清單中找到 AWS Security Agent 應用程式。
7. 選擇 AWS Security Agent 應用程式旁的設定。
8. 捲動至組態頁面底部，然後選擇解除安裝。
9. 出現提示時確認解除安裝。

步驟 2：從 AWS Security Agent 移除整合

解除安裝 GitHub 應用程式之後，請從 AWS 主控台移除整合註冊。

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 在整合清單中尋找您要移除的 GitHub 整合。
3. 按一下以選取整合。
4. 選擇移除。
5. 檢閱確認對話方塊，該對話方塊會警告您影響：

> **Warning**
>
> 移除此整合將影響從此 GitHub 組織或使用者帳戶連線儲存庫的所有 Agent Spaces。這將中斷：
>
> - 連線儲存庫的程式碼檢閱功能
> - 來自連線儲存庫的滲透測試內容
> - 連線儲存庫的滲透測試修復功能
>
> 在繼續之前，請確定您已從 GitHub 解除安裝 AWS Security Agent GitHub 應用程式。

6. 如果您尚未從 GitHub 解除安裝 GitHub 應用程式，您將會收到警告。返回步驟 1 先完成解除安裝。
7. 如果您已解除安裝 GitHub 應用程式並了解影響，請選擇確認移除。
8. 整合會從整合清單中移除。具有來自此整合之儲存庫的任何 Agent Spaces 都無法再存取這些儲存庫，以進行程式碼檢閱、滲透測試內容或自動化修復。

將 AWS 安全代理程式連線至 GitLab 儲存庫

將您的 AWS Security Agent 連線至 GitLab Cloud 儲存庫，以啟用程式碼檢閱、威脅建模、滲透測試和自動化修復功能。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 之間重複使用註冊並跨 功能共用。

GitLab 整合提供多種用途：

- 程式碼檢閱 - 根據您的組織安全需求自動分析每個合併請求中的程式碼變更，並執行隨需完整儲存庫掃描
- 威脅建模 - 透過分析原始碼、資料流程和架構來提供應用程式理解
- 滲透測試內容 - 透過分析原始程式碼，為滲透測試提供應用程式理解
- 自動化修復 - 提交合併請求，並修正安全性評估期間發現的漏洞

將 GitLab 連線至 AWS Security Agent 需要提供具有適當許可的 GitLab 存取權杖，然後在 AWS 主控台中註冊連線。

GitLab 整合的運作方式

合併請求分析會在 GitLab 內進行。在您提供存取權杖、連接專案並在 AWS 管理主控台中啟用程式碼檢閱註解後，AWS Security Agent 會自動掃描每個新合併請求中的變更（僅對變更的程式碼進行差異掃描），並將調查結果發佈為合併請求註解。

您可以在 AWS Security Agent Web 應用程式中建立並執行完整的程式碼檢閱，也就是掃描專案的整個程式碼庫，而不是在 GitLab 中。

滲透測試和威脅建模會在 AWS Security Agent Web 應用程式內啟動。使用者指定目標網域，然後選取連線的儲存庫以提供應用程式內容。如果您啟用自動修復，使用者可以透過開啟對連線儲存庫的合併請求，請求 AWS Security Agent 修正問題清單。

> **Note**
>
> 公有 GitLab 儲存庫無法使用自動化修復，以避免在漏洞修正之前揭露漏洞。

先決條件

開始前，請確保您具備以下條件：

- 具有您想要連線之專案維護者或擁有者存取權的 GitLab.com 帳戶
- 具有連線類型所需範圍的 GitLab 存取字符：
  - 個人 - 具有所有讀取許可和 `api` 許可的個人存取字符。
  - 群組 - 具有 `read_api`和 `read_repository`範圍的群組存取字符。
- 在 AWS Security Agent Management Console 中設定整合的許可

> **Important**
>
> 將字符過期設定為目前日期後最多 365 天。

> **Note**
>
> GitLab Personal Access Token 可用於多個 AWS 帳戶。與 GitHub 和 Atlassian 整合不同，將相同的 GitLab 帳戶連線至多個 AWS Security Agent 執行個體沒有限制。

註冊 GitLab 連線

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 GitLab，然後選擇下一步。
4. 在選擇帳戶類型下，選取下列其中一項：
   - Personal - 連接您的個別 GitLab 使用者帳戶。
   - 群組 - 連接包含多個專案的 GitLab 群組。輸入您的群組 ID。
5. 在存取字符欄位中，貼上您的 GitLab 存取字符。
6. 在註冊名稱欄位中，輸入此連線的描述性名稱，例如 `Engineering-Team-GitLab`。有效字元包括字母、數字、句點、底線和連字號。
7. 選擇連線。

   您會返回整合頁面，其中會顯示新連線及其註冊名稱。

故障診斷 GitLab 整合

如果您在 GitLab 整合程序期間遇到問題，請使用下列指引來解決常見問題。

字符無效或過期

徵狀

- 整合無法連線
- 先前運作的整合會停止運作

- 指出身分驗證失敗的錯誤訊息

Resolution

1. 在 GitLab 中，導覽至您的使用者設定，然後選取存取字符。
2. 確認您的字符尚未過期。
3. 確認字符具有其類型所需的範圍。
4. 如果字符已過期，請建立新的字符，並在 AWS 主控台中更新整合。

速率限制

徵狀

- 程式碼檢閱期間的間歇性失敗
- 延遲合併請求分析

Resolution

- GitLab 會將速率限制套用至 API 請求。如果您有大量的儲存庫或頻繁的合併請求，某些請求可能會受到調節。
- 等待速率限制時段重設，或聯絡 GitLab Support 以提高速率限制。

後續步驟

將 GitLab 連線至 AWS Security Agent 之後：

- 導覽至您要使用這些儲存庫的客服人員空間
- 選擇啟用程式碼檢閱或設定滲透測試，將特定專案連線至您的客服人員空間，並設定其用量
- 啟用程式碼修復，以允許 AWS Security Agent 使用漏洞修正來提交合併請求
- 如需私有託管的 GitLab 執行個體，請參閱 the section called “將 AWS 安全代理程式連線至 GitLab 自我管理”

將 AWS 安全代理程式連線至 GitLab 自我管理

將您的 AWS 安全代理程式連線至 GitLab 自我管理執行個體，以針對託管在您自己的基礎設施上的儲存庫啟用程式碼檢閱、威脅建模、滲透測試和自動化修復功能。

GitLab 自我管理整合的運作方式與 GitLab Cloud （請參閱the section called “將 AWS 安全代理程式連線至 GitLab 儲存庫”) 相同，具有額外的組態，可連線至您的私有執行個體。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 之間重複使用註冊並跨 功能共用。

> **Note**
>
> GitLab 自我管理是透過 GitLab 整合註冊，而不是單獨的整合類型。在註冊流程中，您選取使用 GitLab 自我託管端點，並提供執行個體 URL。雲端託管的 GitLab 流程如 中所述the section called “將 AWS 安全代理程式連線至 GitLab 儲存庫”。

先決條件

開始前，請確保您具備以下條件：

- 符合下列任一條件的 GitLab 自我管理執行個體：
  - 透過網際網路公開存取，或
  - 可透過私有連線存取 （請參閱 the section called “連線至私有託管來源控制”)
- 具有連線類型所需範圍的 GitLab 存取字符：
  - Personal - 具有所有讀取許可和`api`許可的個人存取字符。
  - 群組 - 具有 `read_api`和 `read_repository`範圍的群組存取字符。
- 維護者或擁有者存取您要連線的專案
- 您的 GitLab 執行個體必須提供最低 TLS 版本為 1.2 的 HTTPS 流量

> **Note**
>
> 如果您的 GitLab 自我管理執行個體使用私有憑證授權單位發行的 TLS 憑證，您可以在建立私有連線時提供憑證的 PEM 編碼公有金鑰。這可讓 AWS Security Agent 信任執行個體的 TLS 連線。

註冊 GitLab 自我管理連線

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 GitLab，然後選擇下一步。
4. 在選擇帳戶類型下，選取個人或群組。如果您選取群組，請輸入您的群組 ID。
5. 選取使用 GitLab 自我託管端點。
6. 在 GitLab 自我託管端點 URL 欄位中，輸入執行個體的 URL，例如 `https://gitlab.example.com`。
7. 如果您的執行個體無法公開存取，請選取使用私有連線連線至端點，然後選擇現有的私有連線或建立新的私有連線。請參閱 the section called “連線至私有託管來源控制”。
8. 在存取字符欄位中，貼上您的 GitLab 存取字符。
9. 在註冊名稱欄位中，輸入此連線的描述性名稱。有效字元包括字母、數字、句點、底線和連字號。
10. 選擇連線。

    您會返回整合頁面，其中會顯示新連線及其註冊名稱。

私有連線

如果您的 GitLab 自我管理執行個體無法公開存取，您必須在註冊整合之前建立私有連線。如需詳細說明the section called “連線至私有託管來源控制”，請參閱 。

> **Important**
>
> 服務受管私有連線需要在建立 Agent Space 的相同 AWS 帳戶中執行 GitLab 自我管理執行個體。對於跨帳戶存取，請使用自我管理的私有連線，您可以在其中提供自己的 VPC Lattice 資源組態。

故障診斷 GitLab 自我管理整合

除了 中的疑難排解步驟之外the section called “將 AWS 安全代理程式連線至 GitLab 儲存庫”，下列問題專屬於自我管理的執行個體：

無法連線執行個體

徵狀

- 連線失敗，並出現逾時或網路錯誤
- 整合之前正常運作，但停止運作

Resolution

- 驗證您的 GitLab 執行個體是否正在執行並可存取
- 如果使用私有連線，請確認 VPC Lattice 資源閘道運作狀態良好，且 ENIs具有執行個體的網路連線能力
- 驗證安全群組是否允許已設定連接埠上的流量
- 驗證 TLS 憑證是否有效且未過期

TLS 憑證錯誤

徵狀

- 連線失敗並出現 SSL/TLS 錯誤

Resolution

- 驗證您的執行個體是否提供具有 TLS 1.2 或更新版本的 HTTPS
- 如果使用私有憑證授權單位，請確保在私有連線設定期間提供 PEM 編碼的公有金鑰
- 確認憑證未過期

後續步驟

將 GitLab 自我管理連線至 AWS 安全代理程式之後：

- 導覽至您要使用這些儲存庫的客服人員空間
- 選擇啟用程式碼檢閱或設定滲透測試以連接特定專案
- 啟用合併請求型修正的程式碼修復

移除 GitLab 整合

當您不再需要 AWS 安全代理程式從特定 GitLab 帳戶或群組存取儲存庫時，請移除 GitLab 整合。

移除的先決條件

在移除 GitLab 整合之前，請確定您有：

- 檢查哪些 Agent Spaces 具有從此整合連線的儲存庫
- 了解影響：移除此整合將中斷儲存庫連線，以便使用此整合中的儲存庫進行程式碼檢閱、滲透測試內容、威脅建模和自動修復所有客服人員空間

從 AWS Security Agent 移除整合

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 找到您要移除的 GitLab 整合。
3. 按一下以選取整合。
4. 選擇移除。
5. 檢閱確認對話方塊，然後選擇確認移除。

> **Note**
>
> 移除整合之後，您可能也想要撤銷 GitLab 中的個人存取字符，以防止進一步存取。導覽至您的 GitLab 使用者設定，並刪除或撤銷權杖。

相同的程序適用於 GitLab 自我管理整合。

將 AWS 安全代理程式連線至 GitHub Enterprise Server

將您的 AWS 安全代理程式連線至 GitHub Enterprise Server (GHES) 執行個體，以針對您自己基礎設施上託管的儲存庫啟用程式碼檢閱、威脅建模、滲透測試和自動化修復功能。

GitHub Enterprise Server 整合提供與雲端託管的 GitHub 相同的功能（請參閱將 AWS 安全代理程式連線至 GitHub 儲存庫），以及額外的組態，可讓您連線至自我託管執行個體的網路連線。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 之間重複使用註冊並跨 功能共用。

> **Note**
>
> GitHub Enterprise Server 透過 GitHub 整合註冊，而不是單獨的整合類型。在註冊流程中，您可以選擇 GitHub Enterprise Server 做為執行個體類型。雲端託管的 GitHub.com 流程如中所述。 the section called “將 AWS 安全代理程式連線至 GitHub 儲存庫”

GitHub Enterprise Server 整合的運作方式

提取請求分析會在 GitHub Enterprise Server 內進行。在您授權連線、連接儲存庫，並在 AWS 管理主控台中啟用程式碼檢閱註解後，AWS Security Agent 會自動掃描每個新提取請求中的變更（僅對變更的程式碼進行差異掃描），並將調查結果發佈為提取請求註解。

您可以在 AWS Security Agent Web 應用程式中，而不是在 GitHub Enterprise Server 中建立和執行完整的程式碼檢閱，以掃描儲存庫的整個程式碼庫。

滲透測試和威脅建模會在 AWS Security Agent Web 應用程式內啟動。使用者指定目標網域，然後選取連線的儲存庫以提供應用程式內容。如果您啟用自動修復，使用者可以向連線的儲存庫開啟提取請求，請求 AWS Security Agent 修正問題清單。

先決條件

開始前，請確保您具備以下條件：

- 符合以下任一條件的 GitHub Enterprise Server 執行個體：
  - 透過網際網路公開存取，或
  - 可透過私有連線存取（請參閱 the section called “連線至私有託管來源控制”)
- GHES 執行個體上的網站管理員或組織管理員存取權
- 您的 GHES 執行個體必須提供最低 TLS 版本為 1.2 的 HTTPS 流量
- 在 AWS Security Agent Management Console 中設定整合的許可

> **Note**
>
> GitHub Enterprise Server 整合可用於多個 AWS 帳戶。

註冊 GitHub Enterprise Server 連線

註冊 GitHub Enterprise Server 連線會使用以 OAuth 為基礎的授權流程。

> **Important**
>
> 完成此程序中的所有步驟，無需關閉瀏覽器或瀏覽。如果註冊程序中斷，您可能需要從頭開始重新啟動。

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 GitHub，然後選擇下一步。
4. 在執行個體類型下，選取 GitHub Enterprise Server。
5. 在 GitHub Enterprise Server URL 欄位中，輸入執行個體的 HTTPS URL，例如 `https://github.example.com`。
6. 如果您的執行個體無法公開存取，請選取使用私有連線連線至端點，然後選擇現有的私有連線或建立新的私有連線。請參閱 the section called “連線至私有託管來源控制”。
7. 在註冊詳細資訊區段中，設定下列欄位：
   a. 註冊名稱 - 輸入此連線的描述性名稱。有效字元包括字母、數字、句點、底線和連字號。
   b. GitHub 帳戶類型 - 選取組織或使用者。
   c. 組織名稱（只有在您選取組織時才會顯示） - 輸入 GitHub Enterprise Server 組織的確切名稱。名稱區分大小寫。
8. 選擇連線。

> **Note**
>
> AWS Security Agent 會將您從主控台重新導向，以完成 GitHub Enterprise Server 執行個體的授權。授權完成後，您會返回 主控台，並在整合頁面上顯示新的連線。

私有連線

如果您的 GitHub Enterprise Server 執行個體無法公開存取，您必須在註冊整合之前建立私有連線。如需詳細說明the section called “連線至私有託管來源控制”，請參閱 。

> **Important**
>
> 服務受管私有連線需要在建立 Agent Space 的相同 AWS 帳戶中執行 GHES 執行個體。對於跨帳戶存取，請使用自我管理的私有連線。

> **Note**
>
> 如果您的 GHES 執行個體使用私有憑證授權單位發行的 TLS 憑證，請在建立私有連線時提供憑證的 PEM 編碼公有金鑰。

故障診斷 GitHub Enterprise Server 整合

如果您在將 AWS Security Agent 連線至 GitHub Enterprise Server 時遇到問題，請使用下列指引來診斷和解決常見問題。

OAuth 重新導向失敗

徵狀

- 瀏覽器重新導向在授權流程中失敗
- 在 GHES 上授權後顯示的錯誤頁面

Resolution

- 確認您的 GHES 執行個體可從瀏覽器存取
- 確保已正確設定 OAuth 回呼 URL
- 從頭開始重新啟動整合程序

無法連線執行個體

徵狀

- 連線失敗，並出現逾時或網路錯誤

Resolution

- 驗證您的 GHES 執行個體是否正在執行並可存取
- 如果使用私有連線，請驗證 VPC Lattice 連線
- 驗證安全群組是否允許已設定連接埠上的流量
- 驗證 TLS 憑證是否有效 （最低 TLS 1.2)

後續步驟

將 GitHub Enterprise Server 連線至 AWS Security Agent 之後：

- 導覽至您要使用這些儲存庫的客服人員空間
- 選擇啟用程式碼檢閱或設定滲透測試，以連接特定儲存庫
- 啟用程式碼修復，以允許 AWS Security Agent 提交具有漏洞修正的提取請求

移除 GitHub Enterprise Server 整合

當您不再需要 AWS Security Agent 從特定 GHES 執行個體存取儲存庫時，請移除 GitHub Enterprise Server 整合。

移除的先決條件

移除 GHES 整合之前：

- 檢查哪些 Agent Spaces 具有從此整合連線的儲存庫
- 了解影響：移除 將中斷所有連線儲存庫的程式碼檢閱、滲透測試內容、威脅建模和自動化修復

移除整合

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 找到您要移除的 GitHub Enterprise Server 整合。
3. 選取整合。
4. 選擇移除。
5. 檢閱確認對話方塊，然後選擇確認移除。

> **Note**
>
> 移除後，您可能也想要撤銷 GHES 執行個體上的 OAuth 應用程式。導覽至您的 GHES 組織設定，並移除授權的應用程式。

尋找您的 Atlassian 安裝 ID

在 AWS 主控台中註冊 Bitbucket 或 Confluence 整合之前，您需要從安裝在 Atlassian 網站上的 AWS Security Agent Forge 應用程式找到安裝 ID。將此 ID 貼到註冊流程中。

對於 Bitbucket

1. 在 Bitbucket 中，導覽至工作區設定。
2. 從邊欄中選取 Forge 應用程式。
3. 在已安裝的應用程式清單中找到與 AWS Security Agent 的連線。
4. 選擇複製到剪貼簿以複製安裝 ID。

針對 Confluence

1. 在 Confluence 中，導覽至 Confluence 管理。
2. 從側邊列選取應用程式。
3. 在已安裝的應用程式清單中找到與 AWS Security Agent 的連線。
4. 選擇複製到剪貼簿以複製安裝 ID。

在 AWS Security Agent Management Console 中完成註冊時，您將需要此安裝 ID。

將 AWS Security Agent 連線至 Bitbucket 儲存庫

將您的 AWS Security Agent 連線至 Bitbucket Cloud 儲存庫，以啟用程式碼檢閱、威脅建模、滲透測試和自動化修復功能。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 之間重複使用註冊並跨 功能共用。

Bitbucket 整合提供多種用途：

- 程式碼檢閱 - 根據組織安全需求自動分析每個提取請求中的程式碼變更，並執行隨需完整儲存庫掃描

- 威脅建模 - 透過分析原始程式碼、資料流程和架構來提供應用程式理解
- 滲透測試內容 - 提供滲透測試的應用程式理解
- 自動化修復 - 提交提取請求，並修正安全性評估期間發現的漏洞

將 Bitbucket 連線至 AWS Security Agent 需要在 Atlassian 網站上安裝 AWS Security Agent Forge 應用程式，並完成 OAuth 授權流程。

> **Note**
>
> AWS Security Agent 僅支援 Bitbucket Cloud。不支援 Bitbucket Server 和 Bitbucket Data Center。

Bitbucket 整合的運作方式

提取請求分析會在 Bitbucket 內進行。在您安裝 Forge 應用程式、連接儲存庫，並在 AWS 管理主控台中啟用程式碼檢閱註解後，AWS Security Agent 會自動掃描每個新提取請求中的變更（僅變更程式碼的差別掃描），並將調查結果發佈為提取請求註解。

您可以在 AWS Security Agent Web 應用程式中，而不是在 Bitbucket 中建立並執行完整的程式碼檢閱，以掃描儲存庫的整個程式碼庫。

滲透測試和威脅建模會在 AWS Security Agent Web 應用程式內啟動。使用者指定目標網域，然後選取連線的儲存庫以提供應用程式內容。

> **Note**
>
> 自動化修復不適用於公有 Bitbucket 儲存庫，以避免在漏洞修正之前予以揭露。

先決條件

開始前，請確保您具備以下條件：

- 具有管理員存取權的 Bitbucket Cloud 工作區
- 具有網站管理員權限的 Atlassian 帳戶
- 在 AWS Security Agent Management Console 中設定整合的許可

Important

一個 Atlassian 網站只能與每個區域的一個 AWS 帳戶相關聯。如果您需要將相同的 Bitbucket 網站連線到相同區域內不同 AWS 帳戶中的 AWS 安全代理程式，您必須先移除現有的整合。

Note

Atlassian Forge 應用程式定價適用於此整合。如需詳細資訊，請參閱 Atlassian 文件中的 Forge 平台定價。

註冊 Bitbucket 連線

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 Bitbucket，然後選擇下一步。
4. 遵循螢幕上的指示，在 Bitbucket 工作區中安裝 AWS Security Agent Forge 應用程式。安裝之後，請複製安裝 ID。請參閱 the section called “尋找您的 Atlassian 安裝 ID”。
5. 在安裝 ID 欄位中，貼上您從 Bitbucket 複製的安裝 ID。
6. 在 Bitbucket 工作區欄位中，輸入您的 Bitbucket 工作區 Slug，例如 `acme-corp`。
7. 選擇 Authorize (授權)。

   系統會將您重新導向至 Atlassian，以授權 AWS Security Agent 存取您的 Bitbucket 工作區。授權完成後，您會返回 主控台。
8. 在註冊詳細資訊區段中，輸入此連線的註冊名稱。有效字元包括字母、數字、句點、底線和連字號。
9. 選擇連線。

Note

連線後佈建延遲

選擇 Connect 後，在 Atlassian 端佈建可能需要幾分鐘的時間。在此期間，整合會報告待定狀態。如果您在佈建完成之前嘗試選取儲存庫或啟用程式碼檢閱，您可能會看到安裝尚未提供的訊息。請等待幾分鐘後再試一次。

針對 Bitbucket 整合進行故障診斷

如果您在 Bitbucket 整合過程中遇到問題，請使用下列指引來解決常見問題。

無法完成註冊

如果註冊程序中斷（瀏覽器關閉、工作階段逾時），Forge 應用程式可能安裝在您的 Atlassian 網站上，但未在 AWS 主控台中註冊。

Resolution

- 返回 AWS Security Agent 主控台並重新啟動整合程序。
- Forge 應用程式會保持安裝狀態，不需要重新安裝。

網站已連線至另一個 AWS 帳戶

徵狀

- 指出 Atlassian 網站已與其他帳戶建立關聯的錯誤

Resolution

- 一個 Atlassian 網站只能連接到每個區域的一個 AWS 帳戶。
- 識別哪些 AWS 帳戶具有現有的整合並使用該帳戶，或先移除現有的整合。

安裝無法使用

徵狀

- 當您選取儲存庫或啟用程式碼檢閱時，您會看到 Bitbucket 安裝處於 狀態的訊息`PENDING_INSTALLATION`，而不是 `AVAILABLE`。

Resolution

- 安裝仍在 Atlassian 端佈建。佈建通常會在幾分鐘內完成。請稍候幾分鐘，然後再試一次。
- 如果狀態在幾分鐘後仍無法使用，請移除整合並再次註冊。重新註冊之前，請從 Bitbucket 工作區解除安裝 Forge 應用程式。如需說明，請參閱移除 Bitbucket 整合。如果您在未解除安裝先前的 Forge 應用程式的情況下重新註冊，安裝可能會卡在待定狀態。

後續步驟

將 Bitbucket 連線至 AWS Security Agent 之後：

- 導覽至您要使用這些儲存庫的客服人員空間
- 選擇啟用程式碼檢閱或設定滲透測試，將特定儲存庫連線至您的客服人員空間
- 啟用程式碼修復，以允許 AWS Security Agent 提交具有漏洞修正的提取請求

移除 Bitbucket 整合

當您不再需要 AWS Security Agent 從特定的 Bitbucket 工作區存取儲存庫時，請移除 Bitbucket 整合。

移除的先決條件

在移除 Bitbucket 整合之前：

- 檢查哪些 Agent Spaces 具有從此整合連線的儲存庫
- 了解影響：移除 將中斷所有連線儲存庫的程式碼檢閱、滲透測試內容、威脅建模和自動化修復

步驟 1：從 AWS Security Agent 移除整合

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 找到您要移除的 Bitbucket 整合。
3. 選取整合。
4. 選擇移除。
5. 檢閱確認對話方塊，然後選擇確認移除。

步驟 2：解除安裝 Forge 應用程式

從 AWS Security Agent 移除整合後，從您的 Atlassian 網站解除安裝 Forge 應用程式：

1. 在 Bitbucket 中，導覽至工作區設定。
2. 選取 Forge 應用程式。
3. 尋找與 AWS Security Agent 的連線。
4. 選擇解除安裝。

> **Important**
>
> Forge 應用程式不會自動解除安裝
>
> 在 AWS Security Agent 主控台中移除整合並不會從您的 Atlassian 網站解除安裝 Forge 應用程式。如果您打算再次註冊相同的 Bitbucket 工作區，您必須先解除安裝 Forge 應用程式。否則，新安裝可能會卡在待定狀態。

將 AWS 安全代理程式連線至 Confluence

將您的 AWS Security Agent 連線至 Confluence Cloud，以提供安全評估的文件內容。與程式碼提供者不同，Confluence 可做為文件來源，提供威脅模型、架構文件、API 規格，以及其他可增強安全審查品質的資料。開始之前，請檢閱 the section called “整合如何與 Agent Spaces 搭配使用” 以了解如何在 Agent Spaces 中重複使用註冊。

Confluence 整合提供多種用途：

- 設計審查內容 - 為安全設計審查提供架構文件和設計規格
- 威脅建模 - 為威脅分析提供現有的威脅模型和系統文件
- 滲透測試內容 - 提供應用程式文件，以在滲透測試期間深入了解

將 Confluence 連接到 AWS Security Agent 需要在 Atlassian 網站上安裝 AWS Security Agent Forge 應用程式並完成 OAuth 授權流程。

> **Note**
>
> AWS Security Agent 僅支援 Confluence Cloud。不支援 Confluence Data Center 和 Confluence Server。

Confluence 整合的運作方式

Confluence 是文件提供者，而不是原始程式碼提供者。在 AWS 管理主控台中安裝 Forge 應用程式並連接空間或頁面後，AWS Security Agent 可以存取您的 Confluence 內容，以在安全評估期間提供內容。

AWS Security Agent 會讀取頁面內容，以了解您的應用程式架構、安全需求和設計決策。此內容可改善設計檢閱、程式碼檢閱和滲透測試期間安全調查結果的品質和相關性。

先決條件

開始前，請確保您具備以下條件：

- 具有管理員存取權的 Confluence Cloud 網站
- 具有網站管理員權限的 Atlassian 帳戶
- 在 AWS Security Agent Management Console 中設定整合的許可

> **Important**
>
> 一個 Atlassian 網站只能與每個區域的一個 AWS 帳戶相關聯。如果您需要將相同的 Confluence 網站連線到相同區域內不同 AWS 帳戶中的 AWS Security Agent，您必須先移除現有的整合。

> **Note**
>
> Atlassian Forge 應用程式定價適用於此整合。如需詳細資訊，請參閱 Atlassian 文件中的 Forge 平台定價。

註冊 Confluence 連線

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇新增整合作業。
3. 選取 Confluence，然後選擇下一步。
4. 遵循螢幕上的指示，在您的 Atlassian 網站安裝 AWS Security Agent Forge 應用程式。安裝之後，請複製安裝 ID。請參閱 the section called “尋找您的 Atlassian 安裝 ID”。
5. 在 Confluence 網站 URL 欄位中，輸入您的網站 URL，例如 `https://acme.atlassian.net`。
6. 在安裝 ID 欄位中，貼上您從 Confluence 複製的安裝 ID。
7. 選擇 Authorize (授權)。

   系統會將您重新導向至 Atlassian，以授權 AWS Security Agent 存取您的 Confluence 網站。授權完成後，您會返回 主控台。
8. 在註冊詳細資訊區段中，輸入此連線的註冊名稱。有效字元包括字母、數字、句點、底線和連字號。
9. 選擇連線。

選取客服人員空間的頁面

註冊 Confluence 整合之後，請將特定頁面連接至 代理程式空間。選取頁面可讓 AWS Security Agent 讀取（擷取）存取該頁面的內容。沒有每個頁面的功能選項 — 代理程式會讀取每個連線的頁面。在連線精靈的檢閱步驟中，您可以移除不希望代理程式存取的任何頁面。

Confluence 整合疑難排解

如果您在 Confluence 整合程序期間遇到問題，請使用下列指引來解決常見問題。

無法完成註冊

如果註冊程序中斷，Forge 應用程式可能安裝在您的 Atlassian 網站上，但未在 AWS 主控台中註冊。

Resolution

- 返回 AWS Security Agent 主控台並重新啟動整合程序。
- Forge 應用程式會保持安裝狀態，不需要重新安裝。

從 Atlassian 解除安裝的 Forge 應用程式

如果 AWS Security Agent Forge 應用程式已從 Atlassian 網站解除安裝，而整合仍存在於 AWS Security Agent 中：

徵狀

- 整合會顯示在 AWS 主控台中，但無法存取 Confluence 內容
- 嘗試列出或讀取頁面時發生錯誤

Resolution

- 從 Atlassian Marketplace 重新安裝 Forge 應用程式
- 如果問題仍然存在，請在 AWS 主控台中移除整合並重新註冊

網站已連線至另一個 AWS 帳戶

Resolution

- 一個 Atlassian 網站只能連接到每個區域的一個 AWS 帳戶。
- 識別哪些 AWS 帳戶具有現有的整合並使用該帳戶，或先移除現有的整合。

後續步驟

將 Confluence 連線至 AWS Security Agent 之後：

- 導覽至您要使用本文件的客服人員空間
- 選取要包含做為設計檢閱和滲透測試內容的特定頁面
- 如有需要，請透過 S3 上傳其他文件（請參閱 the section called “從 S3 儲存貯體提供代理程式資源”）

移除 Confluence 整合

當您不再需要 AWS 安全代理程式從特定 Confluence 網站存取文件時，請移除 Confluence 整合。

移除的先決條件

在移除 Confluence 整合之前：

- 檢查哪些 Agent Spaces 具有從此整合連線的頁面
- 了解影響：移除 將破壞所有 Agent Spaces 使用來自此整合的內容進行設計審查、滲透測試和威脅建模的文件內容

步驟 1：從 AWS Security Agent 移除整合

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 找到您要移除的 Confluence 整合。
3. 選取整合。
4. 選擇移除。
5. 檢閱確認對話方塊，然後選擇確認移除。

步驟 2：解除安裝 Forge 應用程式 （選用）

從 AWS Security Agent 移除整合之後，您可以選擇從 Atlassian 網站解除安裝 Forge 應用程式：

1. 在 Confluence 中，導覽至 Confluence 管理。
2. 選取應用程式。
3. 尋找與 AWS Security Agent 的連線。
4. 選擇解除安裝。

連線至私有託管來源控制

> **Important**
>
> 私有連線為 AWS Security Agent 的預覽版本，可能會有所變更。

AWS Security Agent 可以連線到在私有網路中執行的來源控制系統，例如 GitLab 自我管理執行個體和 GitHub Enterprise Server。私有連線使用 Amazon VPC Lattice 在 AWS Security Agent 與您的私有基礎設施之間建立安全連線，而不會將您的系統暴露在公有網際網路。

私有連線的運作方式

私有連線會在 AWS Security Agent 與 VPC 中的目標資源之間建立安全的網路路徑。在幕後，AWS Security Agent 會使用 Amazon VPC Lattice 來建立此連線。VPC Lattice 是一種 AWS 應用程式聯網服務，可在 VPCs 和帳戶之間連線、保護和監控服務之間的流量，而無需管理基礎網路基礎設施。

當您建立私有連線時：

1. 您可以提供具有目標服務網路連線能力的 VPC、子網路和 （選用） 安全群組。
2. AWS Security Agent 會建立服務管理的資源閘道，並在您指定的子網路中佈建彈性網路介面 (ENIs)。
3. 代理程式會使用 資源閘道，透過私有網路路徑將流量路由到目標服務的 IP 地址或 DNS 名稱。

服務受管與自我管理的私有連線

AWS Security Agent 支援兩種私有連線模式：

服務受管 （建議簡單設定）：

- AWS Security Agent 會為您建立和管理 VPC Lattice 資源閘道。
- 目標服務必須在建立 Agent Space 的相同 AWS 帳戶中執行。
- 無法跨越多個 AWS 帳戶。

自我管理 （適用於進階或跨帳戶設定）：

- 您可以建立和管理自己的 VPC Lattice 資源組態。
- 您提供現有資源組態的 Amazon Resource Name (ARN)。
- 支援跨帳戶存取 （客戶管理跨帳戶聯網）。

先決條件

建立私有連線之前，請確認您已：

- 作用中的客服人員空間
- 可使用已知私有 IP 地址或 DNS 名稱存取的私有目標服務 (GitLab 自我管理或 GitHub Enterprise Server)

- 目標服務必須提供最低 TLS 版本為 1.2 的 HTTPS 流量
- VPC 中的子網路 (1-20 個子網路）。建議您選取多個可用區域中的子網路，以獲得高可用性。
- （選用）用於控制 ENIs（最多 5 個）

> **Note**
>
> VPC Lattice 不支援下列可用區域：use1-az3、usw1-az2、apne1-az3、apne2-az2、euc1-az2、euw1-az4、cac1-az3、ilc1-az2。

> **Note**
>
> 私有連線是帳戶層級的資源。建立私有連線後，您可以在需要連接相同主機的多個整合和客服人員空間中重複使用該連線。

> **Note**
>
> 當您為使用 OAuth 身分驗證的提供者選取私有連線時，私有連線會同時套用至提供者端點和字符交換端點。確保私有連線已設定主機地址，可將流量路由到兩個端點。

建立私有連線

1. 在 AWS 安全代理程式管理主控台中，導覽至整合。
2. 選擇私有連線索引標籤。
3. 選擇建立私有連線。
4. 在連線詳細資訊下，輸入名稱。名稱可以包含小寫字母、數字和連字號，且必須以字母或數字開頭和結尾。
5. 在連線詳細資訊區段中，選擇 AWS Security Agent 如何連線至您的目標服務：
   - 若要讓 AWS Security Agent 建立和管理連線，請保持清除使用現有資源組態，然後完成資源位置、存取控制和服務目標詳細資訊區段。
   - 若要透過您已建立的 VPC Lattice 資源組態進行路由，請選取使用現有資源組態，然後完成現有資源組態區段。服務受管區段不適用。
6. （服務受管）在資源位置下：

a. 選取資源所在的 VPC。

b. 選取一或多個子網路。建議您至少選取兩個可用區域中的子網路。

c. （選用）選取 IP 地址類型 (IPv4、IPv6 或雙堆疊）。

7. （服務受管）在存取控制下：

a. 選取與您連線資源相關聯的安全群組。

b. （選用）在進階組態下，輸入連接埠範圍 — 最多 11 個逗號分隔的 TCP 連接埠或範圍，例如 `443, 8080-8090`。

8. （服務受管）在服務目標詳細資訊下：

a. 在主機地址欄位中，輸入目標服務的 DNS 主機名稱或 IP 地址。

b. 對於 DNS 解析，為可公開解析的主機地址選取公有，或在 VPC （私有 DNS) 中為只能在 VPC 內解析的地址選取公有。

c. （選用）如果您的目標使用自我簽署憑證或私有憑證授權單位，請在憑證公有金鑰欄位中輸入 PEM 編碼的公有金鑰。這可讓 AWS Security Agent 信任 TLS 連線。

9. （自我管理）在現有資源組態下，針對資源組態，從清單中選擇 VPC Lattice 資源組態，或輸入資源組態 ID （從 開始`rcfg-`) 或其 ARN。此清單顯示目前區域中的資源組態。

10選擇建立連線。

連線狀態會變更為建立進行中。這最多可能需要 10 分鐘。完成後，狀態會變更為可用。

搭配 整合使用私有連線

當您註冊 GitLab 自我管理或 GitHub Enterprise Server 整合時，請選取使用私有連線連線至端點，然後從私有連線清單中選擇您的連線。AWS Security Agent 會透過該連線將流量路由至提供者。只有具有可用狀態的連線才會出現在清單中。

您也可以在註冊期間選擇建立私有連線。AWS Security Agent 會在您建立連線時保留您的註冊草稿，然後將您傳回至註冊流程。

安全

- 沒有公有網際網路暴露 - 所有流量都會保留在 AWS 網路上。
- 客戶控制的安全群組 - 您可以管理傳入和傳出流量規則。
- 具有最低權限的服務連結角色 - AWS Security Agent 使用服務連結角色，範圍限定為以 標記的資源`AWSSecurityAgentManaged`。

> **Note**
>
> 如果您的組織具有限制 VPC Lattice API 動作的服務控制政策 (SCPs)，則服務受管資源閘道會透過服務連結角色建立。確保您的 SCPs服務連結角色的必要動作。

驗證私有連線

在私有連線達到可用狀態後，驗證連線：

- 確保您的目標服務正在執行並接受預期連接埠上的連線
- 驗證連接到 ENIs的安全群組是否允許目標連接埠上的傳出流量
- 驗證服務的安全群組是否允許來自 VPC CIDR 範圍內 VPC Lattice 資料平面 IPs傳入流量
- 確認子網路路由表允許 ENI 子網路與服務之間的流量

刪除私有連線

1. 在 AWS Security Agent 主控台中，導覽至整合。
2. 選擇私有連線索引標籤。
3. 選取您要刪除的私有連線。
4. 選擇 刪除。

> **Important**
>
> 您無法刪除 整合目前正在使用的私有連線。首先移除整合，然後刪除私有連線。

後續步驟

- the section called “將 AWS 安全代理程式連線至 GitLab 自我管理” - 連接私有 GitLab 執行個體
- the section called “將 AWS 安全代理程式連線至 GitHub Enterprise Server” - 連接私有 GitHub Enterprise Server

將代理程式連線至私有 VPC 資源

如果您想要在 上執行滲透測試的應用程式無法在公有網際網路上使用，您需要向 AWS Security Agent 提供 VPC 組態。AWS Security Agent 將使用此 VPC 組態來存取應用程式，包括 VPC、子網路和安全群組。

> **Note**
>
> 在私有 VPC 中測試端點時，僅允許解析為已知私有 IPs的端點 （ 如需詳細資訊，請參閱 VPC CIDR 區塊)。允許下列 IPv4 和 IPv6 範圍：
>
> ```
> 10.0.0.0/8
> 172.16.0.0/12
> 192.168.0.0/16
> fd00::/8
> ```

> **Note**
>
> 連線至子網路時，AWS Security Agent 會在為滲透測試設定的子網路中建立 ENI (彈性網路界面)。此 ENI 沒有相關聯的公有 IP 地址，這表示無法與公有子網路中的 VPC 網際網路閘道通訊。如果您的滲透測試需要開放網際網路存取，請改用具有相關聯 VPC NAT Gateway 的私有子網路

您可以從 AWS 管理主控台授予 VPC 的一般存取權。在 Security Agent Web 應用程式中，使用者會選取滲透測試的特定組態。

在客服人員空間中新增 VPC

1. 導覽至客服人員空間概觀頁面
2. 選取動作，然後選取編輯滲透測試組態
3. 在 VPC 標題下，指定 VPC、子網路和安全群組

您最多可以新增 5 VPCs。

必要的客服人員空間服務角色許可

若要使用 VPC 執行滲透測試，您的代理程式空間服務角色必須包含下列許可：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "ec2:CreateNetworkInterfacePermission",
      "Resource": "arn:aws:ec2:{{region}}:{{accountId}}:network-interface/*",
      "Condition": {
        "ArnEquals": {
          "ec2:Subnet": [
            "arn:aws:ec2:{{region}}:{{accountId}}:subnet/[[subnetIds]]"
          ]
        }
      }
    }
  ]
}
```

若要在 Security Agent Web 應用程式中選取滲透測試的特定 VPC 組態

1. 導覽至滲透測試概觀頁面
2. 選取您需要為其新增 VPC 組態的滲透測試，然後選擇修改滲透測試詳細資訊
3. 選取下一步以到達 VPC 資源區段

4. 選取 VPC、子網路和安全群組
5. 選取下一步以到達最後一個區段，並儲存滲透測試

> **Note**
>
> 使用 AWS Resource Access Manager 共用的 VPC 資源（子網路和安全群組）目前支援跨帳戶滲透測試。用於身分驗證登入資料的 Secrets Manager 秘密和 Lambda 函數必須在與您的 AWS Security Agent 設定相同的 AWS 帳戶中設定。

針對另一個 AWS 帳戶中的 VPC 資源執行滲透測試

您可以使用 AWS Resource Access Manager，針對與您的帳戶共用的 VPC 資源執行滲透測試。兩個帳戶都必須屬於相同的 AWS Organization。

1. （選用）為您的 AWS 組織啟用自動資源共用

```
aws ram enable-sharing-with-aws-organization
```

2. 使用擁有 VPC 資源之 AWS 帳戶的登入資料，與滲透測試擁有者帳戶共用子網路和安全群組資源

```
aws ram create-resource-share \
    --name SharePentestResources \
    --resource-arns <subnet ARN> <security group ARN> \
    --principals <penetration test owner account ID>
```

3. 導覽至客服人員空間概觀頁面
4. 選取滲透測試並尋找服務角色名稱
5. 驗證 IAM 角色是否授予共用 VPC 資源的存取權
6. 選取動作，然後編輯滲透測試組態
7. 在 VPC 標題下，指定共用 VPC、子網路和安全群組，並儲存更新的組態。
8. 導覽至 AWS Security Agent Web 應用程式上的滲透測試概觀頁面
9. 選取您需要為其新增 VPC 組態的滲透測試，然後選擇修改滲透測試詳細資訊
10. 更新滲透測試以使用共用 VPC 資源

設定 功能

為您的 Agent Spaces 啟用和設定滲透測試、程式碼檢閱和威脅建模，包括修復、S3 來源和目標網域。

啟用滲透測試

設定 AWS Security Agent 在您的應用程式上執行自動滲透測試。此設定可讓 AWS Security Agent 存取您的 AWS 資源、驗證網域擁有權，以及執行全面的安全測試，以識別 Web 應用程式和 APIs漏洞。

啟用滲透測試可讓 AWS Security Agent 持續驗證您的應用程式安全性，而不會延遲手動測試。透過設定必要的 AWS 整合，您可以確保 AWS Security Agent 可以測試公有和私有應用程式、將問題清單記錄到 CloudWatch，以及存取憑證以進行已驗證的測試。

在此程序中，您將設定目標網域、選擇性設定 VPC、CloudWatch 記錄、登入資料儲存和 Lambda 函數進行測試、設定 S3 整合以提供其他內容，以及透過 IAM 角色設定服務存取。

先決條件

開始前，請確保您具備以下條件：

- 具有建立 IAM 角色管理許可的 AWS 帳戶
- 網域擁有權驗證功能 (DNS 或 HTTP 記錄修改）
- 要測試的目標網域或應用程式
- （選用）如果測試私有應用程式，則為 VPC 組態詳細資訊
- （選用）如果提供額外的成品給 AWS Security Agent，則為 S3 儲存貯體

步驟 1：設定網域

在精靈的第一個步驟中，指定您要測試的目標網域，以及 AWS Security Agent 應如何驗證擁有權。

1. 在目標網域區段中，在網域欄位中輸入您的網域。
2. 選取驗證方法：
   - DNS_TXT – 將 TXT 記錄新增至網域的 DNS 組態，以驗證網域擁有權。
   - HTTP_ROUTE – 透過在網域上的特定 URL 託管驗證檔案來驗證網域擁有權。
   - PRIVATE_VPC - 僅適用於私有 VPC 滲透測試。驗證網域是否解析為私有 CIDR 範圍內的 IP

- 如需詳細資訊，請參閱the section called “啟用用於滲透測試的應用程式網域”。

3. 選擇新增另一個網域以新增其他網域（最多總共 5 個）。
4. 選擇下一步以繼續網域驗證。

步驟 2：驗證網域

在精靈的第二個步驟中，驗證您設定的每個網域的擁有權。AWS Security Agent 需要經過驗證的擁有權，才能執行滲透測試。

1. 檢閱目標網域資料表中列出的網域。
2. 針對每個網域，選取它，並根據您選擇的方法觸發驗證：
   - Route 53 網域（相同的 AWS 帳戶）：選擇一鍵式驗證。AWS Security Agent 會自動建立 DNS 記錄並完成驗證。
   - DNS TXT（其他 DNS 供應商）：複製驗證字符，使用 DNS 註冊商新增 TXT 記錄，然後選取網域，然後選擇驗證。
   - HTTP 路由：將驗證字符放在 Web 伺服器上所需的路由路徑，然後選取網域，然後選擇驗證。如需詳細資訊，請參閱the section called “啟用用於滲透測試的應用程式網域”。
3. 選擇下一步以繼續進行選用組態。

> **Note**
>
> 使用 DNS TXT 或 HTTP 路由驗證時，已驗證網域的子網域不需要個別驗證。對於私有 VPC 驗證，目標端點網域名稱必須符合設定用於驗證的完整網域名稱。

> **Note**
>
> 對於 VPC 內的私有網域，即使網域驗證狀態為不可存取，您也可以繼續。AWS Security Agent 將嘗試在每次五項測試執行開始時驗證私有端點的網域。

步驟 3：（選用）設定其他功能

精靈的第三個步驟可讓您設定選用的 AWS 資源，以擴展滲透測試範圍。此步驟中的所有區段都是選用的，預設為收合。

（選用）設定 VPC 設定

如果您打算測試 VPC 中託管的私有目標網域，請設定 AWS Security Agent 的 VPC 設定。本節是選用的，預設為收合。

1. 展開 VPCs區段。
2. 在 VPC 下拉式清單中，選取託管私有目標網域的 VPC。
3. 在子網路下拉式清單中，選取 AWS Security Agent 應使用的 VPC 子網路：

> **Tip**
>
> 若要取得高可用性，請從多個可用區域選取多個子網路。確保您的子網路包含用於傳出連線的 NAT 閘道。

4. 在安全群組下拉式清單中，選取 AWS Security Agent 應使用的 VPC 安全群組：

> **Important**
>
> 確保您的安全群組允許 AWS Security Agent 的傳出連線執行滲透測試。

（選用）設定 CloudWatch 記錄

設定 CloudWatch 以存放滲透測試執行的日誌。本節是選用的，預設為收合。

1. 展開 CloudWatch 日誌區段。
2. 在日誌群組下拉式清單中，選取現有的 CloudWatch 日誌群組
3. 如果未選取任何日誌群組，AWS Security Agent 會建立名為 `/aws/securityagent/<agent name>/<pentest id>`且具有適當許可的日誌群組。

> **Note**
>
> 確保您的 IAM 角色具有寫入所選 CloudWatch 日誌群組的許可。

（選用）設定測試登入資料的秘密

如果您的應用程式需要身分驗證登入資料進行測試，AWS Security Agent 可以從 AWS Secrets Manager 安全地擷取登入資料。本節是選用的，預設為收合。

1. 展開秘密區段。
2. 選取包含應用程式登入資料的 AWS Secrets Manager 秘密。
3. 在 Web 應用程式中設定滲透測試時，您可以參考這些秘密以進行已驗證的測試。

> **Important**
>
> 登入資料會加密並儲存在 AWS Secrets Manager 中。確保您的 IAM 角色具有存取 Secrets Manager for AWS Security Agent 在測試期間使用這些登入資料的許可。

（選用）設定測試登入資料的 Lambda 函數

設定可在測試期間為您的應用程式提供登入資料的 Lambda 函數。本節是選用的，預設為收合。

1. 展開 Lambda 函數區段。
2. 選取可為您的應用程式提供身分驗證憑證的 Lambda 函數。
3. AWS Security Agent 會在滲透測試期間調用這些函數，以動態取得登入資料。

> **Note**
>
> 確保您的 IAM 角色具有叫用指定 Lambda 函數的許可。Lambda 函數應以預期的格式傳回登入資料，以供 AWS Security Agent 在測試期間使用。

（選用）設定 S3 儲存貯體

如果您打算上傳文件或成品以做為 AWS Security Agent 的輸入，請提供 S3 儲存貯體詳細資訊。本節是選用的，預設為收合。

1. 展開 S3 儲存貯體區段。
2. 在儲存貯體欄位中，輸入或搜尋您的 S3 儲存貯體名稱。

> **Note**
>
> 您也可以稍後連接 GitHub 儲存庫，或直接在 Web 應用程式中上傳檔案。提供的資訊可確保涵蓋範圍徹底、減少誤報，以及提供可行的結果。

（選用）設定服務存取

AWS Security Agent 需要 IAM 角色才能存取您的 AWS 資源 (VPC、CloudWatch 日誌群組、Secrets Manager、Lambda 函數等）以進行滲透測試。本節是選用的，預設為收合。

1. 展開服務存取區段。
2. 根據預設，AWS Security Agent 會使用具有滲透測試所需許可的預設 IAM 角色。
3. 若要自訂 IAM 角色，請選取下列其中一個選項：
   a. 建立預設角色 – AWS Security Agent 會自動建立具有必要許可的新 IAM 角色
   b. 使用現有的服務角色 – 從下拉式選單中選取現有的 IAM 角色
4. 如果使用現有角色：
   a. 選擇現有角色下的下拉式選單
   b. 從清單中選擇您的 IAM 角色
   c. 視需要選擇重新整理圖示以更新清單

> **Note**
>
> 預設 IAM 角色包含存取 VPC 資源、CloudWatch 日誌群組、Secrets Manager、Lambda 函數和其他滲透測試所需服務的許可。建議您使用預設 IAM 角色，除非您有特定的安全需求。

步驟 4：儲存並啟用滲透測試

設定所有必要設定後，為您的 AWS Security Agent 代理程式啟用滲透測試。

1. 檢閱所有組態區段以確保準確性。
2. 選擇儲存。
3. AWS Security Agent 會驗證您的組態，並建立必要的 AWS 資源。

後續步驟

啟用滲透測試之後：

- 在 AWS Security Agent Web 應用程式中設定滲透測試範圍
- 設定測試問題清單的通知偏好設定
- 檢閱並回應發現滲透測試結果的問題
- （選用）為問題清單修復設定其他儲存庫

如需執行和管理滲透測試的詳細資訊，請參閱 AWS Security Agent Web 應用程式文件。

啟用 GitHub 儲存庫的提取請求程式碼檢閱

提取請求程式碼檢閱現在已設定為程式碼檢閱設定精靈的一部分。當您將 GitHub 儲存庫連線至 Agent Space 時，您可以啟用個別儲存庫的提取請求註解，讓 AWS Security Agent 自動分析提取請求並發佈安全調查結果。

如需完整的設定說明，請參閱 the section called “啟用程式碼檢閱”。

如需提取請求調查結果如何在 GitHub 中顯示以及如何回應它們的資訊，請參閱 the section called “檢閱提取請求中的程式碼安全性問題清單”。

啟用程式碼檢閱

透過從 GitHub 儲存庫或 S3 儲存貯體連接原始程式碼，設定您的 Agent Space 以啟用程式碼檢閱。程式碼檢閱會分析您的原始程式碼是否有安全漏洞，以及是否符合組織的自訂安全需求。

設定程式碼檢閱組態是整個客服人員空間的操作。您連線的整合和 S3 儲存貯體會跨 功能共用，包括程式碼檢閱和滲透測試。

完成設定後，使用者可以在 AWS Security Agent Web 應用程式中建立和執行程式碼檢閱，以掃描儲存庫和 S3 來源是否有安全問題。

> **Note**
>
> 如果您已將 GitHub 儲存庫連接到您的客服人員空間（例如，透過滲透測試設定），則程式碼檢閱已啟用。您可以略過此設定，並直接前往 Web 應用程式來建立程式碼檢閱。請參閱 the section called “建立程式碼檢閱”。

先決條件

開始前，請確保您具備以下條件：

- 在 AWS 管理主控台中建立的客服人員空間（請參閱 the section called “建立 代理程式空間”）
- 至少下列其中一個來源碼輸入：
  - 已安裝 AWS Security Agent GitHub 應用程式的 GitHub 組織或使用者帳戶（請參閱 the section called “將 AWS 安全代理程式連線至 GitHub 儲存庫”）
  - 包含您要檢閱之來源碼的 S3 儲存貯體
- 為您的客服人員空間設定整合的許可
- （選用）如果您計劃使用安全需求驗證，則在安全需求套件中至少啟用一個安全需求（請參閱 the section called “管理安全需求”）

存取程式碼檢閱設定精靈

導覽至代理程式空間的程式碼檢閱設定。

1. 在 AWS Security Agent 主控台中，選取您的 Agent Space。
2. 從下列其中一個位置選擇啟用程式碼檢閱：
   - 程式碼檢閱卡片
   - 程式碼檢閱索引標籤，然後選擇啟用程式碼檢閱

> **Tip**
>
> 如果您希望 AWS Security Agent 自動處理程式碼檢閱意見回饋，也請啟用程式碼修復。如需詳細資訊，請參閱 the section called “讓使用者能夠開始修復滲透測試和程式碼檢閱問題清單”。

系統會將您導向至設定程式碼檢閱組態精靈。

步驟 1：連接整合、儲存庫和儲存貯體

在精靈的第一個步驟中，連接您的原始程式碼輸入並設定程式碼檢閱設定。您必須新增至少一個 GitHub 儲存庫或 S3 儲存貯體才能繼續。

> **Important**
>
> 此處設定的整合和 S3 儲存貯體會跨您的客服人員空間共用。變更同時適用於程式碼檢閱和滲透測試功能。

連接 GitHub 儲存庫

從授權的 GitHub 組織或使用者帳戶新增 GitHub 儲存庫。選擇新增會開啟兩步驟的 Connect GitHub 精靈，首先選取儲存庫，然後設定 AWS Security Agent 可以對每個儲存庫採取的動作。

1. 在連線整合區段中，選擇新增。
2. 選取包含您要檢閱之儲存庫的 GitHub 註冊。
3. 在 Connect GitHub 儲存庫步驟中，選取您要連線之每個儲存庫的核取方塊。
4. 選擇下一步以前往管理功能。

> **Note**
>
> 在程式碼檢閱和滲透測試期間，連線的儲存庫會唯讀存取以進行程式碼分析。您可以設定寫入動作，例如在下一個步驟張貼評論和開啟修補提取請求。

> **Note**
>
> 如果您尚未註冊 GitHub 整合，請選擇設定以導覽至整合頁面，您可以在其中授權 AWS Security Agent GitHub 應用程式。如需詳細資訊，請參閱the section called “將 AWS 安全代理程式連線至 GitHub 儲存庫”。

設定 GitHub 儲存庫功能

在 Connect GitHub 精靈的管理功能步驟中，選擇 AWS Security Agent 在每個儲存庫中可以執行的動作。每個儲存庫都會獨立設定程式碼檢閱註解和程式碼修復。

1. 對於每個儲存庫，在 上切換程式碼檢閱評論，讓 AWS Security Agent 將安全調查結果發佈為提取請求的評論。

2. 對於每個儲存庫，在 上切換程式碼修復，讓 AWS Security Agent 修正問題清單。啟用後，Web 應用程式使用者可以請求提取請求以修正問題清單，而提取請求作者可以評論`@AWS-Security-Agent fix all findings.`讓代理程式處理其評論。
3. 選擇儲存以套用您的選擇，並返回程式碼檢閱設定精靈。

為儲存庫啟用程式碼檢閱註解時：

- 當提取請求標記為「準備檢閱」時，AWS Security Agent 會自動分析提取請求。草稿提取請求不會進行分析。
- 安全調查結果會直接發佈為提取請求上的評論，並附有特定的修補指導。
- 分析會使用您設定的程式碼檢閱設定（安全漏洞、自訂需求或兩者）。

> **Note**
>
> 提取請求註解僅適用於私有 GitHub 儲存庫。

為儲存庫啟用程式碼修復時，Web 應用程式使用者可以開始修復該儲存庫上的程式碼檢閱和滲透測試結果，而 AWS Security Agent 會將每個修正做為提取請求交付。如需詳細資訊，請參閱the section called “讓使用者能夠開始修復滲透測試和程式碼檢閱問題清單”。

如需提取請求調查結果如何在 GitHub 中顯示以及如何回應它們的詳細資訊，請參閱 the section called “檢閱提取請求中的程式碼安全性問題清單”。

新增 S3 儲存貯體

新增包含來源碼或內容資源的 S3 儲存貯體以進行程式碼檢閱。

1. 在 S3 儲存貯體區段中，選擇新增 S3 資源。
2. 輸入包含原始碼的儲存貯體或字首的 S3 URI。
3. 選擇新增。

> **Note**
>
> 您最多可以新增 10 個 S3 資源。S3 儲存貯體會跨 功能共用，包括程式碼檢閱和滲透測試。

> **Tip**
>
> 您可以新增包含原始碼、組態檔案、infrastructure-as-code範本或您希望 AWS Security Agent 分析安全問題的其他成品的 S3 儲存貯體。

設定程式碼檢閱設定

設定 AWS Security Agent 在程式碼檢閱期間分析的安全問題類型。此設定適用於在此客服人員空間中啟用程式碼檢閱的所有儲存庫和來源。

1. 在程式碼檢閱設定區段中，選取下列其中一個選項：
   - 安全需求驗證 – 驗證程式碼是否符合您在安全需求套件中啟用的安全需求。
   - 安全漏洞調查結果 – 識別程式碼中的常見安全漏洞。
   - 安全需求和漏洞調查結果 – 分析程式碼是否符合您已啟用的安全需求和常見的安全漏洞。這是預設設定。

> **Note**
>
> 啟用安全需求驗證時，AWS Security Agent 會根據您在安全需求套件中啟用的安全需求檢查程式碼。如果您選取安全需求驗證，但未啟用至少一個安全需求，AWS Security Agent 將不會識別需求型問題清單。如需安全需求的詳細資訊，請參閱「the section called “管理安全需求”」。

1. 選擇下一步以繼續進行選用組態。

步驟 2：選用組態

在精靈的第二個步驟中，為您的程式碼檢閱環境設定選用的 CloudWatch 記錄和服務存取設定。

（選用）設定 CloudWatch 日誌

設定 CloudWatch 日誌群組，以在程式碼檢閱期間擷取和分析應用程式行為。

1. 展開 CloudWatch 日誌區段。
2. 在日誌群組下拉式清單中，從您的 AWS 帳戶選取一或多個現有的 CloudWatch 日誌群組。

> **Note**
>
> 如果您未選取日誌群組，AWS Security Agent 會自動建立預設日誌群組來存放程式碼檢閱執行日誌。

（選用）設定服務存取

設定 AWS Security Agent 用來存取您的 AWS 資源的 IAM 服務角色，例如 S3 儲存貯體和 CloudWatch 日誌以進行程式碼檢閱。

1. 展開服務存取區段。
2. 選取以下其中一個選項：
   - 建立預設角色 – AWS Security Agent 會自動建立具有程式碼檢閱必要許可的新 IAM 角色。
   - 使用現有的服務角色 – 從下拉式功能表中選取現有的 IAM 角色。
3. 如果使用預設角色，請輸入服務角色名稱。名稱在帳戶中的所有角色中必須是唯一的，並使用英數字元和+=,.@-_字元，且不能包含空格。

> **Note**
>
> 服務角色名稱的長度上限為 64 個字元。如果您未選取現有角色，系統會自動建立服務角色。

1. 選擇儲存以完成設定。

設定後

完成程式碼檢閱設定精靈之後：

- 客服人員空間頁面上的程式碼檢閱卡片會顯示就緒狀態。
- 使用者可以啟動 Web 應用程式並建立程式碼檢閱，以掃描連線的儲存庫和 S3 來源。
- 您可以隨時修改組態，方法是在程式碼檢閱索引標籤上選擇編輯組態。

編輯程式碼檢閱組態

若要在初始設定後修改程式碼檢閱組態：

1. 在 AWS Security Agent 主控台中導覽至您的 Agent Space。
2. 選取程式碼檢閱索引標籤。
3. 選擇 Edit Configuration (編輯組態)。
4. 視需要更新您的整合、S3 儲存貯體、程式碼檢閱設定、CloudWatch 日誌或服務存取。
5. 選擇儲存。

後續步驟

設定程式碼檢閱組態之後：

- 啟動 Web 應用程式以建立和執行程式碼檢閱 （ 請參閱 the section called “建立程式碼檢閱”)
- 隨著程式碼庫的成長，連接其他 GitHub 儲存庫或 S3 儲存貯體
- 設定組織特定驗證的安全需求套件 （ 請參閱 the section called “管理安全需求”)
- 檢閱提取請求調查結果在 GitHub 中的顯示方式 （ 請參閱 the section called “檢閱提取請求中的程式碼安全性問題清單”)

啟用威脅建模

透過連接原始程式碼儲存庫和設定 AWS 資源，設定您的 Agent Space 以啟用威脅建模。威脅建模會分析應用程式的架構，並從原始程式碼、設計文件或兩者中識別安全威脅。

設定威脅建模組態是整個客服人員空間的操作。您連線的整合和 S3 儲存貯體會跨功能共用，包括威脅建模、程式碼檢閱和滲透測試。

完成設定後，使用者可以在 AWS Security Agent Web 應用程式中建立和執行威脅模型。

> **Note**
>
> 如果您已將儲存庫或 S3 儲存貯體連接至您的 Agent Space （ 例如，透過程式碼檢閱或滲透測試設定 ），則威脅建模可能已啟用。您可以直接前往 Web 應用程式來建立威脅模型。請參閱 the section called “建立威脅模型”。

先決條件

開始前，請確保您具備以下條件：

- 在 AWS 管理主控台中建立的客服人員空間（請參閱 the section called “建立 代理程式空間”）
- 為您的客服人員空間設定整合的許可
- （選用）安裝 AWS Security Agent 應用程式的 GitHub、GitLab 或 Bitbucket 組織（請參閱將 AWS Security Agent 連線至 GitHub 儲存庫、將 AWS Security Agent 連線至 GitLab 儲存庫，或將 AWS Security Agent 連線至 Bitbucket 儲存庫）

存取威脅建模設定精靈

導覽至 Agent Space 的威脅建模組態。

1. 在 AWS Security Agent 主控台中，選取您的 Agent Space。
2. 從威脅模型卡或從威脅模型索引標籤中選擇設定威脅模型。

系統會將您導向至設定威脅模型精靈，其中有兩個選用步驟。

步驟 1：連接原始程式碼儲存庫（選用）

連接您要啟用威脅建模的 GitHub、GitLab 或 Bitbucket 儲存庫。威脅模型本身會在 Web 應用程式中建立和檢視。

> **Important**
>
> 此處設定的整合會跨您的客服人員空間共用。變更適用於威脅建模、程式碼檢閱和滲透測試功能。

連接儲存庫

1. 在連線整合區段中，選擇新增。
2. 選取包含您要使用之儲存庫的註冊。
3. 選取您要連線之每個儲存庫的核取方塊。
4. 選擇儲存以套用您的選擇。

> **Note**
>
> 如果您尚未註冊整合，請選擇設定以導覽至整合頁面，您可以在其中授權 AWS Security Agent 應用程式。如需詳細資訊，請參閱將 AWS 安全代理程式連線至 GitHub 儲存庫、將 AWS 安全代理程式連線至 GitLab 儲存庫，或將 AWS 安全代理程式連線至 Bitbucket 儲存庫。

1. 選擇下一步以繼續進行選用組態。

步驟 2：選用組態

為您的威脅建模環境設定 S3 儲存貯體、CloudWatch 記錄和服務存取設定。這些設定會與 Agent Space 上的其他功能共用。

S3 儲存貯體 （選用）

新增 S3 儲存貯體，其中包含您希望代理程式在威脅建模期間用作內容的原始碼。

1. 在 S3 儲存貯體區段中，選擇新增 S3 資源。
2. 輸入儲存貯體或字首的 S3 URI。
3. 選擇新增。

> **Note**
>
> 您最多可以新增 10 個 S3 資源。

CloudWatch 日誌 （選用）

設定 CloudWatch 日誌群組，以擷取和分析威脅模型執行期間的應用程式行為。

1. 在 CloudWatch 日誌區段中，從您的 AWS 帳戶選取一或多個現有的 CloudWatch 日誌群組。

服務存取

設定 AWS Security Agent 用來存取 AWS 資源的 IAM 服務角色，例如 S3 儲存貯體和 CloudWatch 日誌以進行威脅建模。需要服務角色才能啟用威脅建模。

1. 在服務存取區段中，從下拉式清單中選取現有的 IAM 服務角色，或將其保留空白，以自動建立服務角色。

> **Note**
>
> 如果您未選取現有角色，系統會自動建立服務角色。

1. 選擇儲存以完成組態。

設定後

完成威脅建模組態之後：

- 客服人員空間頁面上的威脅模型卡會顯示就緒狀態。
- 使用者可以啟動 Web 應用程式，並從連線的原始碼、上傳的範圍文件、Confluence 頁面或這些輸入的組合建立威脅模型。

後續步驟

啟用威脅建模之後：

- 啟動 Web 應用程式以建立和執行威脅模型（請參閱建立威脅模型）
- 隨著程式碼庫的成長，連接其他儲存庫或 S3 儲存貯體
- 連接 Confluence 以啟用選取頁面做為範圍文件（請參閱將 AWS 安全代理程式連接到 Confluence）

讓使用者能夠開始修復滲透測試和程式碼檢閱問題清單

在 AWS 管理主控台中，您可以啟用自動修復，以便 AWS Security Agent Web 應用程式的使用者可以請求修正特定問題清單。AWS Security Agent 會將每個修正交付為受影響儲存庫上的 GitHub 提取請求。

您可以從 代理程式空間內啟用每個儲存庫的修復。滲透測試和程式碼檢閱索引標籤會開啟相同的儲存庫組態精靈，因此您可以使用任一索引標籤來管理啟用自動修復的設定。修補適用於兩個功能的問題清單，因此每個儲存庫只需要啟用一次。

先決條件

開始前，請確保您具備以下條件：

1. 已啟用滲透測試（請參閱 the section called “啟用滲透測試”) 或程式碼檢閱（請參閱 the section called “啟用程式碼檢閱”)
2. 為您的 GitHub 組織安裝並授權 AWS Security Agent GitHub 應用程式（請參閱 the section called “將 AWS 安全代理程式連線至 GitHub 儲存庫”)

開啟儲存庫組態精靈

選擇符合您儲存庫設定方式的進入點。

從滲透測試索引標籤

使用此路徑來新增 GitHub 儲存庫以進行滲透測試，並在相同的傳遞中設定修復。

1. 導覽至客服人員空間概觀頁面。
2. 選擇滲透測試索引標籤。
3. 選取擁有您儲存庫的 GitHub 註冊：
   a. 如果您尚未將任何 GitHub 註冊與客服人員空間建立關聯，請在將 GitHub 連接至 AWS 安全客服人員資訊方塊中選擇新增以選取註冊。
   b. 如果一或多個 GitHub 註冊已關聯，請在連線整合區段中選擇新增以選取另一個。
4. 選擇下一步以選擇 GitHub 儲存庫。
5. 選擇下一步以設定儲存庫功能。

從程式碼檢閱索引標籤

當您的儲存庫已連線進行程式碼檢閱時，請使用此路徑。

1. 導覽至客服人員空間概觀頁面。
2. 選擇程式碼檢閱索引標籤。
3. 在 GitHub 儲存庫資料表中，選擇編輯以開啟儲存庫組態精靈。

啟用自動修復並儲存

到達儲存庫功能後，請逐步完成上述任一路徑：

1. 在啟用自動修復的欄中，將您要修復的每個儲存庫標記為已啟用。
2. 選擇連線以儲存組態。

Web 應用程式的使用者現在可以開始修復這些儲存庫上的問題清單，AWS Security Agent 將使用建議的修正來開啟提取請求。

從 S3 儲存貯體提供代理程式資源

您可以授予 AWS Security Agent 對 S3 儲存貯體中資源的存取權，例如支援滲透測試的 API 文件、威脅模型文件和原始碼。

您可以在 AWS 管理主控台中授予 AWS Security Agent 對 S3 儲存貯體的讀取存取權。在 Security Agent Web 應用程式中，使用者從 S3 選取相關的個別資源。

1. 導覽至客服人員空間概觀頁面
2. 選取動作，然後編輯滲透測試組態
3. 在 S3 儲存貯體區段下，定義包含 S3 儲存貯體的 S3 URI。 S3 您可以新增多個 S3 儲存貯體。

啟用用於滲透測試的應用程式網域

您需要新增目標網域並驗證擁有權，才能在應用程式上執行滲透測試。AWS Security Agent 只會針對已驗證的網域執行滲透測試。

> **Note**
>
> 您不需要驗證應用程式可能使用的輔助網域。您只需要驗證您將主動執行滲透測試的網域。

步驟 1：新增目標網域

若要新增目標網域，請導覽至客服人員空間概觀頁面上的滲透測試索引標籤。根據您是否已設定滲透測試，請使用下列其中一種方法：

- 第一次設定：選擇設定滲透測試以開啟滲透測試精靈。在第一個步驟中，輸入網域並選擇驗證方法。
- 將網域新增至現有組態：在目標網域表格中，選擇新增網域。輸入網域，然後在模態中選擇驗證方法。

您可以新增基本網域或子網域，例如 example.com或 billing.example.com。AWS 建議您使用具有建立TXT記錄許可的子網域。

> **Note**
>
> 當您使用 DNS TXT 或 HTTP 路由驗證來驗證網域時，會自動涵蓋該網域的子網域。例如，驗證example.com可讓您測試 api.example.com或 ，billing.example.com無需額外的驗證。對於私有 VPC 驗證，目標端點網域名稱必須符合設定用於驗證的完整網域名稱。

選擇下列其中一種驗證方法：

- DNS TXT 記錄：使用 DNS 供應商建立 DNS TXT 記錄來驗證網域擁有權。
- HTTP 路由：在您的 Web 伺服器上建立包含 AWS Security Agent 所提供唯一字符的路由，以驗證網域擁有權。
- 私有 VPC：僅適用於私有 VPC 滲透測試。驗證網域是否解析為私有 CIDR 範圍內的 IP。設定的網域名稱必須符合任何相關聯目標端點的完整網域名稱

步驟 2：驗證網域擁有權

新增網域之後，請使用您選取的方法驗證擁有權。您可以隨時從目標網域資料表觸發驗證，方法是選取網域，然後選擇驗證。

使用 DNS TXT 記錄進行驗證

AWS Security Agent 會產生 TXT DNS 記錄值。您必須向 DNS 供應商新增此記錄，才能證明擁有權。

如果網域已在 Route 53（相同的 AWS 帳戶）中註冊：

AWS Security Agent 可以自動建立 DNS 記錄。從目標網域資料表中選取網域，然後選擇一鍵式驗證。AWS Security Agent 會建立 DNS 驗證記錄，並自動完成驗證。

如果網域已向另一個 DNS 供應商註冊：

1. 複製 AWS Security Agent 提供的 TXT 記錄值。
2. 使用 DNS 註冊商新增 TXT 記錄。
3. 返回目標網域表格，選取網域，然後選擇驗證。

使用 HTTP 路由驗證進行驗證

此方法會在 Web 伺服器上放置唯一的字符來證明網域擁有權。只有網域擁有者或授權的 Web 管理員可以在 Web 伺服器上建立路由，以證明擁有權。

1. 在 Web 伺服器上的下列路徑建立檔案：

```
.well-known/aws/securityagent-domain-verification.json
```

2. 使用此格式，將 AWS Security Agent 提供的字符放入 檔案：

```
{
  "tokens": ["<insert-token>"]
}
```

3. 返回目標網域表格，選取網域，然後選擇驗證。AWS Security Agent 會將 HTTPS GET 請求傳送至驗證 URL，並驗證權杖。
4. 如果網域可在公有網際網路上存取，請確保您的網域在執行驗證之前擁有有效的 SSL 憑證。

> **Note**
>
> 如果您的網域已在多個代理程式空間中註冊，而且您使用 HTTP 路由驗證，則可以將兩個代理程式空間的字符放在同一個`tokens`陣列中。

略過網域擁有權驗證

有權在端點上執行滲透測試但無法完成所有權驗證的客戶可以向我們請求手動驗證。請開啟客戶支援案例以提出此請求。您的請求必須包含您的業務使用案例和手動所有權驗證的理由 （即您獲得授權在端點上執行滲透測試的原因）。

用於滲透測試的受管目標網域

在 AWS 管理主控台中，您可以新增和管理代理程式空間用於滲透測試的目標網域。這些目標網域必須先經過驗證，才能用於滲透測試。如需驗證目標網域的詳細資訊，請參閱 the section called “啟用滲透測試”

先決條件

開始前，請確保您具備以下條件：

1. 已啟用滲透測試（請參閱 the section called “啟用滲透測試”）

管理目標網域資源

- 導覽至目標網域概觀頁面。
- 您應該會看到與您的帳戶相關聯的所有目標網域資源
  - 如果您沒有看到與您的帳戶相關聯的任何目標網域，請依照 中的步驟the section called “啟用滲透測試”建立和驗證目標網域
- 可以在客服人員空間和共用驗證狀態之間重複使用目標網域
  - 若要將現有目標網域新增至客服人員空間，請導覽至客服人員空間的滲透測試索引標籤。選取新增網域，然後在網域名稱欄位中從先前註冊的可用網域中選擇所需的網域
  - 目標網域必須與代理程式空間相關聯，才能用於滲透測試
- 從代理程式空間移除網域並不會刪除網域。關聯的目標網域可以從目標網域概觀頁面永久刪除

驗證目標網域

若要在滲透測試中使用目標網域，必須先使用下列其中一種方法進行驗證：

- Route 53 網域（相同的 AWS 帳戶）：選擇一鍵式驗證。AWS Security Agent 會自動建立 DNS 記錄並完成驗證。
- DNS TXT（其他 DNS 供應商）：複製驗證字符，使用 DNS 註冊商新增 TXT 記錄，然後選取網域，然後選擇驗證。
- HTTP 路由：將驗證字符放在 Web 伺服器上所需的路由路徑，然後選取網域，然後選擇驗證。如需詳細資訊，請參閱the section called “啟用用於滲透測試的應用程式網域”。
- 私有 VPC：驗證目標網域的 IP 落在私有 CIDR 範圍內 (the section called “將代理程式連線至私有 VPC 資源”如需私有 CIDR 範圍的清單，請參閱)。滲透測試目標端點網域名稱必須符合設定用於驗證的完整網域名稱。僅適用於私有 VPC 滲透測試

> **Note**
>
> 對於 DNS TXT、HTTP 路由和 Route 53 驗證，驗證網域的子網域會自動涵蓋，不需要個別驗證。對於私有 VPC 驗證，每個網域都必須個別驗證。

管理安全需求

使用從您自己的文件匯入的 AWS 受管套件、自訂套件或需求，定義 AWS Security Agent 在設計和程式碼檢閱期間評估的安全需求。

管理安全需求

將 AWS Security Agent 用來分析應用程式的安全需求整理為安全需求套件。套件是安全要求的集合。您可以啟用 AWS受管套件、建立自己的自訂套件，並透過上傳安全文件來產生自訂套件的需求。

概觀

安全需求定義了安全標準或政策， AWS Security Agent 會在設計審查和程式碼審查期間評估您的應用程式。當設計或程式碼不符合要求時， AWS Security Agent 會報告問題清單。每個安全需求都屬於安全需求套件。安全需求會跨所有 Agent Spaces 共用，並在設計和程式碼檢閱期間進行評估。

AWS Security Agent 提供兩種類型的套件，顯示在不同的索引標籤上：

- 受管安全需求套件 AWS– 根據產業標準和最佳實務提供的安全需求套件。您可以立即啟用這些套件，以針對沒有自訂組態的常見安全政策進行評估。受管套件中的要求為唯讀。您可以使用 AWS受管需求做為自訂需求的範本。如需可用 受管套件的清單，請參閱 AWS 受管安全需求套件。
- 自訂安全需求套件 – 您為強制執行組織的特定政策和標準而建立的套件。您可以從頭開始在自訂套件中建置需求、以自訂 AWS受管需求做為起點，或上傳您的安全文件來產生需求。

> **Note**
>
> 合規架構套件旨在協助識別可能與特定架構合規相關的安全要求。他們不會評估您的合規，也不會保證您將通過稽核。

您可以一併啟用和停用套件。啟用套件時， AWS Security Agent 會在設計和程式碼檢閱期間，根據該套件中的要求評估您的應用程式。

> **Note**
>
> 啟用或停用套件適用於新的設計檢閱和程式碼檢閱。現有的檢閱不會受到影響。

啟用或停用受管套件

啟用 AWS受管套件，根據一組 AWS維護的安全需求來評估您的應用程式。當您不再需要它時停用它。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇受管安全需求套件索引標籤。
4. 選取您要啟用或停用的套件。
5. 執行以下任意一項：
   a. 若要啟用套件，請選擇啟用套件。
   b. 若要停用套件，請選擇停用套件。

> **Tip**
>
> 選擇套件以檢視其包含的安全需求。選擇要求以檢視其完整定義，包括其適用性、合規條件和修補指引。

建立自訂套件

建立自訂套件，將組織特定的安全需求分組。建立套件後，請手動新增需求、自訂 AWS受管需求，或上傳文件以產生需求。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤。
4. 選擇建立安全需求套件。
5. 輸入安全需求套件名稱和選用描述。
6. 選擇建立安全需求套件。

> **Note**
>
> 建立套件後，您可以新增或上傳來源文件，以產生套件的安全需求。

手動新增安全需求

將安全性需求新增至自訂套件，以定義 AWS Security Agent 強制執行的安全標準。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤，然後選擇您要新增需求的套件。
4. 選擇手動新增需求。
5. 設定下列欄位：
   a. 安全需求名稱 – 輸入識別安全控制的描述性名稱，例如 `enforce-encryption-at-rest`（最多 80 個字元）。
   b. 描述 – 提供此安全需求檢查項目的簡短描述（最多 500 個字元）。
   c. 適用性 – 描述應評估此安全需求的案例、系統類型或條件。包含應標記為 NOT_APPLICABLE（最多 10，000 個字元）的案例。
   d. 合規條件 – 定義什麼構成此安全需求的合規與不合規。提供 AWS Security Agent 在評估合規時所尋找的指標、範例和技術詳細資訊（最多 10，000 個字元）。
   e. 修補指引（選用）– 提供如何修正違規的指引，包括組織內部文件或標準的連結（最多 10，000 個字元）。
6. 執行以下任意一項：
   a. 若要建立需求而不啟用，請選擇建立安全需求。
   b. 若要建立需求並啟用，請選擇建立並啟用安全需求。

> **Note**
>
> 只有在啟用包含它的套件時，才會在安全性審查期間評估需求。若要控制是否評估套件，請啟用或停用套件。

自訂 AWS受管安全需求

建立使用 AWS受管需求做為起點的自訂安全需求。 AWS 安全代理程式會從受管需求預先填入需求詳細資訊，然後您編輯它們以符合您的組織。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。

3. 選擇自訂安全需求套件索引標籤，然後選擇要新增需求的自訂套件。
4. 選擇建立自訂安全需求。
5. 若要預先填入表單，請在自訂 AWS受管安全需求 區段中，搜尋並選取要用作範本的 AWS受管需求。
6. 編輯任何欄位以符合組織的需求。
7. 執行以下任意一項：
   a. 若要建立需求而不啟用，請選擇建立安全需求。
   b. 若要建立並立即啟用需求，請選擇建立並啟用安全需求。

> **Note**
>
> 自訂 AWS受管需求會建立獨立的自訂安全需求。受管需求的後續變更不會影響您 AWS的自訂版本。

透過上傳文件產生需求

從現有的安全文件中產生自訂套件的安全需求，而不是手動撰寫每個需求。 AWS 安全代理程式會讀取您上傳的文件、識別安全相關內容，並產生結構化需求，供您檢閱和編輯。如需完整程序，請參閱從文件產生安全需求。如需上傳文件中要包含哪些內容的指引，請參閱準備文件以產生需求。

> **Important**
>
> 上傳新的來源文件會重新產生套件的所有需求。取代任何現有的需求，包括您手動新增的要求。

編輯自訂安全需求

修改自訂安全需求，以更新其定義、條件或修補指引。您無法編輯 AWS受管套件中的要求。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤，然後選擇包含需求的套件。
4. 選取您要編輯的需求，然後選擇編輯自訂安全需求。

5. 視需要更新要求欄位。安全需求名稱無法在建立後變更。
6. 選擇更新安全需求。

> **Note**
>
> 安全性需求的變更適用於新的設計審查和程式碼審查。現有的檢閱不會受到影響。

啟用或停用套件

啟用或停用安全需求套件，以控制 AWS Security Agent 是否評估其包含的要求。您可以一併啟用和停用套件。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇套件類型的索引標籤，然後選取套件。
4. 執行以下任意一項：
   a. 若要啟用套件，請選擇啟用套件。
   b. 若要停用套件，請選擇停用套件。

> **Note**
>
> 啟用套件時，會在安全性審查期間評估套件中的要求。停用套件時，不會評估其需求。

刪除自訂安全需求

當您不再希望 Security Agent 對其進行評估時，請刪除自訂 AWS 安全需求。

1. 在 AWS 主控台中，導覽至 AWS Security Agent。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤，然後選擇包含需求的套件。
4. 選取一或多個安全需求，然後選擇刪除安全需求。
5. 確認刪除。

> **Note**
>
> 當您刪除安全要求時，系統不會再在未來的檢閱中評估這些要求。在過去的檢閱中，這些要求仍然可見為唯讀結果。

定義安全需求的最佳實務

當您建立安全需求時，請遵循這些準則，以協助 AWS Security Agent 準確評估您的應用程式並提供可行的問題清單。

安全需求名稱 – 使用明確、特定的名稱來識別安全控制。避免未傳達需求用途的一般術語。

描述 – 說明控制項檢查的內容及其減輕的風險。這有助於使用者了解為什麼需求很重要。

適用性 – 描述應評估需求的工作負載、系統類型或條件。說明何時應該在特定案例中標示 NOT_APPLICABLE，以避免誤報。「此控制項適用於「和「標記為 NOT_APPLICABLE if...」設定明確範圍界限的所有工作負載。

合規條件 – 將此分為兩個部分：哪些 顯示合規，以及哪些 表示不合規。具體說明技術指標並包含邊緣案例。從「如果設計示範...」開始，接著是技術詳細資訊，然後「如果是...」，則設計是不合規的，其模式表示違規。

修補指引 – 提供技術詳細資訊和組態範例的指引。包含組織內部文件的連結，讓開發人員擁有修正違規所需的資源。

後續步驟

設定安全需求套件之後：

- 建立設計檢閱或程式碼檢閱，以針對您啟用的套件評估您的應用程式。
- 啟用滲透測試，以補充您的設計和程式碼檢閱。
- 授予使用者存取 AWS Security Agent Web 應用程式的權限。

AWS 受管安全需求套件

AWS 受管安全需求套件是根據產業標準和最佳實務 AWS 建立和維護的一系列安全需求。您可以啟用受管套件，在設計檢閱和程式碼檢閱期間根據應用程式的需求進行評估，而無需自行撰寫需求。

受管套件中的安全需求為唯讀。您無法變更它們。您可以使用 AWS受管需求做為範本來建立自訂需求，然後編輯複本以符合您的組織。如需詳細資訊，請參閱管理安全需求。

AWS 會隨著時間更新受管套件。當 AWS 新增或修改受管套件中的需求時，變更會套用至每個已啟用套件的帳戶。

> **Note**
>
> 合規架構套件旨在協助識別可能與特定架構合規相關的安全要求。他們不會評估您的合規，也不會保證您將通過稽核。

AWS 受管套件：ASA 基本套件

適用於大多數工作負載的基準安全要求集。此套件涵蓋常見的應用程式安全問題，例如身分驗證和授權、資料保護、記錄和輸入驗證。啟用此套件為您的設計和程式碼檢閱建立安全基準。

AWS 受管套件：AWS Well-Architected Pack

衍生自 AWS Well-Architected Framework 安全支柱的安全需求。此套件會根據保護資料、管理身分和許可，以及偵測和回應安全事件 AWS 的指引來評估您的應用程式。如需架構的詳細資訊，請參閱安全支柱 - AWS Well-Architected Framework。

AWS 受管套件：NIST CSF Pack

衍生自 NIST 網路安全架構 (CSF) 的安全需求。此套件會根據從架構提取的控制區域來評估您的應用程式，例如身分和存取管理、資料安全和保護技術。啟用此套件，讓您的設計和程式碼檢閱與 NIST CSF 保持一致。

AWS 受管套件：PCI DSS Pack

衍生自支付卡產業資料安全標準 (PCI DSS) 的安全需求。此套件會根據與處理持卡人資料相關的控制區域來評估您的應用程式，例如存取控制、傳輸中和靜態資料的加密，以及記錄。啟用此套件，讓您的設計和程式碼檢閱與 PCI DSS 保持一致。

從文件產生安全需求

透過上傳現有的安全文件來產生自訂套件的安全需求。 AWS 安全代理程式會讀取您上傳的文件、識別安全相關內容，並產生具有適用性、合規條件和修補指引的結構化安全需求。您可以檢閱和編輯產生的需求，再將其用於檢閱。

當您已記錄安全政策、標準或指導方針時，請使用此選項，而不是手動撰寫每個要求。如需上傳文件中要包含哪些內容的指引，請參閱準備文件以產生需求。

支援檔案格式

您可以上傳下列檔案格式：

- DOC
- DOCX
- MD
- PDF
- TXT

產生需求

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤。
4. 建立套件，或選擇要產生需求的現有自訂套件。
5. 選擇選擇檔案，或將文件拖放到上傳區域。
6. 選擇產生需求。
7. 等待 AWS 安全代理程式處理文件。產生會在背景執行，大型檔案可能需要幾分鐘的時間。您無法在產生時開始對套件進行另一次上傳。
8. 檢閱產生的安全需求。視需要編輯、刪除或新增需求。當您希望 AWS 安全代理程式在安全審查期間評估其需求時，請啟用套件。

> **Note**
>
> AWS Security Agent 會在工作負載層級產生需求，並著重於文件中與安全相關的內容。產生的需求數量取決於您提供的內容。檢閱產生的要求，以確認它們反映您的意圖。

以新的上傳取代需求

上傳新的來源文件會重新產生套件的需求。

> **Important**
>
> 當您將新的來源文件上傳至套件時， AWS Security Agent 會重新產生該套件的所有需求。取代任何現有的需求，包括您手動新增的要求。若要保留您目前的需求，請勿上傳新文件。

1. 在 AWS 主控台中，導覽至 AWS 安全代理程式。
2. 在導覽窗格中，選擇安全需求。
3. 選擇自訂安全需求套件索引標籤，然後選擇套件。
4. 開始新的上傳，並確認上傳新的來源文件會取代現有的需求。
5. 選擇您的檔案，然後選擇產生需求。

後續步驟

- 檢閱並啟用產生的需求。請參閱管理安全需求。
- 建立設計檢閱或程式碼檢閱，以針對套件評估您的應用程式。

準備文件以產生需求

當您從文件產生安全需求時， AWS Security Agent 會讀取您的文件，並從其找到的安全相關內容產生結構化需求。您不需要以任何特定方式格式化文件，也不需要預先撰寫適用性、合規條件和修補指引。AWS Security Agent 會從您提供的內容衍生這些文件。上傳您已擁有的安全文件，以您自己的文字撰寫。

此頁面說明要包含的內容，以便 AWS Security Agent 產生準確、有用的需求。如需上傳程序和檔案限制，請參閱從文件產生安全需求。

要包含的內容

AWS Security Agent 會尋找描述您安全期望的內容。涵蓋下列主題的文件會產生最強大的要求：

- 存取控制和授權
- 身分驗證
- 資料保護和加密
- 網路安全
- 記錄和稽核

- 事件回應
- 漏洞和修補程式管理
- 適用於工作負載的合規義務

良好的來源文件包括安全政策、工程標準、控制目錄、架構準則和安全編碼標準。

在工作負載層級寫入

AWS Security Agent 會產生適用於個別工作負載或應用程式的需求，與在設計和程式碼檢閱期間評估的範圍相同。說明如何建置和操作安全工作負載的內容會產生需求。整個組織的控管主題，例如多帳戶策略、根使用者管理和帳戶層級記錄設定，都在此範圍之外，不會產生工作負載需求。

描述結果，而不是單一實作

說明您預期的安全性結果，而不是單一指定的產品或組態。 AWS Security Agent 會產生專注於所需安全功能的需求，並允許多個有效的實作。例如，與指定一個特定服務或設定的陳述式相比，「靜態資料已加密」會產生更清楚、更廣泛適用的需求。

具體說明什麼是好東西

文件描述預期行為的方式越具體， AWS 安全代理程式越能定義合規條件。可以的話，請描述合規設計示範什麼，以及什麼表示違規。相較於具體的工作負載相關陳述式，模糊或純理想陳述式產生的需求較少且較弱。

檢閱您得到的內容

每個產生的需求都包含名稱、適用性、合規條件，以及 AWS Security Agent 衍生自您文件的修補指引。檢閱產生的需求、編輯任何需要精簡的需求，並啟用您希望 AWS Security Agent 評估的需求。如需詳細資訊，請參閱管理安全需求。

管理使用者和存取

控制誰可以存取每個 Agent Space，並設定 AWS Security Agent 使用的 IAM 角色和加密金鑰。

授予使用者存取 AWS Security Agent Web 應用程式的權限

AWS Security Agent 提供兩種方法，讓使用者存取 Web 應用程式，視您在設定期間設定 Agent Space 的方式而定。

存取方法概觀

IAM Identity Center (SSO) - 如果您在建立代理程式空間時啟用了 IAM Identity Center，則使用者可以直接透過 SSO 存取 Web 應用程式。您可以透過 AWS Security Agent 主控台將使用者指派給 Agent Space，以及使用者使用其 Identity Center 登入資料登入。

管理員存取 - 具有 AWS 主控台存取權的使用者可以透過 AWS 管理主控台中客服人員空間概觀頁面上的管理員存取連結啟動 Web 應用程式。

使用 IAM Identity Center (SSO) 授予存取權

如果您已使用 IAM Identity Center 設定 Agent Space，您可以使用 AWS Security Agent 主控台將使用者指派給 Agent Space。

透過 AWS Security Agent 主控台指派使用者

1. 在 AWS 安全代理程式管理主控台中，導覽至您的代理程式空間。
2. 選取 Web 應用程式索引標籤。
3. 在使用者表格中，選擇新增使用者。
4. 從 IAM Identity Center 選取現有使用者或建立新使用者。
5. 確認使用者指派。

> **Tip**
>
> 指派給 Agent Space 的使用者可以使用其 SSO 憑證透過 IAM Identity Center 登入來存取 Web 應用程式。

使用 SSO 存取 Web 應用程式

將使用者指派給客服人員空間後：

1. 使用者導覽至 Agent Space 的 Web 應用程式 URL。

> **Tip**
>
> 選取複製 Web 應用程式 URL，在 AWS Security Agent 主控台的客服人員空間詳細資訊頁面上尋找 Web 應用程式 URL。使用者應將此 URL 加入書籤，以便輕鬆存取。

2. 使用者使用其 SSO 登入資料登入。
3. 身分驗證之後，使用者可以選取客服人員空間，並開始執行安全評估。

授予僅限 IAM 的存取權

如果您將 Agent Space 設定為僅限 IAM 存取，則具有 AWS 主控台存取的使用者可以透過管理員存取連結啟動 Web 應用程式。

使用管理員存取連結

1. 登入 AWS Security Agent 主控台。
2. 導覽至您要存取的客服人員空間。
3. 在客服人員空間登陸頁面的 Web 應用程式索引標籤上，選擇管理員存取按鈕以啟動 Web 應用程式。
4. Web 應用程式會在使用者自動驗證的新索引標籤中開啟。

> **Note**
>
> 管理員存取連結僅適用於已向具有適當 AWS 安全代理程式許可的 AWS 主控台進行身分驗證的使用者。此方法不需要 IAM Identity Center 組態。

後續步驟

授予使用者存取 Web 應用程式的權限之後：

- 使用者可以建立和管理滲透測試組態和執行
- 使用者可以建立和管理設計檢閱
- 使用者可以檢視安全性問題清單和修補指引
- 設定安全性問題清單的通知偏好設定

為 AWS 安全代理程式建立 IAM 角色

AWS Security Agent 以三種方式使用 IAM 角色：

1. 應用程式角色：建立 AWS Security Agent 應用程式時使用。對於 IAM Identity Center 和管理存取連結使用案例，服務會擔任此角色，以授予 WebApp 使用者與 AWS Security Agent APIs許可。
2. 滲透測試服務角色：在建立客服人員空間做為可用角色清單時指定。稍後，WebApp 使用者會在建立滲透測試時選取其中一個角色。AWS Security Agent 服務會擔任此角色，以在測試期間存取您的 AWS 資源。
3. 演員角色：用來驗證和授權對目標 Web 應用程式的請求 （例如 AWS API Gateway APIs)。這些角色會在客服人員空間建立期間提供。AWS Security Agent 代理程式會擔任演員角色來與您的目標應用程式互動。

應用程式角色

在 服務中建立 AWS Security Agent 應用程式時，會使用 應用程式角色。對於 IAM Identity Center 和管理存取連結身分驗證案例，AWS Security Agent 服務會擔任此角色，授予 WebApp 使用者與 AWS Security Agent APIs 互動的必要許可。

所需的許可

此角色需要下列許可：

- 叫用 AWS Security Agent API 操作
- 讀取和寫入應用程式組態資料
- 存取使用者工作階段資訊
- 管理 WebApp 使用者的身分驗證字符

信任政策

信任政策必須允許 AWS Security Agent 服務擔任此角色：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
```

```
  ]
}
```

許可政策

角色應包含下列項目的許可：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "securityagent:GetApplication",
        "securityagent:UpdateApplication",
        "securityagent:ListAgentInstances",
        "securityagent:CreatePentestSession"
      ],
      "Resource": "arn:aws:securityagent:*:*:application/*"
    }
  ]
}
```

> **Note**
>
> 根據您的特定應用程式需求和最低權限原則自訂許可。

滲透測試服務角色

將客服人員空間建立為可用角色清單時，會指定滲透測試服務角色。當 WebApp 使用者建立滲透測試時，他們會選取其中一個角色。然後，AWS Security Agent 服務會擔任此角色來存取和測試您的 AWS 資源。

所需的許可

此角色需要許可，才能在滲透測試期間存取和分析您的 AWS 資源：

- 讀取和描述 VPC 組態和網路拓撲
- 檢查 EC2 執行個體、安全群組和網路 ACLs

- 分析 IAM 政策和資源許可
- 讀取 CloudWatch 日誌和指標
- 存取與安全性測試相關的 AWS 服務組態

信任政策

信任政策必須允許 AWS Security Agent 服務擔任此角色以進行滲透測試操作：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

> **Note**
>
> 允許跨帳戶或服務存取時，請使用外部 ID 來提高安全性。

許可政策

此角色應包含對 AWS 資源的唯讀存取權。請考慮使用這些 受管政策：

- `SecurityAudit` - 用於安全稽核的 AWS 受管政策
- `ViewOnlyAccess` - 對大多數 AWS 服務的唯讀存取

或建立具有特定許可的自訂政策：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:Describe*",
        "vpc:Describe*",
        "iam:Get*",
        "iam:List*",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:List*",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

僅授予滲透測試範圍所需的最低許可。根據您要包含在安全性測試中的 AWS 服務，檢閱和調整許可。

演員角色

執行者角色用於驗證和授權滲透測試期間對目標 Web 應用程式的請求。這些角色會在代理程式空間建立期間提供，AWS Security Agent 代理程式會擔任這些角色，以與您的目標應用程式端點（例如 AWS API Gateway APIs、Lambda URLs 或其他 AWS 託管的應用程式）互動。

所需的許可

此角色需要下列許可：

- 叫用 API Gateway 端點
- 執行 Lambda 函數

- 存取應用程式特定的 AWS 資源
- 使用目標應用程式的身分驗證機制進行身分驗證
- 針對應用程式端點執行 HTTP 操作

信任政策

信任政策必須允許 AWS Security Agent 服務擔任此角色：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

許可政策

許可取決於您的目標應用程式架構。以下是常見案例的範例：

對於 API Gateway 應用程式

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "execute-api:Invoke"
      ],
      "Resource": "arn:aws:execute-api:us-east-1:*:your-api-id/*"
```

```
    }
  ]
}
```

對於 Lambda 函數 URLs

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "lambda:InvokeFunctionUrl",
        "lambda:InvokeFunction"
      ],
      "Resource": "arn:aws:lambda:us-east-1:*:function:your-function-name"
    }
  ]
}
```

適用於具有 Cognito 身分驗證的 Application Load Balancer

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cognito-idp:InitiateAuth",
        "cognito-idp:RespondToAuthChallenge"
      ],
      "Resource": "arn:aws:cognito-idp:us-east-1:*:userpool/your-user-pool-id"
    }
  ]
}
```

Important

設定演員角色許可，以符合目標應用程式的身分驗證和授權要求。角色應具有與 Security Agent 在滲透測試期間模擬的使用者或服務相同的存取層級。

AWS Security Agent 的客戶受管金鑰

根據預設， AWS Security Agent 會使用 AWS受管加密金鑰來加密所有靜態資料。您可以選擇使用來自 AWS Key Management Service (AWS KMS) 的客戶受管金鑰來加密資料，讓您完全控制保護 資源的加密金鑰。

AWS Security Agent 支援資源層級的客戶受管金鑰。當您建立最上層資源，例如 代理程式空間或整合時，您可以指定 KMS 金鑰來加密屬於該資源及其子資源的所有資料。例如，在建立 Agent Space 時指定客戶受管金鑰，會加密與該 Agent Space 相關聯的所有資料，包括 Agent Space 組態、滲透測試組態、任務和執行詳細資訊、安全調查結果、探索的端點和螢幕擷取畫面。您也可以提供已使用自己的客戶受管金鑰加密 AWS 的資源，例如 S3 儲存貯體、CloudWatch Logs 日誌群組或 Secrets Manager 秘密。如需必要的 KMS 許可，請參閱 the section called “必要的 KMS 許可”。

客戶受管金鑰的運作方式

AWS 安全代理程式使用 AWS KMS 階層 keyring，以客戶受管金鑰加密您的資料。當您在資源建立期間指定 KMS 金鑰時，服務會建立受 KMS 金鑰保護的分支金鑰，並將其存放在 Amazon DynamoDB 型金鑰存放區中。對於每個加密操作，階層式 keyring 會從作用中分支金鑰衍生唯一的包裝金鑰，這會為每個請求加密唯一的資料加密金鑰。

階層式 keyring 提供下列優點：

- 每個資源加密範圍 – 每個頂層資源都有自己的分支金鑰，因此不同資源的資料會使用不同的金鑰加密。
- 自動金鑰輪換 – AWS 安全代理程式會定期輪換分支金鑰。輪換後，新資料會使用新的分支金鑰版本加密，而較舊版本則會保留以解密先前加密的資料。

加密內容

AWS 安全代理程式會搭配 KMS 金鑰在所有密碼編譯操作中使用加密內容。加密內容是一組非私密的金鑰值對，可為加密操作提供額外的已驗證資料。

加密內容金鑰遵循格式 `aws:securityagent:_<resource-type>_`，其中 *`<resource-type>`* 是要加密的資源類型（例如 `agent-space`或 `integration`)。值是資源的 Amazon Resource Name (ARN)。

> **Note**
>
> 對於整合，加密內容金鑰包含 `aws-crypto-ec:`字首，因此格式為 `aws-crypto-ec:aws:securityagent:integration`。

您可以使用加密內容，將 KMS 金鑰政策縮小至特定資源類型。如需詳細資訊，請參閱the section called “必要的 KMS 許可”。

Default encryption (預設加密)

當您在不指定客戶受管金鑰的情況下建立資源時， AWS Security Agent 會使用基礎儲存服務 (Amazon DynamoDB 和 Amazon S3) 提供的 AWS受管加密金鑰來加密資源。預設加密不需要額外的組態。

應用程式的預設 KMS 金鑰

您可以在應用程式層級設定預設 KMS 金鑰。設定預設 KMS 金鑰時，當您在資源建立期間未明確指定 KMS 金鑰時， AWS Security Agent 會將其用作新 Agent Spaces 和整合的後援。

若要設定預設 KMS 金鑰，請在使用 CLI 或 SDK AWS 建立或更新應用程式時指定 `defaultKmsKeyId` 參數。主控台會在應用程式設定期間提示您指定預設 KMS 金鑰。

當您在資源建立期間明確指定 KMS 金鑰時，其優先順序會高於應用程式層級預設值。

先決條件

設定客戶受管金鑰之前，請先完成下列先決條件：

- 在 KMS 中建立對稱加密 AWS KMS 金鑰。金鑰必須符合下列要求：
  - 金鑰類型：對稱
  - 金鑰用量：加密和解密
  - 金鑰規格： `SYMMETRIC_DEFAULT`
  - 金鑰狀態：已啟用

  如需說明，請參閱 Key Management Service 開發人員指南中的建立 AWS 金鑰。
- 設定 KMS 金鑰政策和 IAM 政策，以授予 AWS 安全代理程式所需的許可。如需詳細資訊，請參閱the section called “必要的 KMS 許可”。

必要的 KMS 許可

AWS 安全代理程式在兩種情況下需要 KMS 許可：

- 在 AWS 安全代理程式中加密資料 – 當您為代理程式空間或整合指定客戶受管金鑰時，服務需要許可，才能將該金鑰用於資料上的密碼編譯操作。
- 使用 CMK 加密 AWS 的資源 – 當您提供以客戶受管金鑰（例如 S3 儲存貯體、CloudWatch Logs 日誌群組或 Secrets Manager 秘密）加密 AWS 的資源時，服務需要許可，才能使用這些金鑰來存取加密的資源。

AWS 安全代理程式會根據操作，使用不同的身分存取您的 KMS 金鑰：

- AWS 管理主控台操作 – 當管理員在 AWS 管理主控台中建立或管理資源（例如，建立客服人員空間或更新應用程式）時，服務會使用管理員角色來呼叫 AWS KMS。
- Web 應用程式操作 – 當 IAM Identity Center 使用者透過 AWS Security Agent Web 應用程式存取資料時（例如，檢視滲透測試結果或啟動滲透測試），服務會使用 AWS Security Agent 設定期間建立的應用程式角色來呼叫 AWS KMS。
- 存取客戶提供的資源 – 當服務在滲透測試期間存取客戶提供 AWS 的資源時（例如 S3 儲存貯體、CloudWatch Logs 日誌群組和 Secrets Manager 秘密），它會使用滲透測試服務角色來呼叫 AWS KMS。
- 非同步工作流程 – 對於在任何使用者工作階段之外執行的背景操作（例如，滲透測試執行、程式碼檢閱處理和分支金鑰輪換），服務會使用自己的服務主體 (`securityagent.amazonaws.com`) 代表您呼叫 AWS KMS。

下列各節說明每個身分的必要許可。

金鑰政策

您的 KMS 金鑰政策必須授予 AWS 安全代理程式許可，才能使用金鑰進行密碼編譯操作。所需的金鑰政策陳述式取決於您計劃加密的資源類型，以及您提供給服務的 CMK 加密 AWS 資源。

代理程式空間的金鑰政策

下列金鑰政策會授予 AWS 安全代理程式加密和解密代理程式空間資料所需的許可，包括滲透測試結果和螢幕擷取畫面。

取代政策中的下列預留位置值：

- *111122223333* – AWS 您的帳戶 ID

- *MyRole* – 您在主控台中用來管理 AWS Security Agent 的 IAM 角色
- *MyApplicationRole* – 在 AWS Security Agent 設定期間建立的應用程式角色
- *us-east-1* – 您使用 AWS 安全代理程式 AWS 的區域

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidationForApplicationAndAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowUseOfHierarchicalKeyringForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
```

```
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "ArnLike": {
```

```
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForCodeRemediation",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

若要輪換分支金鑰，您的 KMS 金鑰政策必須將 `kms:GenerateDataKeyWithoutPlaintext`、 和 `kms:ReEncryptFrom`許可授予 AWS Security Agent 服務主體 (`securityagent.amazonaws.com`)`kms:ReEncryptTo`，否則分支金鑰輪換將會失敗。如需所需許可的詳細資訊，請參閱輪換分支金鑰。

整合的金鑰政策

下列金鑰政策會授予 AWS 安全代理程式加密和解密整合資料所需的許可，包括程式碼檢閱問題清單。

取代政策中的下列預留位置值：

- *111122223333* – AWS 您的帳戶 ID
- *MyRole* – 您在主控台中用來管理 AWS Security Agent 的 IAM 角色
- *us-east-1* – 您使用 AWS 安全代理程式 AWS 的區域

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Sid": "AllowKeyAccessValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
          "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
"arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKeyMetadataValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "AllowAsynchronousDataAccessForIntegrations",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
```

```
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
 "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        }
      }
    }
  ]
}
```

> **Note**
>
> 如果您想要將相同的 KMS 金鑰用於多種資源類型，您可以將 Agent Space、整合和安全需求套件金鑰政策陳述式合併為單一金鑰政策。

安全性需求套件的金鑰政策

下列金鑰政策會授予 AWS 安全代理程式加密和解密安全需求套件資料所需的許可。安全需求套件不使用 Web 應用程式角色。此政策使用兩個存取路徑：

- 同步路徑 – 當您呼叫 API 時，服務會使用您轉送的登入資料代表您呼叫 AWS KMS （透過 `kms:ViaService`條件）。
- 非同步路徑 – 對於可使用套件的非同步操作，服務會使用自己的服務主體直接呼叫 AWS KMS。

取代政策中的下列值：

- *111122223333* – AWS 您的帳戶 ID
- *MyRole* – 您用來叫用安全需求套件操作的 IAM 角色
- *us-east-1* – 您使用 AWS 安全代理程式 AWS 的區域

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidation",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowUseOfHierarchicalKeyringForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
"arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
```

```
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
 "arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:security-
requirement-pack/*"
        }
      }
    }
  ]
}
```

此政策包含三個陳述式：

- AllowKeyMetadataValidation – 授予 ，kms:DescribeKey讓 AWS 安全代理程式可以驗證您的客戶受管金鑰是否存在、已啟用，並在資源建立期間指定 CMK 時使用對稱加密。使用 kms:ViaService條件來確保呼叫是透過 服務發出。
- AllowUseOfHierarchicalKeyringForSecurityPacks – 針對階層式 keyring 操作授予 kms:GenerateDataKeyWithoutPlaintext（建立新的分支金鑰）、 kms:ReEncryptTo 和 kms:ReEncryptFrom（輪換現有的分支金鑰） 以及 kms:Decrypt（擷取現有的分支金鑰）。這些操作會透過同步路徑使用您的轉送憑證，並依加密內容條件範圍限定為安全需求套件資源。
- AllowAsynchronousDataAccessForSecurityPacks – 當沒有呼叫者登入資料可用時kms:GenerateDataKeyWithoutPlaintext，kms:ReEncryptFrom將非同步操作的 kms:ReEncryptTo、、 和 kms:Decrypt授予 AWS Security Agent 服務主體。

與 Agent Spaces 金鑰政策不同，安全需求套件政策不包含 Web 應用程式角色主體。安全需求套件是使用您的管理員角色透過 AWS 管理主控台存取，而不是透過 AWS Security Agent Web 應用程式存取。所有同步操作都使用單一呼叫者角色 （透過 kms:ViaService)。

> **Note**
>
> 如果您針對 Agent Spaces 和安全需求套件使用相同的 KMS 金鑰，您可以將兩個區段的金鑰政策陳述式合併為單一金鑰政策。

CMK 加密 AWS 資源的金鑰政策

如果您使用客戶受管金鑰加密 AWS 的資源提供給 AWS 安全代理程式，則必須更新每個資源 KMS 金鑰的金鑰政策，以授予必要的許可。這適用於下列資源：

- S3 儲存貯體 – 如果您從使用客戶受管金鑰 (SSE-KMS) 加密的 S3 儲存貯體提供學習資源，則金鑰政策必須允許滲透測試服務角色解密物件。
- Secrets Manager 秘密 – 如果您的滲透測試登入資料存放在使用客戶受管金鑰加密的 Secrets Manager 秘密中，則金鑰政策必須允許滲透測試服務角色解密這些秘密。如果 Web 應用程式代表您建立秘密，金鑰政策也必須允許應用程式角色加密這些秘密。
- CloudWatch Logs 日誌群組 – 如果用於存放滲透測試執行日誌的 CloudWatch Logs 日誌群組使用客戶受管金鑰加密，金鑰政策必須允許 CloudWatch Logs 透過服務角色驗證 KMS 金鑰，並允許 CloudWatch Logs 服務主體執行密碼編譯操作。

> **Important**
>
> AWS 安全代理程式也可以代表您建立 CloudWatch Logs 日誌群組和 Secrets Manager 秘密。設定 KMS 金鑰政策時，也請包含這些服務建立之資源的對應陳述式。

下列金鑰政策陳述式會授予 CMK 加密資源的必要許可。僅包含適用於您組態的陳述式。如果資源使用您為客服人員空間指定的相同 KMS 金鑰，請將這些陳述式新增至該金鑰的政策。如果資源使用不同的 KMS 金鑰，請改為將這些陳述式新增至該金鑰的政策。

取代政策中的下列預留位置值：

- *111122223333* – AWS 您的帳戶 ID
- *MyApplicationRole* – 在 AWS Security Agent 設定期間建立的應用程式角色
- *MyPenTestServiceRole* – 滲透測試服務角色
- *us-east-1* – 您使用 AWS 安全代理程式 AWS 的區域

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForS3Objects",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
```

```
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForSecretsManagerSecrets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCloudWatchLogsLogGroups",
```

```
      "Effect": "Allow",
      "Principal": {
        "Service": "logs.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:logs:arn": "arn:aws:logs:us-
east-1:111122223333:log-group:YOUR-LOG-GROUP-NAME*"
        }
      }
    }
  ]
}
```

Note

- 只有在您從使用客戶受管金鑰加密的 S3 儲存貯體提供學習資源時，才需要 `AllowKmsKeyDecryptionForS3Objects`陳述式。如需設定 SSE-KMS for S3 的詳細資訊，請參閱使用 SSE-KMS 保護資料。
- 只有當您的滲透測試登入資料存放在使用客戶受管金鑰加密的 Secrets Manager 秘密中時，才需要 `AllowKmsKeyDecryptionForSecretsManagerSecrets`陳述式。服務角色需要存取權，才能在滲透測試期間解密秘密。如需秘密加密的詳細資訊，請參閱 Secrets Manager 中的秘密加密和解密。
- `AllowKmsKeyValidationForCloudWatchLogsLogGroups` 陳述式會授予服務角色，`kms:DescribeKey`以便 CloudWatch Logs 在將其與日誌群組建立關聯時，可以透過服務角色驗證 KMS 金鑰。
- 如果 Web 應用程式使用客戶受管金鑰代表您建立 Secrets Manager 秘密，則需要 `AllowKmsKeyAccessForCreatingSecrets`陳述式。應用程式角色需要 `kms:GenerateDataKey`和 ，`kms:Decrypt`才能使用 KMS 金鑰加密秘密。

- `AllowKmsKeyAccessForCloudWatchLogsLogGroups` 陳述式會授予 CloudWatch Logs 服務主體 (`logs.us-east-1.amazonaws.com`) 加密和解密日誌資料所需的許可。如果 AWS Security Agent 在您未提供現有日誌群組時代表您建立日誌群組，也需要此陳述式。如需詳細資訊，請參閱使用 AWS KMS 在 CloudWatch Logs 中加密日誌資料。
- 如果多個資源共用相同的 KMS 金鑰，您可以將陳述式合併為單一金鑰政策。

管理員角色

管理員角色（您在主控台中用來管理 AWS Security Agent 的 IAM 角色）不需要額外的 IAM 政策。

應用程式角色

除了 KMS 金鑰政策之外，請將下列 IAM 政策連接至 AWS Security Agent 設定期間指定的應用程式角色。此政策授予角色許可，以使用您的客戶受管金鑰來加密和解密 Agent Space 資料，並在 AWS Secrets Manager 中建立秘密。

取代政策中的下列預留位置值：

- *111122223333* – AWS 您的帳戶 ID
- *us-east-1* – 您使用 AWS Security Agent AWS 的區域
- 中的 KMS 金鑰 ARNs `Resource` – KMS 金鑰ARNs

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333",
```

```
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    }
  ]
}
```

> **Note**
>
> 應用程式角色是您指定的 IAM 角色，或主控台在 AWS Security Agent 設定期間建立的 IAM 角色。Web 應用程式會擔任此角色來擷取滲透測試執行日誌，並代表您建立 Secrets Manager 秘密。
>
> - 如果您設定滲透測試的身分驗證資源，並選擇直接輸入登入資料，而不是指定現有的秘密，則需要 `AllowKmsKeyAccessForCreatingSecrets`陳述式。Web 應用程式會代表您建立秘密，而應用程式角色需要此陳述式中的許可，才能使用為 Agent Space 指定的客戶受管

金鑰來加密秘密。更新此陳述式中的 Resource ARNs，以符合用於代理程式空間的 KMS 金鑰。

服務 角色

在滲透測試期間， AWS Security Agent 會擔任滲透測試服務角色來存取您的 AWS 資源。如果其中任何資源使用客戶受管金鑰加密，您必須授予服務角色使用對應 KMS 金鑰的許可。這適用於下列資源：

- S3 儲存貯體 – 如果您從使用客戶受管金鑰加密的 S3 儲存貯體提供學習資源 （例如 API 文件、威脅模型或原始程式碼），服務角色需要許可才能解密該儲存貯體中的物件。如需設定 SSE-KMS for S3 的詳細資訊，請參閱使用 SSE-KMS 保護資料。
- Secrets Manager 秘密 – 如果您的滲透測試登入資料存放在使用客戶受管金鑰加密的 Secrets Manager 秘密中，服務角色需要許可才能解密這些秘密。如需秘密加密的詳細資訊，請參閱 Secrets Manager 中的秘密加密和解密。
- CloudWatch Logs 日誌群組 – 如果用於存放滲透測試執行日誌的 CloudWatch Logs 日誌群組使用客戶受管金鑰 （包括由 AWS Security Agent 代表您建立的日誌群組） 加密，服務角色需要`kms:DescribeKey`許可才能驗證金鑰。CloudWatch Logs 服務主體會處理日誌資料的實際加密和解密，並透過金鑰政策授予這些許可 （請參閱 the section called “金鑰政策”)。如需加密日誌資料的詳細資訊，請參閱使用 AWS KMS 在 CloudWatch Logs 中加密日誌資料。

> **Important**
>
> 如果您使用不同於您為客服人員空間指定的 KMS 金鑰來加密這些資源，您必須授予每個 KMS 金鑰的服務角色許可，以保護服務角色在滲透測試期間存取的資源。

將下列 IAM 政策連接至滲透測試服務角色。僅包含適用於您組態的陳述式。

取代政策中的下列預留位置值：

- *111122223333* – AWS 您的帳戶 ID
- *us-east-1* – 您使用 AWS Security Agent AWS 的區域
- 中的 KMS 金鑰 ARNs `Resource` – 用於加密每個資源的客戶受管金鑰ARNs

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Sid": "AllowKmsAccessForEncryptedS3Buckets",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-S3-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsAccessForEncryptedSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-SECRETS-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Action": [
```

```
        "kms:DescribeKey"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-CLOUDWATCH-LOGS-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    }
  ]
}
```

Note

- 只有在您從使用客戶受管金鑰加密的 S3 儲存貯體提供學習資源時，才需要 `AllowKmsAccessForEncryptedS3Buckets`陳述式。更新 `Resource` ARN 以符合用於加密 S3 儲存貯體的 KMS 金鑰，並更新`kms:EncryptionContext:aws:s3:arn`值以符合您的儲存貯體名稱。
- 只有當您的滲透測試登入資料存放在使用客戶受管金鑰加密的 Secrets Manager 秘密中時，才需要 `AllowKmsAccessForEncryptedSecrets`陳述式。更新 `Resource` ARN 以符合用於加密秘密的 KMS 金鑰，並更新`kms:EncryptionContext:SecretARN`值以符合秘密名稱。
- 只有當您的 CloudWatch Logs 日誌群組使用客戶受管金鑰加密時，才需要 `AllowKmsKeyValidationForCloudWatchLogsLogGroups`陳述式，包括由 AWS Security Agent 代表您建立的日誌群組。更新 `Resource` ARN 以符合用於加密日誌群組的 KMS 金鑰。
- 如果多個資源共用相同的 KMS 金鑰，您可以合併陳述式，並在 `Resource` 欄位中列出共用金鑰 ARN 一次。

使用客戶受管金鑰建立資源

您可以在設定 AWS Security Agent 或建立個別資源時指定客戶受管金鑰。KMS 金鑰會加密屬於該資源及其子資源的所有資料。

在設定期間設定預設 KMS 金鑰（主控台）

您可以在初始 AWS 安全代理程式設定期間設定預設 KMS 金鑰。除非您指定不同的金鑰，否則此金鑰會做為新 Agent Spaces 和整合的預設值。

1. 在設定 AWS 安全代理程式頁面上，展開加密 - 選用區段。
2. 選取自訂加密設定（進階）。
3. 針對選擇 AWS KMS 金鑰，請從您的帳戶選取 KMS 金鑰，或輸入 KMS 金鑰 ARN。
4. 完成其餘的設定步驟，然後選擇設定 AWS 安全代理程式。

AWS Security Agent 會驗證 KMS 金鑰，以確認其存在、已啟用並使用對稱加密。如果驗證失敗，設定不會繼續，而且您會收到錯誤訊息。

使用客戶受管金鑰建立客服人員空間（主控台）

1. 在 AWS 安全代理程式主控台中，導覽至代理程式空間頁面。
2. 選擇建立客服人員空間。
3. 輸入代理程式空間的名稱和選用描述。
4. 展開 Advanced (進階) 區段。
5. 在資料加密下，選擇下列其中一項：
   - 使用應用程式預設金鑰 – 使用 AWS Security Agent 設定期間設定的預設 KMS 金鑰。如果已設定預設金鑰，則預設會選取此選項。
   - 使用不同的金鑰 – 為此客服人員空間指定不同的 KMS 金鑰。選取自訂加密設定（進階），然後針對選擇 AWS KMS 金鑰，從您的帳戶選取 KMS 金鑰或輸入 ARN。
6. 選擇建立。

建立與客戶受管金鑰的整合（主控台）

1. 在 AWS Security Agent 主控台中，導覽至整合頁面。
2. 選擇新增整合，然後選取您的提供者（例如 GitHub)。
3. 完成步驟 1：安裝並授權提供者應用程式。

4. 在步驟 2：註冊詳細資訊中，輸入註冊名稱並設定提供者設定。
5. 在資料加密區段中，選取自訂加密設定 （進階）。
6. 針對選擇 AWS KMS 金鑰，請從您的帳戶選取 KMS 金鑰，或輸入 KMS 金鑰 ARN。
7. 選擇連線。

使用客戶受管金鑰 (AWS CLI 或 SDK) 建立資源

若要使用 CLI 或 SDK AWS 指定客戶受管金鑰：

- 呼叫 `CreateApplication` 為您的應用程式設定預設 KMS 金鑰時，請包含 `defaultKmsKeyId` 參數。在沒有明確 KMS 金鑰的情況下建立 Agent Spaces 或整合時，此金鑰會用作備用金鑰。
- 呼叫 `CreateAgentSpace`或 `CreateIntegration` 以加密特定資源時，請包含 `kmsKeyId` 參數。這優先於應用程式層級預設值。

如需參數詳細資訊，請參閱 AWS Security Agent API 參考。

如果您未指定 KMS 金鑰，則如果已設定 KMS 金鑰， AWS Security Agent 會使用應用程式層級的預設 KMS 金鑰。否則，資源會使用 AWS受管金鑰加密。

金鑰輪換

AWS Security Agent 會定期自動輪換分支金鑰。分支金鑰輪換會建立新的分支金鑰作用中版本，同時保留所有先前的版本。輪換後：

- 新資料會使用新的分支金鑰版本加密。
- 先前加密的資料仍然可以使用較舊的分支金鑰版本來解密。
- 不需要資料重新加密。

分支金鑰輪換與 KMS 金鑰輪換不同。您可以為客戶受管金鑰獨立啟用自動 KMS 金鑰輪換。如需詳細資訊，請參閱 Key Management Service 開發人員指南 AWS 中的輪換 KMS 金鑰。

在分支金鑰輪換之後，有一段短暫的期間，上一個分支金鑰的快取複本仍可用於新的加密操作。此時段由內部快取time-to-live(TTL) 決定，不會影響您資料的安全性。

考量事項

搭配 AWS Security Agent 使用客戶受管金鑰時，請記住下列事項：

- 客戶受管金鑰僅適用於新資源。在您設定客戶受管金鑰之前建立的現有資源會繼續使用 AWS受管加密。現有資料不會遷移。
- 客戶受管金鑰會在建立時指定。您可以在建立資源時指定 KMS 金鑰。您無法新增或變更現有資源的 KMS 金鑰。
- 支援多個 KMS 金鑰。您可以針對不同的資源使用不同的 KMS 金鑰。例如，您可以將一個金鑰用於生產 Agent Spaces，將另一個金鑰用於開發 Agent Spaces。
- 停用或刪除 KMS 金鑰會使資料無法存取。如果您停用或排程刪除 KMS 金鑰， AWS Security Agent 無法解密該金鑰下加密的資料。受影響的資源會變成無法存取，直到金鑰重新啟用為止。永久刪除 KMS 金鑰可防止存取在該金鑰下加密的所有資料。
- 金鑰政策許可是必要的。如果您從 KMS 金鑰政策中移除必要的許可， AWS Security Agent 無法存取加密的資料。確保金鑰政策授予管理員角色（用於在 AWS 主控台中管理服務）、應用程式角色（由 Web 應用程式使用）、滲透測試服務角色（由客服人員擔任以執行滲透測試）和 AWS Security Agent 服務主體的許可，如中所述the section called “必要的 KMS 許可”。
- AWS 適用 KMS 配額。針對 KMS 金鑰的加密操作會計入 AWS KMS 請求配額。在正常使用情況下，階層式 keyring 架構會透過分支金鑰快取將 KMS 呼叫降至最低。如需詳細資訊，請參閱 AWS Key Management Service 開發人員指南中的請求配額。

疑難排解

尋找使用 AWS Security Agent 時常見錯誤的解決方案。

拒絕存取：選取不正確的 GitHub 帳戶類型或指定的組織名稱不正確

- 您已將 AWS Security Agent 應用程式安裝到所需的 GitHub 組織中，但錯誤地`GitHub Account Type`將 設定為 ，`User`而不是 `Organization`
- 您已將 AWS Security Agent 應用程式安裝到所需的 GitHub 組織中，並正確`GitHub Account Type`將 設定為 ，`Organization`但將 `Organization Name` 欄位保留空白，或輸入與您安裝應用程式的組織不相符的不正確組織名稱

解決方案：

1. 前往 GitHub 並從組織解除安裝應用程式。如需詳細資訊，請參閱the section called “步驟 1：從 GitHub 解除安裝 AWS 安全代理程式 GitHub 應用程式”。
2. 返回整合頁面，按一下 重新啟動整合程序`Add Integrations`，再次將應用程式安裝並授權至您想要的 GitHub 組織。

3. Organization 從 的下拉式清單中選取 GitHub account type
4. 請確定Organization Name您輸入的 EXACT 與您安裝應用程式的 EXACT 相同。
5. 選擇連線以建立 GitHub 組織整合。

拒絕存取：在組織中安裝 GitHub App 的許可不足

當您嘗試將 AWS Security Agent 應用程式安裝到所需的 GitHub 組織時，您會在安裝頁面的 按鈕上看到兩個不同的訊息。

- 組織Member將看到 Authorize & Request
- 組織Owner將看到 Install & Authorize

您可以依照下列步驟，驗證您是 GitHub 組織的 Member還是 Owner 。

1. 前往 https：//github.com
2. 在網站上選擇您的設定檔
3. 導覽至下拉式選單Organizations上的 ，然後選擇它
4. 從組織清單中尋找您想要安裝 AWS Security Agent 的組織，它會指定您是組織名稱Owner旁的 Member還是 。

可能的解決方案：

- 在您嘗試建立整合之前，請擁有者核准您的安裝請求
- 讓擁有者將 GitHub 組織中的角色從 更新Member為 ，Owner然後重新啟動整合程序

客服人員無法在滲透測試期間連線至端點

如果滲透測試代理程式無法呼叫設定的目標 URL，或無法成功導覽目標端點：

- 如果您的端點呼叫所設定目標 URL 以外的網域，請確認其他網域已新增為 pentest 組態中的可存取 URLs
- 滲透測試目前僅適用於在連接埠 80 或 443 上提供流量的 HTTP/HTTPS 端點
- 如果您已設定 WAF，請檢查您的 WAF 是否未封鎖滲透測試流量。您可以依User-Agent標頭允許列出滲透測試流量，預設會設為 securityagent

取得其他協助

如果您在嘗試這些故障診斷步驟後仍遇到問題：

- 檢查 AWS Security Agent 服務狀態頁面
- 如需複雜組態問題的協助，請聯絡 AWS Support

適用於使用者和開發人員 (Web 應用程式和 IDE)

在 Web 應用程式、IDE 或連線來源提供者中執行設計檢閱、威脅模型、程式碼檢閱和滲透測試，然後檢閱和修復問題清單。

登入 AWS Security Agent Web 應用程式

您應該登入 AWS Security Agent Web 應用程式來完成下列任務：

- 執行設計檢閱
- 執行滲透測試

存取方法

AWS Security Agent 提供存取 Web 應用程式的不同方法，取決於您的組織在初始設定期間如何設定存取，以及您是否具有 AWS 主控台存取。

IAM Identity Center (SSO) - 如果您的組織已啟用 IAM Identity Center，您可以使用 SSO 憑證登入。您必須先指派給 IAM Identity Center 中的至少一個代理程式空間，才能存取 Web 應用程式。

管理員存取 (IAM) - 如果您有具有適當許可的 AWS 主控台存取，您可以透過任何客服人員空間頁面上的管理員存取連結來啟動 Web 應用程式。此方法透過您現有的主控台工作階段提供身分驗證。

> **Note**
>
> 您的組織使用的主要存取方法是在初始 AWS Security Agent 設定期間決定的。如需設定使用者存取和指派的資訊，請參閱 the section called “授予使用者存取 AWS Security Agent Web 應用程式的權限”。

使用 IAM Identity Center (SSO) 登入

如果您的組織設定了 IAM Identity Center 存取，您可以透過多個進入點使用您的 SSO 憑證登入 Web 應用程式。

先決條件

- 您已在 IAM Identity Center 中指派給至少一個客服人員空間

- 您有 IAM Identity Center SSO 登入資料

存取 Web 應用程式

根據您的需求選擇下列其中一種方法：

若要檢視指派給您的所有客服人員空間：

1. 在 AWS Security Agent 主控台中，導覽至設定。
2. 找到 Web 應用程式網域並複製 URL。

> **Tip**
>
> 將此 URL 加入書籤以方便存取。這是檢視您獲指派的所有客服人員空間的通用進入點。

3. 導覽至 Web 應用程式 URL。
4. 出現提示時，輸入您的 IAM Identity Center 憑證。
5. 身分驗證之後，您會看到指派給您的所有客服人員空間清單。
6. 選取您要使用的客服人員空間。

若要直接存取特定客服人員空間（需要 AWS 主控台存取）：

1. 登入 AWS 管理主控台。
2. 導覽至 AWS Security Agent 主控台。
3. 導覽至您要存取的客服人員空間。
4. 選擇下列其中一項：
   - 客服人員空間概觀頁面上的啟動 Web 應用程式按鈕
   - 程式碼檢閱區段中的啟動 Web 應用程式按鈕
   - 滲透測試區段中的啟動 Web 應用程式按鈕
5. 出現提示時，輸入您的 IAM Identity Center 憑證。
6. 身分驗證之後，您將直接前往 Web 應用程式中的該 Agent Space。

> **Note**
>
> 如果您沒有 AWS 主控台存取權，請從設定頁面使用 Web 應用程式網域來存取所有指派的客服人員空間。

使用管理員存取登入 (IAM)

如果您有具有適當 AWS Security Agent 許可的 AWS 主控台存取，您可以透過具有自動身分驗證的管理員存取連結來啟動 Web 應用程式。

先決條件

- 您已登入 AWS 管理主控台
- 您擁有存取 AWS Security Agent 資源的適當許可

存取 Web 應用程式

選擇下列其中一種方法：

若要存取特定客服人員空間：

1. 登入 AWS 管理主控台。
2. 導覽至 AWS Security Agent 主控台。
3. 導覽至您要存取的客服人員空間。
4. 選擇客服人員空間概觀頁面上的管理員存取按鈕。
5. Web 應用程式會在新標籤中開啟，並自動驗證該代理程式空間。

> **Note**
>
> 無論您的組織是否使用 IAM Identity Center 作為主要存取方法，管理員存取按鈕一律可供具有 AWS 主控台存取的使用者使用。

建立設計檢閱

透過上傳檔案供 AWS Security Agent 檢閱，根據組織安全需求評估您的設計文件。設計審查有助於在開發生命週期的早期識別安全問題，讓您能夠在最符合成本效益的情況下解決架構問題。

AWS Security Agent 會根據組織的安全需求分析您的設計文件，提供詳細的安全調查結果，以在實作開始之前改善安全狀態。

在此程序中，您將上傳設計檔案進行分析，以建立設計檢閱。

先決條件

開始前，請確保您具備以下條件：

- 存取 AWS Security Agent Web 應用程式
- 設計可上傳的文件 (DOC、DOCX、JPEG、MD、PDF、PNG 和 TXT)
- 每個檔案都必須為 2MB 或更小，且所有檔案總計為 6MB
- 了解您的組織已啟用哪些安全要求

步驟 1：開始建立設計檢閱

導覽至 Agent Web 應用程式中的設計檢閱建立頁面。

1. 登入 AWS Security Agent Web 應用程式。
2. 導覽至設計檢閱區段。
3. 選擇建立設計檢閱。

> **Tip**
>
> 您可以導覽至 AWS Security Agent 主控台中的安全需求頁面，以檢視組織已啟用的安全需求。選取任何啟用的需求以檢視其詳細資訊。這些要求用於分析您的設計檔案。

步驟 2：為您的設計檢閱命名

提供描述性名稱，以協助識別此設計檢閱的目的和範圍。

1. 在設計檢閱名稱區段中，找到名稱欄位。
2. 輸入設計檢閱的描述性名稱。

> **Note**
>
> 名稱應清楚識別要檢閱的專案、功能或元件。最多 80 個字元。

步驟 3：上傳設計檔案

上傳您希望 AWS Security Agent 分析安全性合規性的設計文件。

1. 在要檢閱的檔案區段中，檢閱檔案需求：

> **Important**
>
> 每次設計檢閱最多可以上傳 5 個檔案。每個檔案都必須為 2MB 或更小，所有檔案的總和為 6MB。支援的格式：DOC、DOCX、JPEG、MD、PDF、PNG 和 TXT。

2. 使用下列其中一種方法上傳您的檔案：
   a. 拖放 – 直接將檔案拖曳到檔案放置區域區域
   b. 瀏覽 – 使用選擇檔案按鈕，從您的電腦瀏覽和選取檔案
3. 確認已上傳所有必要的檔案。

> **Tip**
>
> 為了獲得最佳結果，請包含架構圖、設計規格，以及描述系統安全相關元件和資料流程的技術文件。

步驟 4：啟動設計檢閱

設定所有必要資訊後，啟動設計文件的安全分析。

1. 檢閱所有上傳的檔案和設定，以確保準確性。
2. 選擇開始設計檢閱。
3. AWS Security Agent 會根據已啟用的安全需求來分析您的設計文件。

> **Note**
>
> 設計檢閱程序通常會在幾分鐘內完成，取決於上傳的檔案數量和大小。根據組織的安全需求，您會收到安全調查結果。

後續步驟

開始設計檢閱後：

- 在 Agent Web 應用程式中監控檢閱進度
- 檢閱安全性問題清單
- 與您的開發團隊共用問題清單
- 解決設計中已識別的安全性問題清單
- 更新設計文件並視需要重新提交

檢閱設計檢閱的問題清單

檢閱調查結果可協助您了解符合哪些安全需求、需要注意哪些安全需求，以及在實作開始之前要採取哪些動作來改善設計的安全狀態。

在此程序中，您將了解如何存取、篩選和解釋設計審查問題清單，以有效地解決安全問題清單。

先決條件

開始前，請確保您具備以下條件：

- 已完成的設計檢閱
- 存取 AWS Security Agent Web 應用程式
- 熟悉您組織已啟用的安全需求

步驟 1：存取設計檢閱

導覽至您的設計檢閱，以檢視問題清單和摘要資訊。

1. 登入 AWS Security Agent Web 應用程式。

2. 導覽至設計檢閱區段。
3. 從清單中選擇您要檢查的設計檢閱。

> **Tip**
>
> 設計檢閱詳細資訊頁面會顯示檢閱狀態、完成日期和檢閱檔案數量的摘要。

步驟 2：檢閱調查結果摘要

檢查高階摘要，以了解您設計的整體安全狀態。

1. 找到摘要區段。
2. 檢閱每個合規狀態類別的計數：合規、不合規、資料不足和不適用。

> **Note**
>
> 摘要提供每種狀態類型的計數，協助您快速評估需要注意的問題清單數量。如需每個狀態的詳細說明，請參閱步驟 4。

步驟 3：篩選和導覽問題清單

使用篩選和搜尋功能來專注於特定問題清單或合規狀態。

1. 在檢閱問題清單區段中，找到篩選條件控制項。
2. 若要依狀態篩選：
   a. 選擇狀態下拉式選單。
   b. 選取特定合規狀態，以僅檢視具有該狀態的問題清單。
3. 若要搜尋特定安全需求：
   a. 在搜尋欄位中輸入關鍵字。
   b. 結果會在您輸入時自動更新。
4. 使用分頁控制項瀏覽多個問題清單頁面。

> **Tip**
>
> 首先依不合規狀態篩選，以排定在設計中需要立即關注的問題清單的優先順序。

步驟 4：了解合規狀態

每個調查結果都會顯示合規狀態，指出您的設計如何處理特定安全需求：

- 合規 – 根據分析，您的設計符合安全要求
- 不合規 – 您的設計違反或不適當解決安全要求
- 資料不足 – 上傳的檔案缺少足夠的資訊來判斷合規性
- 不適用 – 安全需求不適用於您的系統設計

> **Important**
>
> 專注於不合規和資料不足狀態，因為這些需要採取動作。透過更新您的設計來解決不合規的問題清單，並透過上傳其他設計文件來解決資料問題清單不足的問題。

步驟 5：檢視問題清單詳細資訊

選取個別問題清單，以檢視詳細的理由和修補指引。

1. 在問題清單表格中，選取安全需求名稱。
2. 檢閱調查結果詳細資訊，其中包括：
   - 正在評估的特定安全需求
   - 說明調查結果為何收到其合規狀態的評論，包括有關遺漏或不合規內容的特定詳細資訊
   - 解決問題清單的建議修補指引
   - 連結至您組織的內部文件或安全要求標準

> **Note**
>
> 註解會說明 AWS Security Agent 的推理與特定詳細資訊。對於資料調查結果不足，註解會識別缺少哪些資訊，例如「設計文件未提及身分驗證機制」或「找不到靜態資料加密的相關資訊」。

步驟 6：地址問題清單

針對需要注意的問題清單採取動作，以改善設計的安全狀態。

對於不合規的問題清單：

1. 檢閱建議的修補指引。
2. 檢閱任何連結的內部文件以取得其他內容。
3. 更新您的設計文件以因應安全需求。
4. 記錄您所做的變更，以供日後參考。

對於資料問題清單不足：

1. 仔細閱讀註解，以了解缺少哪些特定資訊。
2. 建立或更新缺少詳細資訊的設計文件。
3. 準備更新的檔案以重新提交。

後續步驟

檢閱您的設計調查結果之後：

- 將調查結果報告下載為 CSV 檔案，以便與您的團隊共用
- 更新設計文件以解決不合規的問題清單
- 為資料問題清單不足建立其他文件
- 與您的開發團隊共用問題清單以進行討論
- 複製此設計檢閱以建立新的檢閱，其中包含預先載入的原始文件，可讓您更新名稱並再次執行分析以確認改善
- 繼續實作符合合規要求的設計

如需管理安全需求的詳細資訊，請參閱 the section called “管理安全需求”。

如需建立設計檢閱的詳細資訊，請參閱 the section called “建立設計檢閱”。

建立威脅模型

在 AWS Security Agent Web 應用程式中建立威脅模型，以分析應用程式的架構並識別安全威脅。威脅模型會產生兩個主要輸出：描述系統如何建置和運作的系統概觀，以及描述系統如何遭受攻擊的一組威脅，每個威脅都有嚴重性等級、STRIDE 分類和可行的建議。

威脅模型接受兩種類型的輸入，您可以提供 或兩者：

- 範圍文件 – 定義威脅模型重點的技術設計文件、API 規格或架構文件。提供時，代理程式會將其分析範圍限定為這些文件中所述的內容。您可以上傳檔案 (DOC、DOCX、JPEG、MD、PDF、PNG、TXT)、提供 S3 URIs，或透過整合連接 Confluence 頁面。
- 來源 – 來自連線儲存庫 (GitHub、GitHub Enterprise Server、GitLab、GitLab Self-Managed 或 Bitbucket) 或從 S3 位置的來源碼。AWS Security Agent 會讀取原始程式碼，以了解您現有的系統。與範圍文件結合時，原始程式碼會提供目前實作的相關內容。

在此程序中，您可以設定威脅模型的輸入和許可，然後開始執行。

先決條件

開始前，請確保您具備以下條件：

- 存取 AWS Security Agent Web 應用程式
- 至少下列其中一個項目：
  - 做為來源使用的連線儲存庫 (GitHub、GitHub Enterprise Server、GitLab、GitLab Self-Managed 或 Bitbucket)（請參閱 the section called “啟用威脅建模”）
  - 包含原始碼的 S3 儲存貯體
  - 設計上傳為範圍文件或 Confluence 整合的文件

> **Tip**
>
> 如果您已有儲存庫或 S3 儲存貯體連接到您的 Agent Space（例如，透過程式碼檢閱或滲透測試設定），您可以立即建立威脅模型，而不需要額外的設定。

AWS Security Agent 如何執行威脅模型

您提供的輸入會決定 AWS Security Agent 如何執行威脅模型。有三種情況。

您提供的輸入	AWS Security Agent 如何執行威脅模型
僅限來源（原始程式碼）	AWS Security Agent 會分析原始程式碼，以了解應用程式的結構、分類元件、擷取資料流程、建置威脅模型，並根據系統實際實作的方式識別威脅。
僅限範圍文件（設計文件）	AWS Security Agent 會執行著重於文件中所述設計的威脅模型。它根據範圍文件中描述的架構、資料流程和元件來識別威脅，有助於在存在任何程式碼之前對設計進行威脅建模。
來源和範圍文件	AWS Security Agent 會將威脅模型範圍限定為範圍文件中所述的內容（例如，新功能的設計文件）。它使用原始碼來了解您現有的系統，然後威脅模型範圍文件中描述的設計，無論該設計是否已實作。

存取威脅模型頁面

導覽至 Web 應用程式中的威脅建模區段。

1. 登入 AWS Security Agent Web 應用程式。
2. 在左側邊欄中，選擇威脅模型。
3. 您會看到現有威脅模型的清單，其中包含其標題、最新執行狀態，以及依嚴重性列出的威脅計數。

建立威脅模型

透過設定其輸入和許可來設定新的威脅模型。

1. 在威脅模型頁面上，選擇建立威脅模型。

設定威脅模型詳細資訊

為您的威脅模型提供標題。

1. 在標題欄位中，輸入威脅模型的描述性名稱。

> **Tip**
>
> 使用可識別要建模之應用程式或服務的名稱。例如，「billing-service」或「checkout-api」。

新增範圍文件

提供定義威脅模型重點的技術設計文件，例如架構文件、API 規格或設計提案。提供範圍文件時，代理程式會將其分析範圍限定在這些文件中描述的內容。如果您提供來源，範圍文件是選用的。

1. 在範圍文件區段中，使用下列一或多個方法新增文件：
   - 上傳檔案 – 直接上傳設計文件 (DOC、DOCX、JPEG、MD、PDF、PNG 或 TXT)。
   - S3 URIs – 輸入指向存放在已連線 S3 儲存貯體中的文件的 S3 URIs。
   - Confluence 頁面 – 如果您的 Confluence 整合已連接到您的客服人員空間，請從 Confluence 工作區中選取頁面。

> **Tip**
>
> 為了獲得最佳結果，請包含架構圖、API 規格，以及描述系統元件、資料流程和信任界限的技術文件。

新增來源

選取 AWS Security Agent 用來了解現有系統的原始碼。與範圍文件結合時，原始程式碼會提供目前實作的相關內容 — 代理程式會使用它來了解已存在的內容。如果您提供範圍文件，則來源是選用的。

1. 在來源區段中，使用下列一或多個方法新增來源碼：

整合式儲存庫

從連線至 Agent Space 的儲存庫中選取 (GitHub、GitHub Enterprise Server、GitLab、GitLab Self-Managed 或 Bitbucket)。

1. 選取您要包含為來源的每個儲存庫旁的核取方塊。
2. 使用搜尋欄位依名稱尋找特定儲存庫。

> **Note**
>
> 只有連線至 Agent Space 的儲存庫才會顯示在這裡。若要新增更多儲存庫，請要求您的管理員更新 Agent Space 組態。

S3 來源

新增指向存放在已連線 S3 儲存貯體中原始碼的 S3 URIs。

1. 輸入您要包含的每個來源碼位置的 S3 URI。

設定 AWS 資源

選取此威脅模型的 IAM 服務角色和選用的 CloudWatch 日誌群組。

1. 在服務角色下拉式清單中，從設定的服務角色中選取 IAM 角色。

> **Note**
>
> 服務角色必須具有許可，才能存取 S3 中的原始程式碼並寫入 CloudWatch 日誌。服務角色是在 AWS 管理主控台中的客服人員空間設定期間設定。

2. （選用）在日誌群組下拉式清單中，選取要存放威脅模型執行日誌的 CloudWatch 日誌群組。

建立威脅模型

1. 檢閱您的組態。
2. 選擇建立威脅模型。

系統會將您重新導向至威脅模型詳細資訊頁面，您可以在其中開始執行。

執行威脅模型

建立威脅模型後，開始執行以開始分析。

1. 在威脅模型詳細資訊頁面上，選擇開始執行。
2. AWS Security Agent 會開始分析您的來源和範圍文件。

您也可以選擇您要執行的威脅模型旁邊的開始執行，從威脅模型清單頁面開始執行。

監控威脅模型執行

追蹤威脅模型執行時的進度。

執行狀態

執行頁面標頭會顯示狀態指示燈：

- 進行中 – 威脅模型代理程式正在執行。
- 停止 – 已請求取消；代理正在停止之前完成其目前的任務。
- 已完成 – 執行已成功完成。
- 失敗 – 執行發生錯誤。錯誤訊息會顯示詳細資訊。在解決問題後啟動新的執行。
- 已停止 – 使用者已取消執行。在停止之前產生的任何威脅都會保留。

執行頁面索引標籤

在執行詳細資訊頁面上，在索引標籤之間導覽：

- 概觀 – 檢視執行摘要（任務 ID、開始時間、狀態、持續時間），其嚴重性層級圓餅圖會依嚴重性（關鍵、高、中、低）顯示威脅的明細。在摘要下方，檢視 STRIDE 威脅類別長條圖，顯示有多少威脅屬於每個 STRIDE 類別。執行完成後，檢視代理程式產生的系統概觀。
- 組態 – 檢視用於此執行的來源、文件和許可（服務角色、CloudWatch 日誌群組）。
- 預檢 – 檢視威脅分析開始之前執行的預檢狀態，包括記錄基礎設施設定、S3 來源存取驗證（如適用）和測試環境設定。進度列會顯示已完成多少檢查。
- 日誌 – 檢視代理程式在執行期間執行的可篩選任務清單。選取任何任務，以在側邊面板中檢視其詳細日誌輸出。
- 威脅 – 檢視執行期間識別的威脅（請參閱 the section called “檢閱來自威脅模型的威脅”）。

執行歷史記錄

每個威脅模型都會維護所有執行的歷史記錄。在威脅模型詳細資訊頁面上：

- 執行資料表會列出所有執行及其開始時間、狀態、持續時間、威脅計數和 ID。
- 選擇任何執行的開始時間連結，以檢視其完整詳細資訊。

> **Tip**
>
> 更新原始程式碼或修改範圍文件後，請重新執行相同的威脅模型，以查看系統概觀和威脅如何隨時間變化。

編輯威脅模型

若要修改您的威脅模型組態：

1. 在威脅模型詳細資訊頁面上，選擇編輯。
2. 視需要更新標題、來源、範圍文件或 AWS 資源。
3. 選擇儲存變更。

> **Note**
>
> 編輯威脅模型不會影響先前完成的執行。這些快照會保留執行時所使用的組態快照。

刪除威脅模型

若要刪除威脅模型及其所有相關聯的執行和威脅：

1. 在威脅模型詳細資訊頁面上，開啟動作選單，然後選擇刪除。
2. 確認刪除。

> **Important**
>
> 如果執行目前正在進行中，您必須在刪除威脅模型之前將其停止。

後續步驟

執行威脅模型之後：

- 檢閱系統概觀和威脅（請參閱 the section called “檢閱來自威脅模型的威脅”）
- 在進行變更以確認威脅已解決之後，重新執行威脅模型

- 隨著應用程式的演進，調整您的來源和範圍文件

檢閱來自威脅模型的威脅

威脅模型執行完成後，請檢閱系統概觀和威脅，以了解您的應用程式如何遭到攻擊以及如何處理。系統概觀是描述應用程式架構、信任界限、資料流程和安全狀態的完整文件。每個威脅都包含陳述式、嚴重性等級、STRIDE 分類、受影響的資產，以及解決該威脅的建議。

先決條件

開始前，請確保您具備以下條件：

- 已完成的威脅模型執行
- 存取 AWS Security Agent Web 應用程式

步驟 1：存取威脅模型執行

導覽至您已完成的威脅模型執行。

1. 登入 AWS Security Agent Web 應用程式。
2. 在左側邊欄中，選擇威脅模型。
3. 選取您要檢查的威脅模型。
4. 在執行資料表中，選擇其開始時間連結，以選取已完成的執行。

步驟 2：檢閱系統概觀

系統概觀是 AWS Security Agent 如何了解您系統的完整說明。這是結構化文件，可包含：

- 目的 – 系統做什麼，以及服務對象。
- 功能 – 系統提供的主要功能。
- 設計意圖 – 正在建模的設計變更或功能（提供範圍文件時）。
- 架構 – 系統的建置方式，包括部署模式和通訊協定。
- 元件 – 具有其用途和金鑰互動的系統元件資料表。
- 信任界限 – 安全內容變更的地方，包括每個交叉存在的保護。
- 資料流程 – 詳細描述資料如何在系統中移動，包括每個步驟的通訊協定、登入資料和保護。
- 安全狀態 – 目前的身分驗證、加密和存取控制機制。

- 敏感資產 – 需要保護的資料和登入資料，以及其分類和公開點。
- 關鍵假設 – 客服人員對系統所做的安全相關假設。

若要檢閱系統概觀：

1. 選取概觀索引標籤。
2. 檢閱執行摘要區段，其中顯示任務 ID、開始時間、狀態和持續時間。
3. 檢閱嚴重性等級圖表，其中顯示依嚴重性（關鍵、高、中、低）分類的威脅明細，其中包含計數和百分比。
4. 檢閱威脅類別圖表，其中顯示有多少威脅屬於每個 STRIDE 類別（詐騙、竄改、複寫、資訊揭露、拒絕服務、提升權限）。
5. 在系統概觀區段中，讀取客服人員的完整分析。

> **Tip**
>
> 如果系統概觀無法準確反映您的系統，請精簡您的輸入，將相關儲存庫新增為來源或上傳更完整的範圍文件，然後再次執行威脅模型。

步驟 3：檢閱威脅

導覽至威脅索引標籤，以檢視執行期間識別的所有威脅。

1. 選取威脅索引標籤。
2. 威脅會以清單形式顯示，其中每張卡片都顯示威脅標題、嚴重性徽章、狀態和上次更新的時間戳記。您可以依嚴重性、狀態或依標題搜尋來篩選威脅。
3. 從清單中選擇威脅，在右側面板中檢視其完整詳細資訊。

威脅嚴重性

每個威脅都會指派一個嚴重性等級：

- 關鍵 – 需要立即採取行動；利用可能會導致完整的系統入侵。
- 高 – 需要立即注意；利用可能會導致重大的安全影響。
- 中 – 應在合理的時間範圍內解決；造成整體安全風險。

- 低 – 可以作為定期維護的一部分來處理；將立即風險降至最低。

威脅詳細資訊

選取威脅以在右側面板中檢視其詳細資訊。詳細資訊會整理為下列各節：

中繼資料列：

- 嚴重性 – 代理程式指派的風險層級（關鍵、高、中或低）。
- STRIDE 類別 – 威脅分類：詐騙、竄改、複寫、資訊揭露、拒絕服務或提升權限。
- 在 建立 – 識別威脅時。
- 上次更新 – 上次修改威脅的時間。

描述區段：

- 陳述式 – 威脅的自然語言描述：威脅來源可以做什麼、影響是什麼，以及啟用它的條件。
- 來源 – 威脅的演員或來源（例如，「已驗證的使用者」或「外部攻擊者」）。
- 動作 – 威脅來源可以執行的動作（例如，「將 SQL 查詢插入搜尋參數」）。
- 影響 – 威脅動作的直接結果（例如，「未經授權存取客戶記錄」）。

參考區段：

- 受影響的資產 – 受威脅影響的特定資產（例如「客戶付款記錄」或「DynamoDB 資料表」）。
- 先決條件 – 必須為 true 才能利用威脅的條件。
- 受影響的目標 – 受影響的安全目標：機密性、完整性、可用性、授權、身分驗證或不可否認。
- 證據 – 支援威脅的原始程式碼檔案路徑，可連結回儲存庫中的特定檔案。
- 錨點 – 將威脅連結到原始程式碼中特定節點的程式碼參考。

建議區段：

- 建議 – 解決威脅的可行指引。

步驟 4：手動建立威脅

您可以新增代理程式無法識別的威脅，例如，手動檢閱期間或從外部來源發現的威脅。

1. 在已完成執行的威脅索引標籤上，選擇建立威脅。
2. 填寫威脅詳細資訊：
   - 陳述式 – 威脅的自然語言描述。
   - 嚴重性 – 選取關鍵、高、中或低。
   - 來源 – 威脅的演員或原始伺服器。
   - 先決條件 – 威脅可供利用所需的條件。
   - 動作 – 威脅來源可以執行的動作。
   - 影響 – 威脅動作的直接後果。
   - 受影響的資產 – 受影響的特定資產（逗號分隔）。
   - 受影響的安全目標 – 從機密性、完整性、可用性、授權、身分驗證或不可否認中選擇。
   - STRIDE 類別 – 選取適用的類別。
   - 建議 – 解決威脅的指引。
3. 選擇建立。

手動建立的威脅與客服人員產生的威脅一起顯示在威脅清單中。

步驟 5：編輯和分類威脅

當您檢閱威脅時，您可以編輯其詳細資訊並更新其狀態以追蹤進度。

1. 從清單中選擇威脅。
2. 選擇威脅詳細資訊面板中的編輯圖示，以開啟編輯表單。
3. 您可以修改下列欄位：
   - 狀態 – 追蹤威脅生命週期：
     - 開啟 – 已確認威脅並需要注意（預設）。
     - 已解決 – 您已修正此問題。
     - 已解除 – 您已檢閱威脅，並判斷其不適用。
   - 嚴重性 – 如果客服人員的評估不符合您的內容，請調整嚴重性等級。
   - 陳述式 – 精簡威脅描述。
   - 來源、先決條件、動作、影響 – 根據您的網域知識更新威脅詳細資訊。
   - 受影響的資產 – 新增或移除受影響的資產（以逗號分隔）。
   - 受影響的安全目標 – 選取受影響的安全目標（機密性、完整性、可用性、授權、身分驗證、不可否認）。

4. 選擇儲存來套用您的變更。

步驟 6：下載報告

執行完成後，您可以下載摘要系統概觀和所有已識別威脅的 PDF 報告。

1. 在已完成的執行頁面上，選擇產生報告。
2. PDF 下載到您的電腦。

步驟 7：檢閱預檢檢查和日誌

如果您需要調查客服人員如何得出結論或偵錯部分失敗：

1. 選取預檢標籤，以檢視威脅分析開始之前執行的預檢檢查狀態。每個檢查會顯示其狀態（完成、進行中或待定），進度列則指出已完成的檢查數量。您可以展開已完成的檢查，以檢視其詳細日誌輸出。
2. 選取日誌索引標籤，以檢視代理程式在執行期間執行的可篩選任務清單。從清單中選取任何任務，以在側邊面板中檢視其詳細日誌輸出。

後續步驟

檢閱您的威脅模型結果之後：

- 先根據客服人員的建議解決高嚴重性的威脅
- 在您實作修正時更新威脅狀態
- 執行新的威脅模型，以驗證您的變更是否解決已識別的威脅
- 隨著應用程式的演進，調整您的來源和範圍文件（請參閱 the section called “建立威脅模型”）

檢閱提取請求中的程式碼安全性問題清單

為您的儲存庫啟用提取請求的程式碼檢閱後，AWS Security Agent 會自動分析提取請求，並直接在來源控制提供者中發佈安全調查結果。這可讓開發人員在正常工作流程中解決安全問題，而無需離開提取請求。

> **Note**
>
> 此頁面適用於 GitHub 提取請求、GitLab 合併請求和 Bitbucket 提取請求。所有供應商的體驗都類似。

程式碼檢閱在提取請求中的運作方式

當您在啟用程式碼檢閱的儲存庫中提交提取請求（或在 GitLab 中合併請求）時，AWS Security Agent 會自動開始分析。

1. 提取請求分析觸發 - 當您在已啟用程式碼檢閱功能的儲存庫中將提取請求標記為「準備檢閱」時，就會觸發程式碼檢閱。草稿提取請求不會進行分析。
2. 分析確認 - 當 AWS Security Agent 開始分析提取請求時，它會發佈初始評論：「AWS Security Agent 正在分析您的程式碼...」這可讓您知道分析已開始並正在進行中。
3. 檢閱完成 - 分析完成後，AWS Security Agent 會將檢閱與結果一起發佈到您的提取請求。所有安全性調查結果都會在單一檢閱中批次處理，以保持提取請求井然有序，並將通知降至最低。

了解程式碼檢閱結果

AWS Security Agent 會根據其在分析期間找到的結果，提供不同類型的結果。

發現安全問題時

如果 AWS Security Agent 在程式碼變更中發現安全問題，它會發佈評論，其中包含：

- 摘要 - 所有安全性調查結果的高階概觀，說明已識別的問題類型及其潛在影響
- 個別問題清單 - 詳細的安全性問題清單會在主要檢閱下顯示為執行註解，每個問題清單包括：
  - 安全問題的描述
  - 在您的程式碼中發現問題的位置
  - 說明如何解決問題的修補指引
  - 根據您的程式碼檢閱設定的相關內容（安全需求違規、常見漏洞或兩者）

> **Note**
>
> 分析的安全性問題類型取決於您的程式碼檢閱設定。如果您已設定安全需求驗證，問題清單會參考組織的自訂安全需求。如果您設定了安全漏洞調查結果，調查結果將識別常見的安全漏洞。如需程式碼檢閱設定的詳細資訊，請參閱 the section called “啟用 GitHub 儲存庫的提取請求程式碼檢閱”。

未發現安全問題時

如果 AWS Security Agent 完成分析並在程式碼中沒有發現任何安全性問題，它會發佈評論：「沒有發現問題」。這可確認檢閱已成功完成，而且您的程式碼變更不會根據您設定的程式碼檢閱設定觸發任何安全性問題清單。

回應安全性問題清單

檢閱 AWS Security Agent 發佈的安全調查結果後，您可以直接在來源控制提供者中採取動作。

- 地址調查結果 - 根據調查結果中提供的修補指引更新您的程式碼，然後將新的遞交推送至提取請求。AWS Security Agent 會分析更新的程式碼。
- 解決對話 - 解決安全問題清單後，將對話標記為已解決，以追蹤您的進度。

> **Tip**
>
> 每個調查結果都包含針對所識別安全問題量身打造的特定修補指引。請仔細檢閱本指南，以了解安全風險以及如何有效解決風險。

篩選程式碼檢閱問題清單

您可以自訂 AWS Security Agent 如何透過將`filtering.md`檔案新增至儲存庫來分析程式碼。此檔案可讓您提供程式碼庫的相關內容，並從分析中排除檔案或資料夾，以減少誤報。

建立篩選檔案

在儲存庫根`filtering.md`目錄的 `.awssecurityagent` 目錄中建立名為 的檔案：

```
.awssecurityagent/filtering.md
```

分析提取請求時，AWS Security Agent 會從儲存庫的主分支讀取此檔案（例如 main或 mainline)。

檔案結構

filtering.md 檔案使用標準 Markdown 格式搭配 AWS Security Agent 辨識的特定區段。檔案必須包含Code Review標題，後面接著下列其中一個或兩個區段： IgnorePatterns和 ContextHints（無空格）。

下列範例顯示filtering.md檔案的完整結構：

```
# filtering.md

## Code Review

### IgnorePatterns

**/*.md

/myapp/src/**/*.snap

/myapp/config/README

### ContextHints

- The backend is a trusted system and won't return non-standard protocols.
- URL is generated from server with presigned token, so no SSRF security
 vulnerabilities.
- AppSec has verified that we are allowed to use cache with an eviction policy.
```

忽略模式

IgnorePatterns 本節指定 AWS Security Agent 在程式碼檢閱期間應略過的檔案和資料夾。使用 glob patterns 定義要從分析中排除的路徑。

格式需求：

- 每個模式都必須位於自己的一行。
- 以空行分隔每個模式。這可確保檔案在 GitHub 或程式碼檢閱工具中檢視時能正確轉譯。
- 模式遵循標準 glob 格式。例如， **/*.md會比對所有 Markdown 檔案，並/myapp/src/**/*.snap比對根/myapp/src/資料夾內的所有.snap檔案。

- 本節支援最多 1000 個忽略模式。

內容提示

`ContextHints` 本節提供有關程式碼庫的其他內容，可協助 AWS Security Agent 進行更準確的評估。使用內容提示來說明可能影響問題清單解譯方式的架構決策、安全例外狀況或其他資訊。

格式需求：

- 每個提示必須以破折號 (-) 開頭，後面接著空格。
- 以單行自由格式文字撰寫每個提示，限制為 500 個字元。
- 每個提示都應描述有關程式碼庫的一個特定內容。
- 本節支援最多 20 個內容提示。

內容提示會在 AWS Security Agent 完成其初始分析後套用，協助篩選不適用於您特定使用案例的問題清單。

後續步驟

檢閱程式碼安全性問題清單之後：

- 根據修補指引更新您的程式碼
- 推送新遞交以觸發變更的重新分析
- 視需要調整程式碼檢閱設定（請參閱 the section called “啟用 GitHub 儲存庫的提取請求程式碼檢閱”）
- 檢閱組織的安全需求，以了解驗證條件
- 考慮滲透測試，以全面驗證已部署應用程式的安全性

建立程式碼檢閱

在 AWS Security Agent Web 應用程式中建立程式碼檢閱，以掃描原始程式碼儲存庫和 S3 來源是否有安全漏洞。程式碼檢閱會在整個程式碼庫中執行全面的靜態分析，識別安全問題並提供修補指導。

與分析個別程式碼變更的提取請求型程式碼檢閱不同（請參閱the section called “檢閱提取請求中的程式碼安全性問題清單”），隨需程式碼檢閱會掃描完整的原始程式碼，以識別安全漏洞並驗證是否符合組織的安全需求。

在此程序中，您將透過選取原始程式碼輸入、設定許可和執行檢閱來建立程式碼檢閱。

先決條件

開始前，請確保您具備以下條件：

- 存取 AWS Security Agent Web 應用程式
- 代理程式空間中至少有一個連接的 GitHub 儲存庫或 S3 儲存貯體

> **Tip**
>
> 如果您已將 GitHub 儲存庫連接到您的 Agent Space，則程式碼檢閱已就緒可供使用 — 不需要額外的設定。從 Agent Space 頁面上的程式碼檢閱卡片選擇在 Web 應用程式中啟動，或直接啟動 Web 應用程式。
> 如果您需要連接其他來源或設定 S3 儲存貯體，請參閱 the section called “啟用程式碼檢閱”。

存取程式碼檢閱頁面

導覽至 Web 應用程式中的程式碼檢閱區段。

1. 登入 AWS Security Agent Web 應用程式。
2. 在左側邊欄中，選擇程式碼檢閱。
3. 您會看到現有程式碼檢閱的清單，其中包含其來源資訊、上次執行狀態和調查結果摘要。

建立程式碼檢閱

透過設定其原始程式碼輸入和許可來設定新的程式碼檢閱。

1. 在程式碼檢閱頁面上，選擇建立程式碼檢閱。

設定程式碼檢閱詳細資訊

提供標題，然後選取要檢閱的原始程式碼。

1. 在標題欄位中，輸入程式碼檢閱的描述性名稱。

> **Tip**
>
> 使用可識別應用程式、儲存庫或檢閱範圍的名稱。例如，「billing-service-security-review」或「infrastructure-code-audit」。

2. 在來源區段中，選取此檢閱的來源碼輸入。從兩個索引標籤中選擇：

GitHub 儲存庫

從連線至 Agent Space 的儲存庫中選取 。

1. 選擇 GitHub 儲存庫索引標籤。
2. 在整合式儲存庫資料表中，選取您要包含在檢閱中的每個儲存庫旁的核取方塊。
3. 使用搜尋欄位依名稱尋找特定儲存庫。

> **Note**
>
> 只有透過程式碼檢閱組態連線至 Agent Space 的儲存庫才會顯示在這裡。若要新增更多儲存庫，請在管理員主控台中選擇管理，或要求管理員更新客服人員空間組態。

S3 來源

從連線至客服人員空間的 S3 儲存貯體中選取 ZIP 檔案。您的 Agent Space 管理員會設定可用的 S3 儲存貯體。存放在其中一個儲存貯體中的任何 ZIP 檔案都可以用作程式碼檢閱的來源。

1. 選擇 S3 來源索引標籤。
2. 輸入您要包含在檢閱中每個 ZIP 檔案的 S3 URI。您最多可以新增 30 個 S3 來源。

> **Note**
>
> S3 來源必須是存放在連線至 Agent Space 的 S3 儲存貯體中的 ZIP 檔案。若要讓其他儲存貯體可用，請參閱 the section called “啟用程式碼檢閱”。

設定 許可

選取此程式碼檢閱的 IAM 服務角色和選用 CloudWatch 日誌群組。

1. 在許可區段中，找到服務角色下拉式清單。
2. 從設定的服務角色中選取 IAM 角色。

> **Note**
>
> 服務角色必須具有許可，才能存取 S3 中的原始碼並寫入 CloudWatch 日誌，以及程式碼檢閱所需的任何其他 AWS 資源。服務角色是在 AWS 管理主控台中的程式碼檢閱設定期間設定。

3. （選用）在 CloudWatch 日誌群組下拉式清單中，選取要存放程式碼檢閱執行日誌的日誌群組。

> **Note**
>
> 如果您未選取日誌群組，AWS Security Agent 會建立預設日誌群組來存放程式碼檢閱日誌。

設定自動程式碼修復

啟用自動修復，讓 AWS Security Agent 在檢閱完成後立即為所有問題清單產生程式碼修正。

1. 在自動程式碼修復區段中，選取啟用自動程式碼修復核取方塊。

AWS Security Agent 如何提供修正取決於來源：

- 私有 GitHub 儲存庫 – AWS Security Agent 會將具有修正的提取請求提交至儲存庫。
- 公有 GitHub 儲存庫 – 為了避免在漏洞修正之前揭露漏洞，AWS Security Agent 不會開啟提取請求。反之，它會將建議的 diff 附加到您可以從 Web 應用程式下載的調查結果，並私下套用。
- S3 來源 – 程式碼修復無法使用。檢閱問題清單詳細資訊並手動套用修正。

> **Important**
>
> 具有讀取存取權的每個人都可看見提交至私有儲存庫的修補提取請求。在合併之前檢閱變更。只有在選取 GitHub 儲存庫做為來源時，才能使用自動程式碼修復。

> **Note**
>
> 停用時，您仍可在檢閱完成後手動觸發個別 GitHub 來源問題清單的程式碼修復。

設定模擬驗證

啟用模擬驗證，以動態確認在執行中的應用程式中是否可利用探索到的漏洞。

1. 在模擬驗證區段中，選取啟用模擬驗證核取方塊。

啟用時，AWS Security Agent 會在靜態分析完成後佈建模擬環境。它會加入您的原始碼、啟動環境中的服務，並嘗試利用掃描期間發現的漏洞。然後，調查結果會更新為驗證狀態，指出漏洞是否已成功利用。

> **Note**
>
> 模擬驗證目前僅適用於獨立的 Dockerizable 應用程式。選取多個儲存庫做為來源時，無法使用模擬驗證。

> **Important**
>
> 模擬驗證會將處理時間新增至程式碼檢閱執行。驗證步驟會佈建環境並執行入侵嘗試。這通常需要 1-3 小時，取決於調查結果數量和應用程式複雜性。

允許的網路目的地

模擬環境具有僅限於預先定義允許清單的傳出網路存取。您的應用程式可以在驗證期間到達下列網域：

下表列出允許的網域及其用途。

網域	用途
.amazonaws.com ，.aws.amazon.com	AWS 服務
.public.ecr.aws	Amazon ECR Public
.docker.com ，.docker.io	Docker Hub
.github.com ，.githubusercontent.com	GitHub
.gitlab.com	GitLab
.npmjs.com ，.npmjs.org	npm 登錄檔
.pypi.org ，.pypi.python.org ，.pythonhosted.org	Python 套件索引
.crates.io ，.rustup.rs	Rust 套件
.maven.org ，.gradle.org	Java/Gradle 套件
.nuget.org	.NET 套件
.rubygems.org ，.ruby-lang.org	Ruby 套件
.golang.org ，.pkg.go.dev ，.goproxy.io	Go 套件
.nodejs.org ，.yarnpkg.com	Node.js

網域	用途
.alpinelinux.org , .debian.org , .ubuntu.com , .centos.org , .fedoraproject.org	Linux 發行版本儲存庫
.cloudfront.net	CloudFront 分佈
.google.com , .googleapis.com	Google APIs
.microsoft.com , .visualstudio.com	Microsoft 服務
.sourceforge.net , .bitbucket.org	來源託管

> **Note**
>
> 如果您的應用程式需要存取此清單中未列出的網域，網路防火牆會封鎖連線。依賴以上未列出的外部 APIs或服務的應用程式可能無法在模擬環境中正確啟動。

建立程式碼檢閱

1. 檢閱您的組態以確保準確性。
2. 選擇建立程式碼檢閱。

系統會將您重新導向至程式碼檢閱詳細資訊頁面，您可以在其中開始檢閱執行。

執行程式碼檢閱

建立程式碼檢閱後，開始執行以開始分析。

1. 在程式碼檢閱詳細資訊頁面上，選擇開始檢閱。
2. AWS Security Agent 會開始分析您的原始程式碼。

您也可以從程式碼檢閱清單頁面開始檢閱，方法是選擇您要執行的程式碼檢閱旁的開始檢閱。

監控程式碼檢閱執行

追蹤程式碼檢閱執行時的進度。

檢閱執行階段

程式碼檢閱執行會進行下列階段，顯示為進度指標：

1. 預檢 – AWS Security Agent 會驗證對原始碼的存取，並設定測試環境。預檢檢查包括：
   - 服務基礎設施設定
   - S3 來源存取驗證
   - 設定測試環境
2. 靜態分析 – AWS Security Agent 會掃描您的原始程式碼是否有安全漏洞和違反要求的情況。
3. 模擬驗證（選用）– 如果啟用模擬驗證，AWS Security Agent 會佈建模擬環境並部署您的應用程式。然後，它會嘗試利用靜態分析期間發現的漏洞。此步驟會確認問題清單是否可在目標執行期中利用。只有當您在程式碼檢閱建立期間啟用模擬驗證時，才會顯示此階段。
4. 完成 – AWS Security Agent 會編譯問題清單並產生結果摘要。

檢視執行詳細資訊

在執行詳細資訊頁面上，在標籤之間導覽以監控進度：

- 程式碼檢閱執行 – 檢視執行摘要，包括執行 ID、建立時間、狀態、持續時間、任務時數、嚴重性等級明細和風險類型圖表。
- 預檢 – 檢視每個驗證步驟的預檢檢查進度和狀態。
- 程式碼檢閱日誌 – 檢視 AWS Security Agent 在檢閱期間識別和執行的任務，以及每個步驟的詳細任務日誌。
- 模擬驗證 – 啟用模擬驗證時，檢視佈建狀態、個別問題清單的驗證任務及其結果。
- 問題清單 – 檢視審核完成後的安全性問題清單（請參閱 the section called “檢閱程式碼檢閱的問題清單”）。

執行歷史記錄

每個程式碼檢閱都會維護所有執行的歷史記錄。在程式碼檢閱詳細資訊頁面上：

- 最新執行區段會顯示最新執行及其開始時間、狀態、持續時間和 ID。
- 所有執行資料表會列出所有先前的執行及其開始時間、狀態、持續時間、調查結果摘要和 ID。
- 選擇監控執行以檢視最新作用中執行的詳細資訊。
- 選擇任何執行的開始時間連結，以檢視其完整詳細資訊。

後續步驟

執行程式碼檢閱之後：

- 檢閱安全性問題清單及其修補指引（請參閱 the section called “檢閱程式碼檢閱的問題清單”）
- 透過自動提取請求或手動修正來修正問題清單（請參閱 the section called “修正程式碼檢閱問題清單”）
- 在實作修正以驗證修復之後執行其他檢閱
- 隨著程式碼庫的演進，調整程式碼檢閱組態或來源

檢閱程式碼檢閱的問題清單

程式碼檢閱執行完成後，請檢閱執行摘要和安全性調查結果，以了解原始程式碼中的漏洞。每個調查結果都包含描述、嚴重性評分、程式碼位置、風險推理和建議的修正，以協助您排定優先順序並解決安全問題。

先決條件

開始前，請確保您具備以下條件：

- 已完成或進行中程式碼檢閱執行
- 存取 AWS Security Agent Web 應用程式

步驟 1：存取程式碼檢閱執行

導覽至您已完成的程式碼檢閱執行，以檢視摘要和調查結果。

1. 登入 AWS Security Agent Web 應用程式。
2. 在左側邊欄中，選擇程式碼檢閱。
3. 選取您要檢查的程式碼檢閱。

4. 在所有執行表中，按一下其開始時間連結來選取已完成的執行。或者，選擇監控執行以檢視最新的執行。

步驟 2：監控執行進度

使用步驟指示器追蹤程式碼檢閱執行的進度。

1. 找到頁面標頭下方的水平步驟指標。
2. 檢閱每個階段的狀態：
   - 預檢 – 驗證對原始程式碼的存取，並設定掃描環境。檢查包括服務基礎設施設定、S3 來源存取驗證和測試環境設定，其中包括 GitHub 存取檢查。
   - 靜態分析 – 掃描您的原始程式碼是否有安全漏洞和違反需求的情況。
   - 模擬驗證（選用）– 啟用時， 會佈建模擬環境，並嘗試針對執行中的應用程式利用探索到的漏洞。
   - 完成 – 編譯問題清單並產生結果摘要。

> **Note**
>
> 每個步驟都會顯示狀態指示燈（已完成或進行中）。當所有階段都顯示已完成時，執行即完成。模擬驗證步驟只會在您於程式碼檢閱建立期間啟用模擬驗證時顯示。

步驟 3：檢閱執行摘要

導覽至程式碼檢閱執行索引標籤以檢視高階結果。

1. 選取程式碼檢閱執行索引標籤。
2. 執行摘要區段提供執行狀態、持續時間和其他高階詳細資訊。它還提供按嚴重性等級和風險類型分類的安全調查結果儀表板。
3. AWS Security Agent 的應用程式概觀會在執行完成時提供程式碼檢閱掃描的摘要

步驟 4：檢閱預檢結果

導覽至預檢索引標籤，以驗證所有來源存取檢查是否通過。

1. 選取飛行前索引標籤。

2. 檢閱預檢進度指示器，顯示已完成的檢查數量。
3. 確認每個檢查都顯示成功狀態：
   - 服務基礎設施設定 – 確認測試環境已就緒
   - S3 來源存取驗證 – 確認對 S3 來源碼的存取 （ 如適用 ）
   - 設定測試環境 – 確認已設定分析環境

> **Note**
>
> 如果任何預檢檢查失敗，執行將不會繼續進行靜態分析。檢閱檢查詳細資訊以識別和解決存取問題，然後開始新的執行。

步驟 5：檢閱程式碼檢閱日誌

導覽至程式碼檢閱日誌索引標籤，以檢視 AWS Security Agent 在分析期間執行的任務。

1. 選取程式碼檢閱日誌索引標籤。
2. 瀏覽左側面板中的任務清單。
3. 選取任務以在右側面板中檢視其詳細日誌。

> **Tip**
>
> 使用搜尋欄位依名稱篩選任務。日誌可讓您深入了解 AWS Security Agent 檢查的內容，以及它如何得出結論。

步驟 6：導覽至問題清單

選取問題清單索引標籤，以檢視執行中的所有安全性問題清單。

> **Note**
>
> 預設可信度篩選條件

根據預設，您只會看到具有高客服人員可信度的問題清單。若要顯示具有中或低客服人員可信度和誤報的問題清單，請關閉隱藏未驗證的問題清單切換。

1. 選取問題清單索引標籤。
2. 調查結果會顯示在分割檢視中，結果清單位於左側，所選調查結果的詳細資訊則位於右側。
3. 檢閱每個問題清單卡片上顯示的資訊：
   - 問題清單標題 – 概述安全問題的描述性名稱
   - 嚴重性徽章 – 顏色編碼嚴重性指標：
     - 嚴重（紅色）– 需要立即採取動作；利用可能會導致系統入侵
     - 高（紅色）– 需要立即注意；利用可能會導致重大的安全影響
     - 中（橘色）– 應在合理的時間範圍內解決；造成整體安全風險
     - 低（黃色）– 可作為定期維護的一部分處理；將立即風險降至最低
   - 上次更新 – 問題清單上次更新的時間戳記

步驟 7：檢閱問題清單詳細資訊

選取個別問題清單，以檢視每個漏洞的完整資訊。

1. 在左側面板中選取問題清單，以在右側面板中顯示其詳細資訊。
2. 檢閱可用的動作：
   - 解決問題清單 – 在您解決問題清單之後，將問題清單標記為已解決
   - 修正程式碼 – 使用修正產生提取請求（適用於 GitHub 來源）
3. 使用此調查結果是否準確提供意見回饋？ – 選取是或否，以協助改善未來的分析準確性。
4. 檢閱概觀區段中的金鑰屬性：
   - 客服人員可信度 – AWS Security Agent 在此調查結果中的可信度層級
   - 嚴重性 – 顏色編碼徽章的風險等級
   - 風險類型 – 安全風險的類別
   - 驗證狀態 – 啟用模擬驗證時，指出調查結果是否已在執行環境中確認可被利用：
     - 已確認 – 已在模擬環境中成功利用漏洞
     - 未重現 – 無法在目標執行時間利用漏洞
     - 驗證失敗 – 驗證嘗試因錯誤而不確定

    - 未驗證 – 未對此調查結果執行模擬驗證。當未啟用驗證或在到達此調查結果之前驗證步驟逾時時，就會發生這種情況。
  - 已解決 – 問題清單是否已解決（是/否）
  - 上次更新 – 最近更新的時間戳記
5. 展開描述區段以讀取：
  - 安全問題的詳細說明
  - 如何利用漏洞
  - 對應用程式的潛在影響
  - 驗證代理程式執行以確認問題清單的步驟
6. 展開程式碼位置區段以檢視：
  - 發現問題的特定檔案路徑
  - 在每個位置找到內容的簡短描述
  - 連結至相關程式碼的行號徽章
7. 展開證據區段以檢閱：
  - 支援調查結果的特定程式碼、組態或觀察
  - 簡短說明每個項目顯示漏洞的原因
  - 每個證據受影響的檔案和行號
8. 展開建議的修正區段以檢視：
  - AWS Security Agent 建議修正調查結果的變更
  - 每個建議變更的範例程式碼或組態

> **Tip**
>
> 使用程式碼位置快速導覽至儲存庫中受影響的檔案。每個位置都包含足夠的內容來了解問題，而無需讀取整個檔案。

步驟 8：排定問題清單的優先順序並加以解決

驗證問題清單並採取行動，以修復漏洞並改善應用程式的安全狀態。

對於具有高客服人員可信度的高嚴重性問題清單：

1. 徹底檢閱描述和程式碼位置區段。

2. 使用修復程式碼來產生具有修正的提取請求，或者如果您啟用該選項，則檢閱自動修復 PRs。
3. 規劃後續程式碼檢閱執行，以確認修正是否有效。

對於具有高客服人員可信度的中等嚴重性問題清單：

1. 根據您的風險承受能力和業務環境來排定優先順序。
2. 在定期開發衝刺規劃中包含修復任務。
3. 考慮多個中嚴重性調查結果是否共同造成較高的風險。

對於中或低可信度調查結果：

1. 檢閱描述和程式碼位置區段，以驗證漏洞發生的假設和條件。
2. 如果漏洞有效，請根據嚴重性和風險承受能力排定優先順序，並考慮修復任務。

步驟 9：追蹤修復進度

使用問題清單界面並重新執行 ，以追蹤已解決的漏洞。

1. 當您實作修正時，請使用解析問題清單將問題清單標記為已定址。
2. 執行新的程式碼檢閱，以確認修正可解決問題清單，而不會引入新問題。
3. 檢閱程式碼檢閱詳細資訊頁面上的所有執行資料表，以比較一段時間內跨執行的問題清單。

> **Tip**
>
> 合併修補提取請求之後，請啟動相同程式碼檢閱的新執行，以確認漏洞已解決。比較執行之間的問題清單計數，以追蹤您的進度。

後續步驟

檢閱您的程式碼檢閱問題清單之後：

- 以高可信度優先處理高嚴重性的問題清單，以便立即修復
- 使用修復程式碼為 GitHub 來源的問題清單產生自動修正 （ 請參閱 the section called “修正程式碼檢閱問題清單”）

- 在實作修正以驗證修復之後執行其他程式碼檢閱
- 隨著程式碼庫的演進，調整程式碼檢閱來源或設定（請參閱 the section called “啟用程式碼檢閱”）

修正程式碼檢閱問題清單

從程式碼檢閱檢閱安全性問題清單後，您可以使用 AWS Security Agent 為 GitHub 儲存庫來源中的問題清單產生程式碼修正。對於私有儲存庫，AWS Security Agent 會開啟包含建議修正的提取請求。對於公有儲存庫，AWS Security Agent 會將可下載的 diff 附加到您可以在本機套用的調查結果。來自 S3 來源的問題清單必須手動修復。

先決條件

開始前，請確保您具備以下條件：

- 包含問題清單的完整程式碼檢閱執行
- 做為程式碼檢閱來源連線的 GitHub 儲存庫
- 存取 AWS Security Agent Web 應用程式
- 熟悉應用程式的架構和安全性需求

程式碼修補的運作方式

當您觸發問題清單的程式碼修復時，AWS Security Agent 會分析問題清單及其程式碼位置，然後產生程式碼修正。修正的交付方式取決於來源：

- 私有 GitHub 儲存庫 – AWS Security Agent 向相關儲存庫提交提取請求。提取請求包含安全問題和所做的變更的描述。
- 公有 GitHub 儲存庫 – AWS Security Agent 會將建議的 diff 連接到調查結果，以便在修正之前不會公開漏洞。您可以從 Web 應用程式下載 diff，並使用 在本機套用`git apply /path/to/code_remediation_changes.diff`。
- S3 來源 – 程式碼修復無法使用。使用調查結果的描述、程式碼位置和風險推理來手動套用修正。

> **Important**
>
> 具有儲存庫讀取存取權的所有使用者都可看見 AWS Security Agent 建立的提取請求。在合併之前檢閱變更，以確保它們符合您應用程式的需求。

自動程式碼修復

如果您在建立程式碼檢閱時啟用自動程式碼修復，AWS Security Agent 會在檢閱完成後，立即為所有合格的問題清單產生修正。您不需要採取任何其他動作 — 提取請求會出現在您連線的私有 GitHub 儲存庫中，差異會出現在公有 GitHub 儲存庫的問題清單上，因為執行會完成。

手動程式碼修復

如果停用自動程式碼修復，您可以從 GitHub 來源觸發個別問題清單的修復。

1. 導覽至已完成程式碼檢閱執行的調查結果索引標籤。
2. 選取您要修復的問題清單。
3. 在調查結果詳細資訊面板中，選擇修正程式碼。
4. AWS Security Agent 會產生程式碼修正，並根據儲存庫的可見性，將提取請求提交至儲存庫，或將可下載的差異附加至調查結果。

檢閱修復提取請求

AWS Security Agent 向私有 GitHub 儲存庫提交修補提取請求後：

1. 導覽至您的 GitHub 儲存庫。
2. 尋找 AWS Security Agent 建立的提取請求。
3. 檢閱變更，包括：
   - 說明安全性問題清單和修正的說明
   - 解決漏洞的程式碼變更
   - 修復方法的任何相關內容
4. 如果修正是適當的，請合併提取請求，或將其關閉並實作替代解決方案。

> **Tip**
>
> 合併修補提取請求後，啟動新的程式碼檢閱執行，以確認修正可解決問題清單，而不會引入新的安全問題。

套用修復差異

對於公有 GitHub 儲存庫的問題清單，請在本機套用可下載的 diff。

1. 在調查結果詳細資訊面板中，從程式碼修復區段下載 diff。
2. 在儲存庫的本機複製中，執行 `git apply /path/to/code_remediation_changes.diff`。
3. 檢閱套用的變更，根據您的應用程式進行測試，並透過正常的開發工作流程遞交變更。

限制

- 程式碼修復僅適用於來自 GitHub 儲存庫來源的問題清單。來自 S3 來源的問題清單必須手動修復。
- AWS Security Agent 會根據其對漏洞的分析產生修正。在合併或遞交變更之前，請檢閱所有變更，以確保它們適合您的應用程式。
- 有些複雜的問題清單可能需要自動修正以外的手動介入。

後續步驟

修復問題清單之後：

- 檢閱和合併 GitHub 儲存庫中的提取請求，或透過正常工作流程遞交套用的差異
- 執行新的程式碼檢閱，以驗證修正並檢查是否有剩餘問題
- 在確認修復之後，解決 Web 應用程式中的問題清單
- 視需要調整程式碼檢閱來源或設定（請參閱 the section called “啟用程式碼檢閱”）

使用 S3 執行差異碼掃描

執行差異（差異）程式碼掃描，以僅分析原始程式碼中變更的行，而不是執行完整的儲存庫掃描。差異掃描比完整掃描更快，並產生針對特定程式碼變更的調查結果，因此非常適合在開發工作流程中進行合併前驗證。

與由第三方來源控制提供者事件自動觸發的提取請求型程式碼檢閱不同（請參閱the section called “檢閱提取請求中的程式碼安全性問題清單”），差異掃描是透過 AWS Security Agent API 或 SDK 以程式設計方式啟動。您可以將統一的 diff 檔案上傳到 S3，並在啟動程式碼檢閱任務時參考該檔案。

差異掃描的運作方式

差異掃描會在儲存庫的完整內容中分析程式碼變更。當您使用上傳至 S3 的原始程式碼建立程式碼檢閱資源時，差異掃描會使用完整的儲存庫做為內容，同時特別將問題清單集中在 diff 中變更的行上。這可讓掃描識別您的變更與現有程式碼互動所產生的安全問題，例如驗證流程中斷、跨模組不安全的資料處理，或公開現有漏洞的變更。

掃描程序：。您可以將統一的 diff 檔案 (的輸出`git diff`) 上傳到連線至 Agent Space 的 S3 儲存貯體。您可以使用指向差異 S3 位置的`diffSource`參數來呼叫 `StartCodeReviewJob` API。AWS Security Agent 只會分析 diff 中變更的程式碼是否有安全漏洞，以及是否符合組織的安全需求。調查結果的範圍是已變更的行，具有嚴重性評分、程式碼位置和修復指導。

先決條件

開始前，請確保您具備以下條件：

- 啟用程式碼檢閱的客服人員空間 （ 請參閱 the section called “啟用程式碼檢閱”)
- 已為目標儲存庫建立的程式碼檢閱資源，並將完整的原始程式碼上傳至連線至您的 Agent Space 的 S3 儲存貯體。完整的儲存庫提供內容，可讓您更深入地分析變更與現有程式碼的互動方式。the section called “建立程式碼檢閱” 如需建立程式碼檢閱和上傳原始程式碼的說明，請參閱 。
- 連接至客服人員空間的 S3 儲存貯體
- 上傳到 S3 儲存貯體和呼叫的 IAM 許可 `securityagent:StartCodeReviewJob`
- 從程式碼產生的統一 diff 檔案變更 （ 例如 的輸出`git diff main..feature-branch`)

> **Tip**
>
> 為了取得最準確的結果，請確定您的程式碼檢閱資源具有上傳至 S3 的最新版本完整原始程式碼。差異掃描會使用此完整儲存庫做為內容，以了解您在分析變更行時的應用程式架構、資料流程和現有安全控制。

步驟 1：產生 diff 檔案

產生代表您要掃描之程式碼變更的統一 diff。

```
git diff main..feature-branch > changes.diff
```

或者，針對階段變更產生差異：

```
git diff --cached > changes.diff
```

> **Tip**
>
> diff 檔案應為標準統一 diff 格式。AWS Security Agent 會剖析檔案路徑，並從 diff 標頭變更行以限制其分析的範圍。

步驟 2：將差異上傳至 S3

將 diff 檔案上傳到連線至 Agent Space 的 S3 儲存貯體。

```
aws s3 cp changes.diff s3://my-security-agent-bucket/diffs/changes.diff
```

> **Important**
>
> S3 儲存貯體必須是已連線至客服人員空間的儲存貯體。與您的 Agent Space 相關聯的 IAM 服務角色必須具有此位置的讀取存取權。如需連接 S3 儲存貯體的說明the section called “啟用程式碼檢閱”，請參閱 。

步驟 3：啟動差異碼檢閱任務

使用 `diffSource` 參數呼叫 `StartCodeReviewJob` API 以啟動差異掃描。

使用 AWS CLI

```
aws securityagent start-code-review-job \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --diff-source '{"s3Uri": "s3://my-security-agent-bucket/diffs/changes.diff"}'
```

使用 AWS 開發套件 (Python)

```
import boto3
```

```
client = boto3.client('securityagent')

response = client.start_code_review_job(
    agentSpaceId='your-agent-space-id',
    codeReviewId='your-code-review-id',
    diffSource={
        's3Uri': 's3://my-security-agent-bucket/diffs/changes.diff'
    }
)

print(f"Job started: {response['codeReviewJobId']}")
```

使用 AWS 開發套件 (JavaScript/TypeScript)

```
import { SecurityAgentClient, StartCodeReviewJobCommand } from '@aws-sdk/client-
securityagent';

const client = new SecurityAgentClient({ region: 'us-east-1' });

const response = await client.send(new StartCodeReviewJobCommand({
  agentSpaceId: 'your-agent-space-id',
  codeReviewId: 'your-code-review-id',
  diffSource: {
    s3Uri: 's3://my-security-agent-bucket/diffs/changes.diff',
  },
}));

console.log(`Job started: ${response.codeReviewJobId}`);
```

API 會立即傳回 `codeReviewJobId`和 `status` `IN_PROGRESS`。

步驟 4：監控掃描

輪詢程式碼檢閱任務狀態，直到完成為止。

```
aws securityagent get-code-review-job \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --code-review-job-id "the-job-id"
```

任務會進行與完整程式碼檢閱相同的階段：預檢 → 靜態分析 → 完成。差異掃描通常會比完整掃描更快完成，因為它們會分析較少的程式碼行。

步驟 5：檢閱問題清單

任務完成後，列出程式碼檢閱任務的問題清單。

```
aws securityagent list-findings \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --code-review-job-id "the-job-id"
```

每個調查結果包括：

- 安全問題的描述
- 嚴重性等級（關鍵、高、中或低）
- 參考 diff 中特定行的程式碼位置
- 解釋潛在影響的風險推理
- 修補指引

如需了解問題清單並採取行動的詳細資訊，請參閱 the section called “檢閱程式碼檢閱的問題清單”。

配額和限制

- 差異中變更的檔案數量上限：3，000 個檔案或總差異大小 5 MB
- 問題清單上限為每次掃描 30 個，依嚴重性排序（關鍵 > 高 > 中 > 低）

後續步驟

執行差異掃描之後：

- 檢閱問題清單並套用修補指引（請參閱 the section called “檢閱程式碼檢閱的問題清單”）
- 針對每個提取請求啟用自動程式碼檢閱的提取請求註解（請參閱 the section called “啟用 GitHub 儲存庫的提取請求程式碼檢閱”）
- 定期執行完整的程式碼檢閱，以捕捉個別變更以外的問題（請參閱 the section called “建立程式碼檢閱”）
- 使用 IDE 整合直接從您的開發環境執行差異掃描（請參閱 the section called “從 IDE 執行程式碼安全掃描”）

從 IDE 執行程式碼安全掃描

使用 Kiro 或 Claude Code 直接從 IDE 執行 AWS Security Agent 程式碼安全掃描。IDE 整合可讓您掃描本機原始程式碼是否有安全漏洞、僅對變更的程式碼執行差異掃描，以及對設計文件執行威脅模型檢閱，而無須離開您的開發環境。問題清單會與您的程式碼一起顯示，並附有修補指引，而且您可以直接從 IDE 套用自動修正。

IDE 整合的運作方式

IDE 整合使用 AWS Security Agent MCP （模型內容通訊協定）伺服器將您的 IDE 連線至 AWS Security Agent 服務：

1. MCP 伺服器將您的原始碼封裝為 ZIP 封存檔。
2. 封存會上傳至您 AWS 帳戶中伺服器自動佈建的 S3 儲存貯體。
3. 伺服器會呼叫 AWS Security Agent API 來啟動掃描。
4. AWS Security Agent 會分析程式碼是否有安全漏洞，以及是否符合組織的安全需求。
5. 調查結果會傳回 IDE，其中包含程式碼位置、描述、嚴重性評分和修復建議。

MCP 伺服器會處理所有協同運作：客服人員空間佈建、S3 儲存貯體建立、IAM 角色設定、程式碼封裝和掃描輪詢，因此您只需在本機設定 AWS 登入資料。

> **Note**
>
> 唯一的先決條件是在本機環境中設定的 AWS 登入資料（例如，透過 `aws configure`、AWS SSO 或環境變數）。MCP 伺服器會在第一次使用時自動佈建 Agent Space、IAM 服務角色和 S3 儲存貯體。

先決條件

開始前，請確保您具備以下條件：

- 在本機設定具有下列許可的 AWS 登入資料：
  - `iam:CreateRole`、`iam:PutRolePolicy`（適用於服務角色的一次性設定）
  - `s3:CreateBucket`, `s3:PutObject`, `s3:PutPublicAccessBlock`, `s3:PutLifecycleConfiguration`

  - `securityagent:CreateAgentSpace`, `securityagent:UpdateAgentSpace`, `securityagent:ListAgentSpaces`, `securityagent:BatchGetAgentSpaces`
  - `securityagent:CreateCodeReview`, `securityagent:StartCodeReviewJob`, `securityagent:BatchGetCodeReviewJobs`
  - `securityagent:ListFindings`, `securityagent:BatchGetFindings`
  - `securityagent:StartCodeRemediation`（適用於自動修正）
  - `sts:GetCallerIdentity`
- 已安裝 uv (Python 套件執行器）
- Python 3.10 或更新版本
- 下列其中一個 IDEs：
  - Kiro（安裝 AWS Security Agent Power)
  - Claude Code（安裝 AWS Security Agent 外掛程式）

安裝 MCP 伺服器

Kiro

從 Kiro 市集安裝 AWS Security Agent Power。電源會綁定 MCP 伺服器組態、轉向指示和生命週期掛鉤，以提供自動安全掃描建議。

或者，在專案的 中手動設定 MCP 伺服器`.kiro/mcp.json`：

```
{
  "mcpServers": {
    "security-agent": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
    }
  }
}
```

Claude 程式碼

安裝 AWS Security Agent 外掛程式，提供設定、完整掃描、差異掃描、威脅模型檢閱和修復的技能。

在專案的 中設定 MCP 伺服器`.mcp.json`：

```
{
  "mcpServers": {
    "awslabs.security-agent-mcp-server": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
    }
  }
}
```

> **Tip**
>
> 如果您不想在本機安裝 Python，也可以透過 Docker 執行 MCP 伺服器。請參閱 Docker 組態的 MCP 伺服器文件。

首次設定

第一次使用時，MCP 伺服器會自動佈建必要的 AWS 資源：

資源	名稱慣例	用途
客服人員空間	使用者選擇或自動建立	掃描、檢閱和滲透的容器
IAM 服務角色	`SecurityAgentScanRole`	由 擔任`securityagent.amazonaws.com` 以讀取上傳的程式碼

資源	名稱慣例	用途
S3 bucket (S3 儲存貯體)	`security-agent-scans-{account}-{region}`	存放壓縮的原始碼 (30 天自動過期生命週期)

若要明確觸發設定，請詢問您的 IDE：

```
Set up the security agent
```

設定會驗證您的 AWS 登入資料、建立或重複使用 Agent Space、佈建 IAM 角色和 S3 儲存貯體，並確認準備狀態。如果您已有來自 AWS 管理主控台的 Agent Spaces，則設定會顯示它們並詢問要使用哪個。

> **Note**
>
> 如果尚未設定，設定會在第一次掃描時自動執行。您不需要單獨執行。

執行完整的安全性掃描

掃描整個專案是否有安全漏洞：

```
Scan this project for security issues
```

IDE：

1. 封存您的原始程式碼 （不包括 `.git`、、`node_modules`組建成品和其他非必要目錄）
2. 將封存上傳至 S3
3. 啟動完整的程式碼檢閱任務
4. 輪詢以完成 （通常大約 1 小時）
5. 呈現依嚴重性分組的問題清單

掃描進度

完整掃描通常需要約 1 小時，取決於程式碼庫大小。IDE 每 5 分鐘檢查一次進度，並在結果就緒時通知您。您可以隨時要求狀態：

```
How's the scan going?
```

執行差異掃描

如需在開發期間更快速的意見回饋，請僅掃描自 git 參考以來變更的程式碼：

```
Scan my changes against main for security issues
```

根據預設，如果您未指定基本參考，掃描會將您未遞交的變更與 進行比較HEAD。您可以明確指定不同的基底：

```
Diff scan my uncommitted changes
```

差異掃描會同時上傳完整的儲存庫內容和 git diff 修補程式，然後執行僅著重於變更行的分析。結果通常會在 5-15 分鐘內送達。

如需 S3 diff 掃描 API 的詳細資訊，請參閱 the section called “使用 S3 執行差異碼掃描”。

執行威脅模型檢閱

使用 STRIDE 方法分析安全姿勢變更的設計文件：

```
Run a threat model review on my spec
```

IDE 可識別您的 `requirements.md`和 `design.md` 檔案（通常在 下`.kiro/specs/`）、將它們與您的原始程式碼一起上傳，並執行威脅模型分析。結果識別：

- 依 STRIDE 分類的安全威脅（詐騙、竄改、複寫、資訊揭露、拒絕服務、提升權限）
- 每個威脅的嚴重性評分
- 對程式碼庫中特定資產的影響
- 緩解的建議

> **Tip**
>
> 從規格產生實作任務之前，請執行威脅模型檢閱。這會在編寫程式碼之前發現安全設計問題。

檢閱調查結果

掃描完成時，問題清單會顯示在依嚴重性分組的 IDE 中：

```
# CRITICAL: SQL Injection in user-service.ts
   File: src/api/user-service.ts:45
   User input flows directly into SQL query without parameterization

# HIGH: Hardcoded credentials in config.ts
   File: src/config/database.ts:12
   Database password stored in source code
```

詳細的報告也會寫入 `.security-agent/findings-{scan_id}.md`，其中包含每個調查結果的完整資訊，包括風險類型、可信度分數、程式碼位置和修補程式碼。

若要檢視先前掃描的結果：

```
Show my security findings
```

套用自動修正

檢閱問題清單後，套用自動化修復：

```
Fix the security findings
```

IDE 會依嚴重性（關鍵 → 高 → 中 → 低）處理調查結果，並使用每個調查結果的修補指引套用程式碼修正。套用所有修正後，會自動執行驗證差異掃描以確認問題已解決。

您也可以修正個別問題清單：

```
Fix the SQL injection in user-service.ts
```

> **Important**
>
> 在遞交之前，請務必檢閱自動修正。確認修正不會破壞現有功能或引入迴歸。

自動掃描建議 (Kiro)

使用 Kiro Power 時，AWS Security Agent 可以在您對安全敏感的程式碼進行變更後自動建議 diff 掃描。Power 安裝勾點，評估每個完成的編碼轉彎，並在變更影響時建議掃描：

- 身分驗證或授權邏輯
- 密碼編譯或秘密處理
- 輸入驗證或資料清理
- 網路請求或外部整合
- 安全組態 (CORS、標頭、工作階段處理）

您可以隨時刪除 來選擇退出自動建議`.kiro/hooks/security-diff-scan-suggester.kiro.hook`。

可用的掃描類型

掃描類型	持續時間	使用案例	命令
完整掃描	約 1 小時	完整檢閱整個程式碼庫	「掃描我的專案是否有安全問題」
差異掃描	5-15 分鐘	僅變更程式碼（遞交前、PR 前）	"掃描我的變更" 或 "Diff scan against main"
威脅模型	5-30 分鐘	設計文件 (requirements.md：//、design.md：//)	「在我的規格上執行威脅模型檢閱」

環境變數

變數	描述	預設
AWS_REGION	SecurityAgent API 呼叫的 AWS 區域	us-east-1
AWS_PROFILE	AWS 登入資料設定檔名稱	預設設定檔
FASTMCP_LOG_LEVEL	MCP 伺服器日誌層級 (DEBUG、INFO、WARNING、 ERROR)	WARNING

疑難排解

「未設定。先執行設定。」

您的 .security-agent/config.json 遺失或代理程式空間不再存在。要求 IDE 執行設定：

```
Set up the security agent
```

AccessDenied 掃描失敗

確保您的 AWS 登入資料具有先決條件區段中列出的必要 IAM 許可。最常見的缺少許可是 iam:CreateRole（只有在第一次設定時才需要）。

掃描逾時或花費太長的時間

- 在大型程式碼庫上完整掃描可能需要超過 1 小時
- 使用差異掃描進行反覆開發 - 它們會在幾分鐘內完成
- 檢查您的來源封存不包含不必要的目錄 (MCP 伺服器會自動排除常見模式）

空差 — 掃描沒有變更

如果您執行 diff 掃描時沒有未遞交的變更，MCP 伺服器會報告「沒有變更 vs HEAD」，而且不會啟動掃描。先遞交或暫存變更，或指定不同的基礎參考。

配額和限制

- 來源封存大小上限：2 GB
- S3 儲存貯體生命週期：上傳的程式碼會在 30 天後自動刪除

後續步驟

執行第一次 IDE 安全掃描之後：

- 啟用自動 GitHub 整合的提取請求程式碼檢閱註解（請參閱 the section called “啟用 GitHub 儲存庫的提取請求程式碼檢閱”）
- 設定組織特定政策驗證的安全需求（請參閱 the section called “管理安全需求”）
- 執行定期完整掃描以在整個程式碼庫中發現問題（請參閱 the section called “建立程式碼檢閱”）
- 在實作之前，對新功能規格使用威脅模型檢閱

建立滲透測試

透過設定測試範圍、目標網域和 AWS 資源存取，為您的 Web 應用程式設定自動化滲透測試。滲透測試透過模擬已驗證網域的真實攻擊案例，協助識別執行中應用程式的安全漏洞。

AWS Security Agent 會根據設定的範圍和許可，對您的 Web 應用程式執行全面的安全測試，在攻擊者可以發現漏洞之前提供詳細的調查結果。

在此程序中，您將透過設定測試詳細資訊、定義測試範圍和設定必要的許可來建立滲透測試。

先決條件

開始前，請確保您具備以下條件：

- 存取 AWS Security Agent Web 應用程式
- 至少一個用於測試的已驗證網域
- 具有 AWS Security Agent 適當許可的 IAM 角色
- 了解應用程式的架構和關鍵路徑

開始建立滲透測試

導覽至 Agent Web 應用程式中的滲透測試建立頁面。

1. 登入 AWS Security Agent Web 應用程式。
2. 導覽至滲透測試區段。
3. 選擇建立滲透測試。

> **Tip**
>
> 只有已驗證的網域可以包含在滲透測試中。請您的管理員在 AWS 管理主控台中驗證網域。請參閱 the section called “啟用用於滲透測試的應用程式網域”。

為您的滲透測試命名

提供描述性名稱，以協助識別此滲透測試的目的和範圍。

1. 在滲透測試名稱欄位中，輸入滲透測試的描述性名稱。

   Example

   名稱應清楚識別要測試的應用程式、環境或元件。最多 100 個字元。

設定滲透測試範圍

定義要測試的網域和 URL 路徑，並設定選用的排除以控制測試界限。

新增目標網域

指定將主動測試安全性漏洞的已驗證網域。

1. 在滲透測試範圍區段中，找到目標 URLs。
2. 展開已驗證網域區段以檢視可用的網域。
3. 在目標 URL 欄位中，輸入目標網域 URL。

> **Important**
>
> 只能測試已驗證的網域。URL 必須位於您先前在 AWS Security Agent 中驗證過的網域下。已驗證網域的子網域不需要個別驗證。

4. 若要新增多個目標網域：
   a. 選擇新增網域。
   b. 輸入每個額外的網域 URL。
5. 若要移除目標網域，請選擇網域 URL 旁的移除。

> **Tip**
>
> 為了獲得最佳結果，請包含應用程式使用者流程中的所有網域，包括 APIs、身分驗證服務和內容交付的子網域。已驗證父網域的子網域不需要個別驗證。

排除風險類型（選用）

如果特定風險類別不適用於您的應用程式，請選擇特定風險類別來排除測試。

1. 找到排除風險類型欄位。
2. 選擇下拉式清單以檢視可用的風險類型。
3. 選取要從滲透測試中排除的一或多個風險類型。

> **Note**
>
> 排除風險類型會限制測試範圍。僅排除與您的應用程式無關或您想要單獨測試的風險類型。

新增out-of-scope URL 路徑（選用）

指定在滲透測試期間不應測試的 URL 路徑。AWS Security Agent 會排除指定的路徑及其下巢狀的所有路徑。例如，如果您將 新增`https://example.com/admin`為out-of-scope的 URL，`https://example.com/admin/tools` 也會out-of-scope。

1. 找到Out-of-scope URLs區段。
2. 在Out-of-scope URLs輸入欄位中，輸入要排除的 URL 路徑（例如 `/admin/delete`或 `/api/reset`)。

> **Warning**
>
> Out-of-scope路徑不會測試漏洞。請務必僅排除測試期間不應存取的路徑，例如破壞性操作或敏感管理函數。

3. 若要新增多個out-of-scope的路徑：
   a. 選擇新增 URL。
   b. 輸入每個額外的路徑。
4. 若要移除路徑，請選擇路徑旁的移除。

新增可存取的網域（選用）

指定測試所需的網域，但不是漏洞測試的目標。

1. 找到可存取URLs 區段。
2. 在可存取 URLs 輸入欄位中，輸入應在測試期間存取的網域。

> **Note**
>
> 為目標網域以外的第三方服務（例如 Okta、Auth0、Stripe) 新增可存取的網域。這是必要的，因此 AWS Security Agent 可以在測試期間存取這些 URLs以進行登入和導覽。AWS Security Agent 不會滲透測試這些網域，它們僅用於存取目的。可存取的網域不需要所有權驗證，即使它們屬於與您的目標不同的網域。只有目標網域需要經過驗證的擁有權。

3. 若要新增多個可存取的網域：
   a. 選擇新增 URL。
   b. 輸入每個額外的網域。
4. 若要移除網域，請選擇網域旁的移除。

新增自訂 HTTP 標頭（選用）

指定自訂 HTTP 標頭，這些標頭將在滲透測試期間新增至 AWS Security Agent 提出的任何請求。

1. 找到自訂 HTTP 標頭區段。
2. 在自訂 HTTP 標頭輸入欄位中，輸入與傳出請求相關聯的標頭名稱和值。

> **Note**
>
> 根據預設，AWS Security Agent 會將 User-Agent 的自訂標頭設定為 securityagent，除非指定了不同的自訂 User-Agent 標頭值。

3. 若要新增多個自訂標頭：
   a. 選擇新增標頭。
   b. 輸入每個額外的自訂標頭。
4. 若要移除自訂標頭，請選擇標頭旁的移除。

設定 IAM 角色

選取此滲透測試的預先設定服務角色。AWS Security Agent 使用以 Agent Space 為基礎的許可模型，其中管理員會在設定您的 Agent Space 時設定 IAM 角色。您可以從已設定且可供使用的角色中選取。

1. 在許可區段中，找到服務角色下拉式清單。
2. 選取授予 AWS Security Agent 必要 AWS 資源存取權的 IAM 角色。

> **Important**
>
> 選取的 IAM 角色必須具有存取 VPC 資源、CloudWatch Logs 和滲透測試所需的任何其他 AWS 服務的許可。確認角色與 AWS Security Agent 具有正確的信任關係。

3. 找到 CloudWatch 日誌群組下拉式清單。
4. 選取將存放滲透測試日誌的日誌群組。（選用）

> **Note**
>
> 選取的 CloudWatch 日誌群組將存放滲透測試執行的詳細日誌，包括提出的請求、收到的回應，以及發現的漏洞。
> 如果您未選取日誌群組，系統會自動建立一個具有 `/aws/securityagent`字首的新 CloudWatch 日誌群組來存放滲透測試日誌。

自動程式碼修復

選取啟用自動修復核取方塊。

> **Important**
>
> 為了修復原始碼儲存庫中的安全調查結果，AWS Security Agent 可能會向您的儲存庫提交提取請求。具有儲存庫讀取存取權的所有使用者都可看見提取請求。

設定 VPC 資源 （選用）

如果您的目標網域是私有且託管在 VPC 內，請設定 AWS Security Agent 應執行滲透測試的 VPC 設定。只有無法公開存取的應用程式才需要此步驟。

> **Note**
>
> 如果您的目標網域可公開存取，請略過此步驟。只有測試 Amazon Virtual Private Cloud 中託管的私有應用程式時，才需要 VPC 組態。

選擇滲透測試環境的 VPC、子網路和安全群組。

1. 在 VPC 區段中，找到 VPC ID 下拉式清單。
2. 選取託管目標網域的 VPC。

   > **Important**
   >
   > 選取的 VPC 必須包含您在步驟 3 中指定的目標網域。確保 VPC 具有適當的路由和網路組態，以允許 AWS Security Agent 存取您的應用程式。

3. 找到子網路下拉式清單。
4. 選取一或多個應執行滲透測試的子網路。

   > **Note**
   >
   > 選擇具有目標應用程式網路存取權的子網路。滲透測試將從這些子網路中部署的資源執行。

5. 找到安全群組下拉式清單。

6. 選取控制滲透測試網路存取的安全群組。

> **Important**
>
> 選取的安全群組必須允許傳出流量到您的目標網域和任何可存取的網域。確保安全群組規則允許必要的網路存取以進行全面測試。

設定身分驗證憑證 （選用）

如果您的目標網域需要身分驗證，請提供登入資料，讓 AWS Security Agent 在滲透測試期間存取您應用程式的受保護區域。只有需要使用者身分驗證的應用程式才需要此步驟。

> **Note**
>
> 如果您的目標網域不需要身分驗證，或您想要測試的所有區域皆可公開存取，請略過此步驟。只有在您需要 AWS Security Agent 測試應用程式的已驗證區段時，才設定登入資料。

新增登入資料

提供 AWS Security Agent 用來存取應用程式的身分驗證憑證。

1. 在登入資料 #1 區段中，選取登入資料輸入方法：
   - 輸入登入資料 - 將您的登入資料直接輸入 AWS Security Agent。
   - 進階設定 - 對於敏感登入資料資訊，請使用進階選項，例如 AWS Secrets Manager 或 AWS Lambda 函數。如需詳細資訊，請參閱 the section called “提供用於滲透測試的身分驗證憑證”。

     > **Tip**
     >
     > 對於生產環境或敏感登入資料，我們建議您使用進階設定選項，安全地參考存放在 AWS Secrets Manager 或 Systems Manager 參數存放區的登入資料。

輸入登入資料詳細資訊

提供已驗證帳戶的使用者名稱和密碼。

1. 在使用者名稱欄位中，輸入身分驗證的使用者名稱。

2. 在密碼欄位中，輸入身分驗證的密碼。

> **Important**
>
> 確保您提供的登入資料對於您想要測試的區域具有適當的存取層級。登入資料應代表典型使用者的存取層級，而非管理權限。

選取存取網域

指定哪個目標網域將使用這些登入資料進行身分驗證。

1. 在存取網域下拉式清單中，選取使用這些登入資料的網域。

> **Note**
>
> 如果您有多個目標網域需要不同的登入資料，您可以在完成此登入資料組態後按一下新增另一個登入資料來新增其他登入資料集。

設定客服人員登入提示（選用）

提供指示，引導 AWS Security Agent 完成應用程式的身分驗證程序。

1. 如果您的身分驗證流程需要特定指示，請展開客服人員登入提示區段。
2. 輸入指示，說明如何在應用程式的登入流程中使用提供的登入資料。

> **Note**
>
> 客服人員登入提示會告知客服人員如何將您的登入資料套用至您的應用程式。這適用於複雜的身分驗證流程、多步驟登入程序，或具有非標準登入程序的應用程式。包含step-by-step指示，例如「導覽至 /登入」，在「電子郵件」欄位中輸入使用者名稱，輸入密碼，然後選擇「登入」。

新增多個登入資料（選用）

如果您的應用程式需要多組登入資料，或不同的網域需要不同的身分驗證，請新增其他登入資料集。

1. 完成第一個登入資料組態後，選擇新增另一個登入資料。
2. 為每個額外的登入資料集重複登入資料組態步驟。
3. 若要移除登入資料集，請選擇登入資料標頭旁的移除。

> **Tip**
>
> 在測試不同的使用者角色、存取多個已驗證的網域，或驗證應用程式中的角色型存取控制時，設定多個登入資料。

連接其他資源（選用）

提供補充資源，以協助 AWS Security Agent 進行更徹底且準確的滲透測試。其他資源可能包括架構圖、API 文件、組態檔案、GitHub 儲存庫或 S3-hosted的資料，以提供應用程式的相關內容。

> **Note**
>
> 其他資源是選用的，但建議使用。提供應用程式的完整資訊有助於確保完整的測試涵蓋範圍、減少誤報，並提供更可行的結果。

將資源新增至滲透測試

選取現有的資源或上傳有助於引導滲透測試的新檔案。

1. 在連線資源區段中，您可以：
   - 選擇從可用中選擇，從已連線至 AWS Security Agent 的資源中選擇（例如 GitHub 儲存庫或 S3 儲存貯體）。
   - 選擇上傳，直接從本機系統新增檔案。

> **Tip**
>
> 有用的資源包括 API 文件、架構圖、OpenAPI/Swagger 規格、組態檔案、身分驗證流程圖，以及描述應用程式結構和行為的任何其他資料。

從可用資源中選取

從已與 AWS Security Agent 整合的資源中進行選擇。

1. 選擇從現有資源中選取。
2. 瀏覽來自連線來源的可用資源清單，例如：
   - GitHub 儲存庫，在 GitHub 儲存庫索引標籤下
   - S3 儲存貯體
   - 先前上傳的檔案
   - 文件儲存庫
3. 選取您要包含在滲透測試中的資源。
4. 選擇新增至滲透測試以連接選取的資源。

Example

我們建議您選取相關 GitHub 儲存庫並將其新增至您的 pentest，以便 AWS Security Agent 能夠了解您的應用程式內容，並透過提取請求產生ready-to-implement程式碼修正（啟用時）

> **Note**
>
> 從可用來源選取的資源會與其原始位置保持同步。如果您更新 GitHub 儲存庫或 S3 檔案，滲透測試將使用更新的版本。

> **Note**
>
> 如果您有與 pentest 相關聯的私有 VPC，且已設定 GitHub 儲存庫，請確定 GitHub 可透過私有 VPC 存取，以提取 GitHub 資源。在大多數情況下，您將需要確保預設允許透過 VPC NAT Gateway 的傳出流量，或設定特定規則以允許 GitHub IPs的傳出流量（請參閱 GitHub Meta API 端點)

上傳新資源

直接從本機系統上傳檔案，或提供純文字內容給 AWS Security Agent。

1. 選擇上傳。

2. 選擇下列其中一個輸入方法：
   - 上傳本機檔案 - 從本機系統選取一或多個檔案。
   - 貼上純文字 - 直接在輸入欄位中輸入或貼上文字內容。選擇上傳。
3. 然後選擇新增以完成上傳。
4. 上傳的資源會出現在連線資源資料表中。

> **Tip**
>
> 當您想要快速提供 API 端點清單、URL 模式、測試指示或其他文字型資訊，而不建立個別的檔案時，請使用純文字選項。

> **Important**
>
> 確保上傳的檔案和貼上的內容不包含敏感資訊，例如生產登入資料、私有金鑰或個人身分識別資訊 (PII)。使用已淨化版本的組態檔案和文件。

連接現有資源

現有的資源可以來自您先前上傳至 AWS Security Agent 的內容、S3 儲存貯體，以及整合的 GitHub 儲存庫。選擇從現有資源中選取以選取它們。

管理連線的資源

檢閱、組織和移除連接到滲透測試的資源。

連線的資源資料表會顯示滲透測試中包含的所有資源，其中包含以下資訊：

- 名稱 - 檔案名稱或資源識別符
- 類型 - 資源類別（上傳的檔案、S3 資源、GitHub 儲存庫等）

若要管理 資源：

1. 使用核取方塊選取一或多個資源。
2. 選擇從滲透測試中移除以分離選取的資源。

> **Note**
>
> 您可以按一下資料欄標頭，依名稱或類型排序資料表。這有助於在處理許多檔案時組織資源。

建立滲透測試

完成並啟動您的滲透測試組態。

設定所有設定後，您就可以建立滲透測試。

1. 檢閱所有組態區段以確保準確性。
2. 請選擇下列其中一個選項：
   - 選擇建立滲透，以儲存組態而不立即執行。
   - 選擇建立並執行以儲存組態，並立即開始滲透測試。
   - 選擇取消以捨棄滲透測試組態。

> **Important**
>
> 執行滲透測試之前，請確認：
>
> - 所有目標網域都已正確驗證並可存取
> - IAM 角色具有適當的許可
> - 已正確設定Out-of-scope的路徑，以防止測試破壞性操作
> - 您有權對所有目標網域執行安全測試

> **Note**
>
> 滲透測試開始之後，您可以從滲透測試執行區段監控其進度。完成測試可能需要幾個小時。大部分會在 16 小時內完成，取決於應用程式的範圍和複雜性。

檢閱滲透測試的問題清單

在 AWS Security Agent 啟動滲透測試後，在滲透測試日誌頁面上即時監控滲透測試執行。AWS Security Agent 會記錄 pentest 期間的每個動作。完成後，請檢閱 pentest 摘要，其中包括應用程式概觀、已識別端點的涵蓋範圍，以及安全調查結果的風險評估。

評估安全性調查結果以解決應用程式漏洞。每個調查結果都包含影響評估、嚴重性評分、支援證據和修補提取請求詳細資訊（啟用自動程式碼修補時）。

先決條件

開始前，請確保您具備以下條件：

- 已完成或進行中滲透測試執行
- 存取 AWS Security Agent Web 應用程式

步驟 1：存取滲透測試執行

導覽至您的滲透測試執行以檢視概觀、日誌和問題清單頁面。

1. 登入 AWS Security Agent Web 應用程式。
2. 導覽至滲透測試區段。
3. 從清單中選擇您要檢查的滲透測試執行。

> **Tip**
>
> 滲透測試詳細資訊頁面會顯示測試狀態、完成日期和已識別問題清單數量的摘要。

步驟 2：監控測試進度

使用步驟指示器追蹤滲透測試執行的進度。

1. 找到頁面標頭下方的水平步驟指標。
2. 檢閱每個測試階段的狀態：
   - 預檢 – 初始設定和連線檢查
   - 靜態分析 – 程式碼和組態分析
   - 滲透測試 – 執行期測試和漏洞掃描

- 完成 – 最終驗證和報告產生

> **Note**
>
> 每個步驟都會顯示狀態指示燈（完成、進行中或待定）。問題清單會在整個測試過程中探索和驗證，每個階段完成時都會出現新的漏洞。

步驟 3：導覽至滲透測試執行概觀索引標籤

1. 執行摘要區段提供測試狀態、持續時間和其他高階詳細資訊。它還提供按嚴重性等級和風險類型分類的安全調查結果儀表板
2. AWS Security Agent 的應用程式概觀提供滲透測試執行的摘要
3. AWS Security Agent 探索的端點提供 AWS Security Agent 在五項測試執行期間探索和測試的所有端點清單

步驟 4：導覽至滲透測試日誌索引標籤

存取 AWS Security Agent 在五項測試期間執行之所有動作的詳細日誌。

1. 動作依動作類型和風險類型分類。
2. 選取特定動作以檢視詳細日誌：
   - 測試摘要 – 客服人員動作和結果的高階摘要
   - 滲透測試日誌 – 所有測試活動的詳細日誌

> **Note**
>
> 驗證器動作提供可驗證每個類別中問題清單的日誌

> **Note**
>
> 問題清單索引標籤會顯示分割檢視，其中包含左側的問題清單，以及右側的所選問題清單詳細資訊。

步驟 5：導覽至問題清單

清單中的每個調查結果都會顯示重要資訊，以協助您快速評估其重要性。

> **Note**
>
> 預設可信度篩選條件
>
> 根據預設，您只會看到具有高客服人員可信度的問題清單。若要顯示具有中或低客服人員可信度和誤報的問題清單，請關閉隱藏未驗證的問題清單切換。

檢閱每個問題清單卡片上顯示的資訊：

- 尋找名稱 – 漏洞的標題和識別符
- 可信度徽章 – 表示客服人員在調查結果中的可信度等級（高、中或低）
- 嚴重性徽章 – 顯示具有顏色編碼的風險等級：
  - 嚴重（紅色）– 需要立即採取動作；利用可能會導致系統入侵
  - 高（紅色）– 需要立即注意；利用可能會導致重大的安全影響
  - 中（橘色）– 應在合理的時間範圍內解決；造成整體安全風險
  - 低（黃色）– 可以作為定期維護的一部分來處理；將立即風險降至最低
  - 資訊性（藍色）– 用於資訊性用途；最低或無立即風險
- 上次更新時間戳記 – 顯示上次修改或驗證調查結果的時間
- 描述預覽 – 漏洞的簡短摘要

> **Important**
>
> 使用關鍵或高嚴重性徽章和高可信度層級來排定調查結果的優先順序，因為這些是需要立即修復的已驗證漏洞。

步驟 6：檢閱問題清單詳細資訊

選取個別問題清單以檢視每個漏洞的完整資訊。

1. 在左側面板中選取問題清單名稱，以在右側面板中顯示其詳細資訊。

2. 檢閱驗證狀態：

> **Note**
>
> 如果問題清單顯示未知的「此問題清單尚未經過 AWS Security Agent 驗證」，表示漏洞偵測仍在確認中。這些調查結果可能需要手動驗證。

3. 檢閱顯示在頂端的關鍵屬性：
   - 客服人員可信度 – AWS Security Agent 在此調查結果中的可信度層級
   - 嚴重性 – 顏色編碼徽章的風險等級
   - 尋找日誌 – 選擇「追蹤動作和日誌」以檢視詳細的執行日誌和證據
   - 風險類型 – 安全風險的類別或類型（例如，Authentication Bypass、SQL Injection)
4. 展開描述區段以讀取：
   - 漏洞的詳細說明
   - 漏洞的運作方式
   - 它代表安全風險的原因
   - 對應用程式的潛在影響
5. 展開風險原因區段，以了解嚴重性計算：
   - CVSS（常見漏洞評分系統）指標明細
   - 攻擊媒介 (AV) – 如何利用漏洞
   - 攻擊複雜性 (AC) – 入侵的難度
   - 必要權限 (PR) – 需要哪些存取層級
   - 使用者互動 (UI) – 是否需要使用者動作
   - 範圍 (S) – 漏洞是否影響其他元件
   - 機密性、完整性和可用性影響
6. 展開要檢視的重現步驟區段：
   - 重新建立漏洞的詳細技術步驟
   - 請求和回應範例
   - 觸發問題的特定參數或條件
7. 驗證指令碼區段提供重現問題清單的可執行方式。展開本節（可用時）以檢視：
   - 指示 – 如何設定和執行驗證指令碼
   - 環境變數 – 列出您在執行指令碼之前必須設定的必要變數。為了安全起見，會修訂敏感值。

- 下載指令碼 – 選擇下載可執行驗證指令碼

> **Note**
>
> 驗證指令碼僅適用於已確認的漏洞。代理程式必須已成功產生並驗證可執行的重製指令碼。

> **Note**
>
> 驗證指令碼是使用生成式 AI 產生的。在執行之前檢閱指令碼，並僅針對您獲授權測試的系統執行指令碼。如需以負責任的方式測試 AI 產生的指令碼的指引，請參閱 AWS 負責任的 AI 政策。

> **Tip**
>
> 驗證指令碼提供可獨立重現問題清單的可執行方式。使用您自己的登入資料設定所需的環境變數，然後針對目標系統執行指令碼，以確認漏洞存在。

> **Tip**
>
> 使用「追蹤動作和日誌」連結來存取完整的證據套件，包括顯示漏洞的 HTTP 請求、回應和入侵嘗試。

編輯問題清單

您可以編輯任何問題清單，以更正其詳細資訊或精簡客服人員的評估。下列欄位可供編輯：

- name — 調查結果的標題
- description — 安全漏洞的 詳細說明
- 狀態 — 調查結果的 目前狀態
- riskType — 已識別的安全風險 類型
- riskLevel — 嚴重性等級 （關鍵、高、中、低、資訊）
- riskScore — 數值風險分數

- 推理 — 指派風險分數的理由
- attackScript — Proof-of-concept程式碼
- customerNote — 選用備註，說明編輯的理由

若要編輯問題清單：

1. 導覽至調查結果詳細資訊頁面。
2. 選擇編輯圖示以修改上述清單中的任何欄位。
3. 儲存您的變更。

> **Note**
>
> 您的編輯會立即儲存並反映在調查結果中。如果啟用問題清單個人化，您變更的所有可編輯欄位都會用於精簡客服人員未來執行的偏好設定。

檢視原始代理程式版本

當您編輯問題清單時，版本選擇器會出現在問題清單詳細資訊標頭中。這可讓您檢視原始客服人員評估：

1. 在問題清單詳細資訊標頭中尋找版本選取器。
2. 選取原始來檢視最初由代理程式產生的問題清單。
3. 選取最新以返回目前版本，並套用您的編輯。

檢閱個人化問題清單

啟用問題清單個人化時，AWS Security Agent 會從對問題清單的編輯中學習。代理程式會將這些偏好設定套用至未來滲透測試執行中的類似問題清單。根據先前的編輯調整問題清單後，詳細資訊面板會顯示個人化變更區段。如需啟用此功能的資訊，請參閱啟用或停用問題清單個人化。

1. 在調查結果詳細資訊面板中尋找個人化變更區段。

> **Note**
>
> 個人化變更區段只會出現在 AWS 安全代理程式符合您先前編輯的問題清單上。在沒有本節的情況下，符合未學習偏好設定的問題清單會與客服人員產生的問題清單完全相同。

2. 檢閱變更內容和原因的摘要。摘要會列出每個調整過的屬性及其原始代理程式產生的值、個人化值和推理。例如：

根據您先前的編輯，已對系統最初產生的調查結果進行下列變更：風險層級從 MEDIUM 變更為 CRITICAL；風險分數從 5.3 變更為 9.5；原因：升級嚴重性，以透過可公開存取的來源映射反映應用程式原始碼的完整暴露。

1. 檢閱摘要，了解在決定如何對其採取行動之前，如何調整調查結果。
2. 若要覆寫個人化問題清單，請像任何其他問題清單一樣再次進行編輯（例如，變更嚴重性或風險分數）。您最新的編輯一律優先：AWS Security Agent 會從中學習，並在下次執行時套用更新的偏好設定，取代先前的規則。

> **Note**
>
> 個人化變更會調整諸如 riskLevel、riskScore、名稱、描述、推理和 attackScript 等調查結果屬性，以反映 AWS Security Agent 從您的編輯中學到的標準。您先前編輯的任何欄位，可能會在未來執行中針對類似的問題清單進行調整。

個人化如何從您的編輯中學習

啟用問題清單個人化時，AWS Security Agent 會從您對問題清單所做的所有編輯中學習，並將類似的調整套用至未來的執行。您編輯的任何欄位（如上述編輯問題清單一節所列）將用於精簡客服人員學習到的偏好設定。

> **Note**
>
> 對於狀態欄位，只有將問題清單標記為 FALSE_POSITIVE 才會被視為可學習的偏好設定。變更其他狀態值不會觸發學習。

下次執行五項測試時，代理程式會根據您學到的偏好設定評估新問題清單，並在適用時套用調整。已調整的任何調查結果上都會顯示個人化變更區段，說明變更的項目。

> **Note**
>
> 個人化只會從最近完成的五項測試執行中所做的編輯中學習。必須在 pentest 開始時啟用此功能，才能進行學習。如果您在稍後的執行中編輯相同的問題清單，則最新編輯優先 — 代理程式會更新其偏好設定，以反映您的最新決策。

啟用或停用問題清單個人化

問題清單個人化預設為啟用。若要停用或重新啟用：

1. 導覽至您的 pentest 組態。
2. 找到問題清單個人化切換。
3. 若要啟用問題清單個人化，請開啟切換。若要停用，請將其關閉。

停用時，問題清單會以原始代理程式產生的狀態顯示，不會套用個人化。先前學習的偏好設定會保留，如果重新啟用此功能，將會再次套用。

步驟 7：解譯 CVSS 指標

了解 CVSS 指標可協助您評估真正的嚴重性，並排定修補工作的優先順序。

檢閱原因區段時，請注意這些關鍵指標：

- 攻擊向量 (Network/Adjacent/Local/Physical) – 指出可遠端執行攻擊的方式
- 攻擊複雜性（低/高）– 顯示是否需要特殊條件才能利用
- 必要權限 (None/Low/High) – 識別攻擊者需要的存取層級
- 使用者互動（無/必要）– 判斷入侵是否需要使用者參與
- Confidentiality/Integrity/Availability(None/Low/High) – 測量對系統安全性的影響

> **Important**
>
> 具有網路攻擊向量、低複雜性和高機密性/完整性影響的調查結果代表需要立即注意的最危險漏洞。

步驟 8：排定問題清單的優先順序並加以解決

對問題清單採取動作，以修復漏洞並改善應用程式的安全狀態。

對於具有高可信度的關鍵和高嚴重性問題清單：

1. 檢閱描述和步驟，以完整重現章節。
2. 透過「追蹤動作和日誌」連結存取詳細日誌，以收集完整的證據。
3. 透過下列其中一種方法存取ready-to-implement修正：
   - 對於自動修復：使用修復區段中的提取請求連結
   - 對於手動請求：在問題清單頁面上選擇「修正程式碼」，以請求提取請求先決條件：
   - 管理員必須在 AWS Security Agent 主控台中啟用 GitHub 儲存庫的程式碼修復
   - 儲存庫必須包含在您的 pentest 組態中
4. 規劃後續滲透測試，以確認修正是否有效。

對於中等和低嚴重性問題清單：

1. 根據您的風險承受能力和業務內容來排定優先順序。
2. 在定期開發衝刺規劃中包含修復任務。
3. 考慮多個低嚴重性調查結果是否共同造成較高的風險。
4. 以適當的理由記錄任何接受的風險。

> **Important**
>
> 請勿忽略低嚴重性的問題清單。多個低嚴重性漏洞通常可以鏈結在一起，以建立更嚴重的入侵，特別是在與社交工程或實體存取結合時。

步驟 9：追蹤修復進度

使用問題清單界面來追蹤已解決的漏洞，以及需要進一步動作的漏洞。

1. 當您進行修復時，請參閱重現步驟一節以驗證您的修正。
2. 記錄每個調查結果的修補方法，以供未來參考和合規稽核。

> **Tip**
>
> 維護修補日誌，將每個調查結果映射至其解決方案，包括程式碼變更、組態更新或為了解決漏洞而做出的架構決策。

後續步驟

檢閱滲透測試結果後：

- 以高度可信度優先處理關鍵和高嚴重性的問題清單，以便立即修復
- 在您的測試環境中下載、檢閱和執行驗證指令碼，以測試指令碼並識別漏洞
- 在問題管理系統中建立追蹤票證，其中包含尋找詳細資訊和證據的連結
- 實作修正和安全控制，以解決已識別的漏洞
- 監控滲透測試執行進度指標是否有新發現的漏洞
- 安排追蹤滲透測試，以確認漏洞已正確修復
- 根據調查結果更新您的應用程式安全測試程序和威脅模型
- 檢閱 CVSS 指標以了解應用程式的整體安全狀態

如需執行滲透測試的詳細資訊，請參閱 the section called “建立滲透測試”。

如需了解安全代理程式生命週期的詳細資訊，請參閱 the section called “了解資源階層和生命週期”。

提供用於滲透測試的身分驗證憑證

提供登入資料，讓 AWS Security Agent 能夠測試 Web 應用程式的已驗證區域。如果沒有登入資料，代理程式只能測試可公開存取APIs。

設定身分驗證憑證

1. 在滲透測試建立工作流程中，找到身分驗證憑證 - 選用區段。
2. 在登入資料 #1 區段中，選擇您的登入資料輸入方法：
   - 輸入登入資料 - 直接輸入登入資料。最適合開發和測試環境。
   - 進階設定 - 使用 AWS 原生憑證管理。建議用於生產環境和敏感登入資料。

進階選項

如果選取進階設定，您可以選擇三種登入資料策略：

- IAM 角色假設 - 適用於使用 AWS Cognito 或 IAM 身分驗證的應用程式
- AWS Secrets Manager - 用於加密和輪換的安全登入資料儲存
- Lambda 函數 - 用於產生動態憑證或複雜身分驗證流程

直接輸入登入資料

1. 選取輸入登入資料。
2. 輸入使用者名稱和密碼。
3. 在存取 URL 下拉式清單中，選取使用這些登入資料的 URL。這必須從目標端點清單中選取。
4. （選用）在 2FA - 選用欄位中，為需要雙重驗證的應用程式提供 TOTP 秘密。您可擇一方法：
   - 直接輸入 TOTP 秘密（例如 `JBSWY3DPEHPK3PXP`），或輸入完整的 `otpauth://totp/` URI（例如 `otpauth://totp/Example:user@example.com?secret=JBSWY3DPEHPK3PXP&issuer=Example`）。
   - 選擇上傳圖示，從您的驗證器應用程式設定頁面上傳 QR 碼映像。QR 碼會在本機掃描，並自動擷取 TOTP URI。

     提供 TOTP 秘密時，代理程式會自動產生新的一次性代碼，並在登入期間偵測到 2FA 提示時輸入。
5. （選用）如果您的應用程式具有複雜的身分驗證流程，請展開客服人員空間登入提示，以提供特定的登入指示。

> **Important**
>
> 使用具有代表性存取權的測試帳戶，而不是個人或管理帳戶。

使用進階設定

當您選取進階登入資料策略 (Secrets Manager、Lambda 或 IAM 角色）時，AWS Security Agent 會使用 服務角色直接從設定的 AWS 資源擷取您的登入資料。使用客服人員空間登入提示，為客服人員提供如何將這些登入資料套用至應用程式的指示，例如，要導覽至哪個登入 URL 以及要填寫的表單欄位。

> **Note**
>
> Secrets Manager 秘密和 Lambda 函數必須與 AWS Security Agent 設定位於相同的 AWS 帳戶中。目前不支援跨帳戶登入資料。

1. 選取進階設定。
2. 在使用者存取策略下拉式清單中，選擇下列其中一項：

選取要擔任的客服人員的可用 IAM 角色

對於使用 AWS Cognito、具有 IAM 身分驗證的 API Gateway 或其他 AWS 原生身分驗證系統的應用程式，請使用此選項。IAM 角色必須具有信任關係，允許 AWS Security Agent 擔任該角色，以及存取應用程式身分驗證系統的許可。

從連接的 AWS Secrets Manager 選取靜態登入資料

使用此選項可透過加密、輪換和存取稽核，從 AWS Secrets Manager 安全地擷取憑證。

IAM 角色必須具有 `secretsmanager:GetSecretValue`和 `secretsmanager:DescribeSecret`許可。

代理程式會直接從 Secrets Manager 擷取秘密值。使用客服人員空間登入提示，告知客服人員如何將這些登入資料套用至應用程式的登入流程 - 例如，要導覽至哪個 URL 以及要填寫的表單欄位。您可以使用任何格式來存放秘密，因為代理程式會使用您在登入提示中提供的指示來解譯格式。

例如，如果客服人員要在 https：//https://example.com/login 提交使用者名稱/密碼登入表單，您可以使用 `username`和 `password` 欄位將您的秘密格式化為 JSON。如果應用程式需要以 TOTP 為基礎的 2FA，請直接包含具有 TOTP 秘密或完整 `otpauth://totp/` URI `totpSecret`的欄位：

```
{
  "username": "test-user",
  "password": "secure-password-here",
  "totpSecret": "JBSWY3DPEHPK3PXP"
}
```

然後，設定身分驗證指示：。將存取 URL 設定為 `https://example.com` (或從目標端點清單中選擇的任何其他 URL)。在客服人員空間登入提示中輸入以下內容：「導覽至 https://example.com/login：//。」

另一個範例是，如果您改為在 HTTP 標頭中提供 API 金鑰，您可以將它儲存為純文字：

```
"api-key-here"
```

然後，設定身分驗證指示：。在客服人員空間登入提示中輸入以下內容：「將所有請求的 X-API-Key 標頭設定為提供的 API 金鑰。」

> **Important**
>
> 僅支援以 TOTP 為基礎的 2FA。不支援 SMS、電子郵件、推送通知、硬體金鑰和 OAuth 身分驗證。

選取可用的 Lambda 函數以動態擷取登入資料

針對複雜的身分驗證系統、動態憑證產生或與外部身分提供者整合，請使用此選項。

IAM 角色必須具有`lambda:InvokeFunction`許可，且函數必須在 30 秒內完成。

當滲透測試執行時，AWS Security Agent 會同步叫用您的 AWS Lambda 函數，並傳遞識別滲透測試和要解決之特定憑證的事件：

```
{
  "pentest_arn": "arn:aws:securityagent:us-west-2:111122223333:pentest/pt-
abcd1234-5678-90ab-cdef-EXAMPLE11111",
  "actor_identifier": "Credential1"
}
```

- `pentest_arn` – 要解析憑證之滲透測試的 Amazon Resource Name (ARN)。使用它來限定、授權或稽核函數中的請求。
- `actor_identifier` – 您在主控台中指派給此登入資料的登入資料名稱 （ 例如，`Credential1`)。當單一函數提供多個登入資料時，請使用它來傳回正確的登入資料。

代理程式會直接使用函數的輸出做為登入資料。使用客服人員空間登入提示，告訴客服人員如何將這些登入資料套用至您的應用程式，就像您使用 AWS Secrets Manager 一樣。如需如何格式化 Lambda 函數輸出和支援身分驗證類型the section called “從連接的 AWS Secrets Manager 選取靜態登入資料”的範例，請參閱 。

設定多個登入資料

若要測試不同的使用者角色或身分驗證系統：

1. 選擇新增另一個登入資料。
2. 使用任一輸入方法設定其他登入資料。
3. 若要移除登入資料，請在登入資料區段中選擇移除。

登入最佳化

登入最佳化可讓 AWS Security Agent 從先前的五項測試執行中學習導覽模式，並將其套用至後續執行。這些模式包括登入流程和多步驟身分驗證工作流程。這可減少代理程式重新探索驗證方式所花費的時間，進而加快掃描速度並提高測試涵蓋範圍的一致性。

登入最佳化的運作方式

在第一次執行 pentest 時，代理程式會探索如何使用您提供的登入資料來導覽應用程式的登入流程。它會記錄成功的導覽步驟，並產生擷取所學登入工作流程的技能檔案。

後續執行時，代理程式會讀取儲存的技能檔案，並直接套用學習到的導覽模式，略過探索階段。這表示：

- 更快完成驗證測試 — 代理程式會立即套用經過驗證的導覽模式，而不是從頭開始探索
- 更一致的行為 — 代理程式遵循相同的經驗證路徑，而不是探索替代方案

> **Note**
>
> 登入最佳化會在執行中的第一個登入工作階段期間學習。相同執行中的後續登入工作階段將受益於第一個工作階段中學到的內容。未來執行時，所有登入工作階段都會受益於儲存的技能。

啟用或停用登入最佳化

登入最佳化預設為啟用。若要停用或重新啟用：

1. 在 pentest 組態中，導覽至身分驗證憑證 - 選用區段。
2. 找到登入最佳化切換。
3. 若要啟用登入最佳化，請開啟切換。若要停用，請將其關閉。

停用時，代理程式會在每次執行時從頭開始重新探索登入流程。先前學習的導覽技能會保留，如果重新啟用此功能，將會再次套用。

> **Tip**
>
> 如果您應用程式的登入流程發生重大變更（例如重新設計的登入頁面或新的身分驗證步驟），代理程式會在下次執行時更新其學到的技能來自動調整。您不需要手動重設最佳化。

修復滲透測試問題清單

檢視滲透測試的問題清單時，您可以請求 AWS Security Agent 嘗試修復問題清單。對於私有 GitHub 儲存庫，AWS Security Agent 會開啟包含建議修正的提取請求。對於公有儲存庫，修復可作為可下載的 diff 檔案，您可以在本機套用。

您必須在 AWS 管理主控台中啟用問題清單修復。(請參閱 the section called “讓使用者能夠開始修復滲透測試和程式碼檢閱問題清單”)。使用者可以從 AWS Security Agent Web 應用程式開始修復特定問題清單。

先決條件

開始前，請確保您具備以下條件：

- 已完成或進行中滲透測試執行
- 存取 AWS Security Agent Web 應用程式
- 熟悉應用程式的架構和安全性需求

步驟 1：啟用或停用自動修復

您可以在建立或修改滲透測試時設定程式碼修復選項。如果您啟用自動修復，則如果代理程式在測試期間確認問題清單，AWS Security Agent 將自動嘗試修復相關聯的 GitHub 儲存庫。您也可以手動啟動程式碼修復。。在編輯滲透測試詳細資訊的檢視中，在自動程式碼修復區段中，啟用或停用程式碼修復。

步驟 2：選取用於程式碼修復的儲存庫

1. 選擇 一直到最後一個步驟 其他學習資源。
2. 選擇從資源中選取。
3. 選擇 GitHub 儲存庫。

4. 選取您要用於程式碼修復的儲存庫。
5. 儲存滲透測試。
6. 您可以在滲透測試學習資源索引標籤下查看成功關聯的儲存庫。

步驟 3：開始滲透測試並檢視問題清單

執行滲透測試以偵測問題清單。如需詳細資訊，請參閱the section called “檢閱滲透測試的問題清單”。

步驟 4：啟動和檢視程式碼修復

1. 導覽至問題清單。
2. 如果您已啟用自動程式碼修復，一旦 AWS Security Agent 確認問題清單，就會開始程式碼修復。
3. 如果您想要手動啟動程式碼修復，請選擇修正程式碼按鈕。
4. 在調查結果的程式碼修復區段中，您可以檢視程式碼修復狀態和提取請求的連結。如果 GitHub 儲存庫是公有的，則程式碼修復可作為可下載的檔案使用，而不是提取請求。您可以執行 `git apply /path/to/code_remediation_changes.diff` ，將變更套用至本機的儲存庫。

安全性和參考

AWS Security Agent 的安全指導、服務配額和文件歷史記錄。

AWS Security Agent 中的安全

的雲端安全性 AWS 是最高優先順序。身為 AWS 客戶，您可以受益於資料中心和網路架構，這些架構專為滿足最安全敏感組織的需求而建置。

安全性是 AWS 與您之間的共同責任。共同責任模型 將此描述為雲端的安全和雲端內的安全：

- 雲端的安全性 – AWS 負責保護在 AWS 雲端中執行 AWS Security Agent 的基礎設施。這包括服務基礎設施、AI 模型和滲透測試代理程式。在 AWS 合規計畫中，第三方稽核員會定期測試並驗證我們的安全功效。
- 雲端內部的安全 – 您的責任包含下列領域：
  - 透過 IAM 政策和許可管理對 AWS Security Agent 的存取
  - 保護您提供給服務的內容，包括用於滲透測試的設計文件、程式碼儲存庫和應用程式 URLs
  - 設定要監控哪些儲存庫和應用程式
  - 檢閱和處理 服務提供的安全性問題清單
  - 根據提供的修補指引來保護您的應用程式
  - 資料的機密性、公司的要求，以及適用法律和法規

本文件可協助您了解如何在使用 AWS Security Agent 時套用共同責任模型。

AWS Security Agent 和 AI 輔助滲透測試的安全考量

AWS Security Agent 是一種前端代理程式，可在所有環境的開發生命週期中主動保護您的應用程式。它會執行根據您的需求自訂的自動化安全性審查，讓安全團隊集中定義在審查期間自動驗證的標準。Security Agent 會執行為您的應用程式自訂的隨需滲透測試，探索並報告已驗證的安全風險。此方法可將安全專業知識擴展至您的應用程式，以符合開發速度，同時提供全面的安全涵蓋範圍。透過將安全性從設計整合到部署，有助於及早大規模防止漏洞。

安全團隊會在 AWS 主控台中定義組織安全需求一次：核准的授權程式庫、記錄標準和資料存取政策。AWS Security Agent 會在開發過程中自動強制執行這些安全要求，根據您的標準評估架構文件和程式碼，並在偵測到違規時提供特定指引。這可跨團隊提供一致的安全強制執行，並擴展審核以符合開發速度。

為了進行部署驗證，AWS Security Agent 會將滲透測試從定期瓶頸轉換為隨需功能。安全團隊提供目標 URLs、身分驗證詳細資訊、原始程式碼和文件。AWS Security Agent 開發對應用程式的深入理解，並執行複雜的攻擊鏈來探索和驗證漏洞，讓團隊在需要時進行測試。

關鍵功能

AWS Security Agent 提供涵蓋整個開發生命週期的完整安全功能。

設計安全審查

AWS Security Agent 會提供設計文件的隨需安全意見回饋，並在編寫程式碼之前評估組織安全需求的合規性。安全團隊會透過 Web 應用程式上傳設計文件，其中代理程式會根據您的安全需求來分析這些文件，並使用修補指引來呈現調查結果。這可將長達數小時的手動審核加速為重點分析，讓團隊能夠在修補最有效率時解決安全問題。

程式碼安全性審查

AWS Security Agent 會分析提取請求或上傳的程式碼，以了解組織安全需求，以及缺少輸入驗證和 SQL 注入風險等常見安全問題。代理程式會直接在您的程式碼儲存庫平台內提供修補指引。安全團隊會設定要監控哪些儲存庫、擴展所有程式碼庫的評估，同時持續監督重大問題。

隨需滲透測試

AWS Security Agent 提供隨需滲透測試，透過量身打造的多步驟攻擊案例來探索和報告已驗證的安全漏洞。AWS Security Agent 部署專門的 AI 代理程式，從提供的文件和登入資料開發應用程式內容，然後執行複雜的攻擊鏈，以識別傳統工具遺漏的複雜漏洞。它使用影響分析、可重現的攻擊路徑和ready-to-implement程式碼修正來記錄調查結果，加速數週到數小時的滲透測試，以及跨應用程式產品組合進行擴展驗證。

FAQs

安全性與控制

AWS Security Agent 如何驗證和維護對系統的存取？

滲透測試是 AWS Security Agent 中唯一可在執行時間向使用者系統進行身分驗證的功能。AWS Security Agent 會接受靜態使用者名稱和密碼登入資料（儲存在 Secrets Manager 中）形式的登入資料，或登入資料供應商（做為 Lambda 函數）做為組態，然後再開始筆測試。這些登入資料用於在筆測試的生命週期內，執行使用者系統/應用程式的正常功能。我們鼓勵使用者建立具有適當範圍許可的新登入資料，以便進行滲透。

使用者是否可以控制測試的範圍和深度，以防止意外的系統影響？

AWS Security Agent 可讓客戶選取要在端點中探索的特定漏洞類別。使用者可以指定out-of-scopeURLs，以防止 AWS Security Agent 對這些目標執行滲透測試。https：//https://docs.aws.amazon.com/securityagent/latest/userguide/perform-penetration-test.html

AWS Security Agent 本身是否會帶來安全風險？

系統會指示 AWS Security Agent 探索安全風險，但使用刻意影響最小的承載（例如擷取 SQL 版本，而不是在發現 SQL 注入攻擊時捨棄資料表）。AWS Security Agent 也受限於確定性護欄，以防止風險行為，例如對目標應用程式造成過多負載。雖然有護欄，但仍可能有無意或非明顯的商業邏輯互動，因此，我們一律建議針對生產前環境進行滲透測試。

AWS Security Agent 會收集哪些資料及其存放位置？

AWS Security Agent 允許使用者上傳成品，以提供其正在測試之應用程式的內容。如需資料保護的詳細資訊，請參閱the section called “資料保護”。AWS Security Agent 會自動選取您地理位置內的最佳區域，以處理您的推論請求。這可將可用的運算資源、模型可用性最大化，並提供最佳客戶體驗。您的資料只會存放在發出請求的區域，但輸入提示和輸出結果可能會在該區域之外處理。所有資料都會透過 Amazon 的安全網路進行加密傳輸。如需詳細資訊，請參閱跨區域推論。

有哪些控制項可以阻止對端點進行未經授權的測試？

指定為滲透目標 URLs端點將需要 DNS 驗證或 HTTP 驗證作為擁有權的指標。AWS Security Agent 會要求客戶將 TXT 記錄新增至端點的 DNS，或公開 HTTP Route 傳回驗證字串作為擁有權證明。只有在證明擁有權之後，使用者才能繼續進行 pentest。網路將會封鎖對目標和可存取 URLs 外部 URLs的請求。

客戶負責確保他們擁有適當的授權，以測試可能受其滲透測試活動影響的所有系統。所有 AWS Security Agent 的使用都必須遵守 AWS 可接受的使用政策 (https://aws.amazon.com/aup/：//)。

使用者如何使用 AWS Security Agent 封鎖和報告任何濫用？

AWS Security Agent 會持續監控請求，並嘗試存取目標 URLs 以外的 URL。如果偵測到濫用，例如嘗試使用 AWS 安全代理程式對第三方端點進行未經授權的測試，帳戶中的任何進行中滲透都將終止。客戶可以聯絡 AWS Support 或其 AWS 客戶團隊尋求協助。

AWS Security Agent 是否可以取代筆測試工作流程？

AWS Security Agent 不是專業的滲透測試服務，我們鼓勵使用者將 AWS Security Agent 整合到其安全性審查工作流程中。AWS Security Agent 可以在軟體生命週期的開發階段提供隨需滲透測試的可存取性，因為與滲透專業人員互動太早、不切實際或需要重新評估太頻繁。安全專業人員可以從 AWS

Security Agent 檢閱問題清單，以驗證問題清單、解釋問題清單，或針對新問題清單（如果有的話）延伸問題清單。

使用者可以為不同的團隊成員設定角色型存取控制 (RBAC) 嗎？

是。AWS Security Agent 與 AWS IAM Identity Center 整合，可讓管理員管理可存取 AWS Security Agent Web 應用程式的團隊成員，以允許使用者建立、管理和檢視設計檢閱和滲透測試。

測試功能

AWS Security Agent 可以偵測哪些類型的漏洞？

AWS Security Agent 會偵測 Web 應用程式 OWASP 前 10 名中的漏洞。AWS Security Agent 提供您可以在下列測試中包含或排除的特定風險類型。問題清單可能會出現在這些風險類別中，或從這些風險類別組合中跟隨潛在客戶所發現的新問題清單。

- 任意檔案上傳
  - 任意檔案上傳會確認應用程式應該能夠防禦假名和惡意檔案，以保護應用程式和使用者的安全
- 程式碼注入
  - Code Injection 是攻擊類型的一般術語，包含注入由應用程式解譯/執行的程式碼
- 命令注入
  - 命令注入是一種攻擊，其目標是透過易受攻擊的應用程式在主機作業系統上執行任意命令
- 跨網站指令碼 (XSS)
  - 跨網站指令碼 (XSS) 攻擊是一種注入，惡意指令碼會注入至其他良性和信任的網站
- 不安全的直接物件參考
  - 當應用程式根據使用者提供的輸入提供直接存取物件時，會發生不安全的直接物件參考 (IDOR)
- JSON Web 權杖漏洞
  - JWTs是常見的漏洞來源，包括應用程式和基礎程式庫中的漏洞實作方式
- 本機檔案包含
  - 檔案包含漏洞可讓攻擊者包含檔案，通常利用目標應用程式中實作的「動態檔案包含」機制
- 路徑周遊
  - 路徑周遊攻擊（也稱為目錄周遊）旨在存取存放在 Web 根資料夾外部的檔案和目錄
- 權限提升
  - 當使用者存取比平常允許的更多資源或功能，且應用程式應該防止這類提升或變更時，就會發生權限提升

- 伺服器端請求偽造 (SSRF)
  - 伺服器端請求偽造 (SSRF) 會在攻擊者可能濫用伺服器上的功能來讀取或更新內部資源時發生
- 伺服器端範本注入
  - 伺服器端範本注入漏洞 (SSTI) 會在使用者輸入以不安全的方式內嵌在範本中，並在伺服器上導致遠端程式碼執行時發生
- SQL Injection
  - SQL Injection 攻擊包含透過從用戶端到應用程式的輸入資料插入或「注入」SQL 查詢
- XML 外部實體
  - XML 外部實體攻擊是一種針對剖析 XML 輸入的應用程式的攻擊。當包含外部實體參考的 XML 輸入由設定較弱的 XML 剖析器處理時，就會發生此攻擊

AWS Security Agent 支援哪些身分驗證方法？

AWS Security Agent 支援常見的身分驗證方法，包括 OAuth 和 JWT。如需詳細資訊，請參閱 文件。

AWS Security Agent 如何處理速率限制和阻斷服務 (DOS) 預防？

AWS Security Agent 具有防護機制，可防止其中斷或關閉測試中的端點，包括 DOS。它具有內部速度控制，可偵測和處理非預期的流量模式。

AWS Security Agent 可以同時測試 REST 和 GraphQL APIs嗎？

是，AWS Security Agent 可以測試 API 端點。我們鼓勵客戶以其他學習資源的形式提供 API 文件，讓 AWS Security Agent 能夠更完善地了解所測試每個 API 的形狀和功能。

使用者如何驗證 AWS Security Agent 已涵蓋所有重要的應用程式邏輯和端點？

AWS Security Agent 會先對目標應用程式進行廣度探索（並在嘗試任何入侵之前嘗試正常執行）。這可讓它在執行時間建立對應用程式的工作了解，並探索重要的應用程式邏輯和端點。鑑於其隨機性質，AWS Security Agent 不保證會探索和測試任何目標應用程式的所有關鍵應用程式和端點。AWS Security Agent Web 應用程式可讓您查看所有探索到的端點，以及在滲透測試日誌中採取的動作。

準確性和可靠性

AWS Security Agent 如何在報告之前驗證問題清單？

AWS Security Agent 使用確定性驗證程式來協助驗證報告的問題清單。在無法使用確定性驗證程式的風險類型中，AWS Security Agent 將獨立重播調查結果步驟，以獲得對調查結果有效性的信心。AWS Security Agent 只會報告高或中可信度問題清單，並依預設隱藏未驗證的問題清單。

AWS Security Agent 可以適應自訂應用程式邏輯嗎？

AWS Security Agent 可選擇接受原始碼、威脅模型、設計文件和 API 文件作為其他學習資源，以取得使用者導向的內容，用於滲透測試生命週期中所使用的目標應用程式。

使用者可以在執行之前檢閱 AWS Security Agent 測試方法嗎？

目前無法預覽 AWS Security Agent 的動作過程。AWS Security Agent 計劃本質上是動態的，以其探索目標應用程式為基礎。客戶可以透過觀察滲透測試日誌，即時監控 AWS Security Agent 探索。如果日誌顯示無效或不想要的軌跡，客戶可以停止持續的滲透執行。

整合與部署

AWS Security Agent 是否與安全工具 (SIEM、漏洞管理） 或 CI/CD 管道整合？

AWS Security Agent 不會與任何現有的安全工具或 CI/CD 管道整合。

AWS Security Agent 如何處理環境特定的組態？

AWS Security Agent 可設定為使用特定 IAM 角色、VPCs 內部、使用客戶指定的應用程式相關登入資料，以及使用 Github 來源儲存庫做為目標應用程式的原始碼參考。

AWS Security Agent 可以在氣隙隔離或隔離的環境中執行嗎？

AWS Security Agent 可以設定為與 VPCs連線，包括沒有傳出網際網路存取的 VPC。

多個團隊成員可以同時執行測試嗎？

AWS Security Agent 支援每個帳戶 5 個並行 pentest 執行，與誰開始測試無關。客戶最多可以建立 100 個客服人員空間和 1，000 個 Pentest 專案。

營運影響

對已測試系統的效能有何影響？

AWS Security Agent 具有防護機制，以防止其中斷或關閉測試中的端點。這包括 AWS Security Agent 可對端點進行呼叫次數的速度控制。系統或測試中的端點預期會因為筆測試活動而觸發一些流量增加和潛在的監控提醒。我們建議只在生產前環境中執行 AWS Security Agent 或任何筆測試活動。

使用者可以排程或調節 AWS Security Agent 嗎？

AWS Security Agent 沒有公有 APIs 或能夠排程筆測試執行。啟動筆測試執行時，AWS Security Agent 也不會對目標端點的請求提供並行控制。如果 AWS Security Agent 導致目標端點發生問題，客戶可以停止持續的 pentest(s)。

完整安全性評估的典型持續時間為何？

每個五項測試的執行時間取決於目標應用程式的廣度，以及設定要評估的風險類型。大多數 pentest 執行會在 16 小時內完成。

AWS Security Agents 的 AWS 受管政策

AWS 托管策略 是由 AWS 创建和管理的独立策略。AWS 受管政策旨在為許多常用案例提供許可，以便您可以開始將許可指派給使用者、群組和角色。

請記住，AWS 受管政策可能不會授予特定使用案例的最低權限許可，因為這些許可可供所有 AWS 客戶使用。我們建議您定義特定於使用案例的客戶管理政策，以便進一步減少許可。

您無法變更 AWS 受管政策中定義的許可。如果 AWS 更新 AWS 受管政策中定義的許可，則更新會影響政策連接的所有主體身分（使用者、群組和角色）。當新的 AWS 服務啟動或新的 API 操作可用於現有服務時，AWS 最有可能更新 AWS 受管政策。

如需詳細資訊，請參閱《IAM 使用者指南》中的 AWS 受管政策。

AWS 受管政策：SecurityAgentWebAppAPIPolicy

准許與 Security Agent Web Application API 互動。此政策可讓使用者設定和執行自動化安全滲透測試、管理測試執行、檢視安全調查結果，以及存取 Security Agent 資源。此政策參考舊版 代理程式執行個體資源類型和特定的舊版 IAM 動作。

許可詳細資訊

此政策授予許可，以與 Security Testing Control 服務 (Securityagent：*) 互動：

- 滲透管理：建立、更新、刪除和列出滲透測試及其執行任務
- 安全性調查結果：檢視、描述和更新已完成測試的安全性調查結果，包括相關內容和中繼資料。
- 任務管理：列出和擷取程式碼檢閱和文件檢閱任務
- 資源探索：列出和檢視客服人員執行個體、成品、整合和探索的端點
- 測試執行：使用即時監控功能啟動和停止 pentest 執行
- 程式碼修復：啟動安全調查結果的自動程式碼修復

若要檢視最新版本的 JSON 政策文件，請參閱《AWS 受管政策參考指南》中的 SecurityAgentWebAppAPIPolicy。

AWS 受管政策：AWSSecurityAgentWebAppPolicy

准許與 Security Agent Web Application API 互動。此政策可讓使用者設定和執行自動化安全滲透測試、管理測試執行、檢視安全調查結果，以及存取 Security Agent 資源。

許可詳細資訊

此政策授予許可，以與 Security Testing Control 服務 (Securityagent：*) 互動：

- 滲透管理：建立、更新、刪除和列出滲透測試及其任務
- 安全性調查結果：檢視、描述和更新已完成測試的安全性調查結果，包括相關內容和中繼資料。
- 任務管理：列出和擷取程式碼檢閱和設計檢閱任務
- 資源探索：列出和檢視代理程式空間、成品、整合和探索的端點
- 測試執行：使用即時監控功能啟動和停止 pentest 任務
- 程式碼修復：啟動安全調查結果的自動程式碼修復

若要檢視最新版本的 JSON 政策文件，請參閱《AWS 受管政策參考指南》中的 AWSSecurityAgentWebAppPolicy。

AWS Security Agents 更新 AWS 受管政策

檢視自此服務開始追蹤這些變更以來，AWS Security Agents 的 AWS 受管政策更新詳細資訊。

如需接收有關此特定文件頁面所有來源檔案變更的通知，您可透過 RSS 閱讀器來訂閱以下 URL：

變更	描述	日期
新增 AWSSecurityAgentWebAppPolicy 的許可。	新增新DesignReviewFeedback 資源類型的 TargetDomain 和資源許可。	2026 年 3 月 31 日
新增了新的受管政策 AWSSecurityAgentWebAppPolicy。	新增新 AgentSpace 資源類型和 IAM 動作名稱變更AWSSecurityAgentWebAppPolicy 的受管政策。	2026 年 2 月 9 日
新增 SecurityAgentWebAppAPIPolicy 的許可。	新增 securityagent:StartCodeRemediation 以允	2026 年 1 月 20 日

變更	描述	日期
	許使用者開始自動程式碼修復安全性問題清單。	
新增 SecurityAgentWebAppAPIPolicy 的許可。	新增`securityagent:BatchGetSecurityTestContentMetadata` 以允許使用者在 主控台中檢視映像。	2025 年 12 月 5 日
AWS Security Agents 開始追蹤變更。	AWS Security Agents 開始追蹤其 AWS 受管政策的變更。	2025 年 12 月 2 日

AWS Security Agent 中的資料保護

AWS 共同責任模型適用於 AWS Security Agent 中的資料保護。如此模型所述，AWS 負責保護執行所有 AWS Cloud 的全球基礎設施。您負責維護在此基礎設施上託管內容的控制權。您也必須負責您使用之 AWS 服務的安全組態和管理任務。如需歐洲資料保護的相關資訊，請參閱 AWS 安全部落格上的 AWS 共同責任模型和 GDPR 部落格文章。基於資料保護目的，我們建議您保護 AWS 帳戶登入資料，並使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用 AWS CloudTrail 設定 API 和使用者活動記錄日誌。如需有關使用 CloudTrail 追蹤擷取 AWS 活動的資訊，請參閱《AWS CloudTrail 使用者指南》中的使用 CloudTrail 追蹤。 CloudTrail
- 使用 AWS 加密解決方案，以及 AWS 服務內的所有預設安全控制。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果您在透過命令列界面或 API 存取 AWS 時需要 FIPS 140-3 驗證的密碼編譯模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，請參閱聯邦資訊處理標準 (FIPS) 140-3。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您使用主控台、API、AWS CLI 或 AWS SDKs來使用 AWS Security Agent 或其他 AWS 服務時。您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日

誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

靜態加密

AWS Security Agent 預設會使用 AWS 受管加密金鑰來加密所有靜態資料。其中包含：

- 設計文件和程式碼 – 您為安全性審查提供的所有設計文件、程式碼儲存庫和應用程式成品都會使用 AES-256 加密進行加密。
- 安全性問題清單 – 所有安全性問題清單、漏洞報告和修補建議都會靜態加密。
- 組態資料 – 安全需求、自訂政策和服務組態都會加密。
- 稽核日誌 – 所有服務活動日誌和稽核追蹤都會加密。

AWS Security Agent 使用 AWS Key Management Service (AWS KMS) 來管理加密金鑰。您可以選擇性地使用客戶受管金鑰來加密資料，讓您完全控制保護 資源的加密金鑰。如需詳細資訊，請參閱the section called “客戶自管金鑰”。

傳輸中加密

AWS Security Agent 會使用 Transport Layer Security (TLS) 1.2 或更高版本來加密傳輸中的所有資料。這適用於：

- API 通訊 – 應用程式與 AWS Security Agent 之間的所有 API 呼叫都會使用 HTTPS 搭配 TLS 加密。
- 主控台存取 – AWS Security Agent 主控台是透過 HTTPS 存取。
- 儲存庫連線 – 與 GitHub 和其他程式碼儲存庫的連線使用加密通訊協定。
- 代理程式通訊 – AWS Security Agent 服務和滲透測試代理程式之間的所有通訊都使用加密通道。

金鑰管理

AWS Security Agent 使用 AWS Key Management Service (AWS KMS) 來管理加密金鑰。根據預設，資料會使用 AWS 受管金鑰加密。您可以在建立 Agent Spaces 和整合等資源時選擇性地指定客戶受管金鑰。如需詳細資訊，請參閱the section called “客戶自管金鑰”。

網際網路流量隱私權

AWS Security Agent 使用公有網際網路與雲端託管來源控制提供者 (GitHub、GitLab、Bitbucket) 和 Confluence Cloud 通訊。

對於自我託管提供者 (GitLab 自我管理，GitHub Enterprise Server)，您可以使用 Amazon VPC Lattice 設定私有連線，以將所有流量保留在 AWS 網路中。如需詳細資訊，請參閱the section called “連線至私有託管來源控制”。

在預設組態中，AWS Security Agent 會使用公有網際網路來連接您的應用程式以進行滲透測試。您可以選擇性地設定滲透測試，以使用 VPC 存取您的應用程式。如需詳細資訊，請參閱the section called “將代理程式連線至私有 VPC 資源”。

跨區域資料處理

AWS Security Agent 使用跨區域推論來最佳化可用的運算資源和模型可用性。根據請求的來源區域，我們可能會在不同的區域中處理輸入提示和輸出結果。

- 在美國東部（維吉尼亞北部）– us-east-1、美國西部（奧勒岡）– us-west-2、亞太區域（雪梨）– ap-southeast-2、亞太區域（東京）– ap-northeast-1、歐洲（法蘭克福）– eu-central-1和歐洲（愛爾蘭）– 中eu-west-1，AWS 安全代理程式使用地理跨區域推論。對於大多數功能，資料處理會保留在發出請求的地理界限（例如美國、歐洲、澳洲或日本）內。對於程式碼修復，來自澳洲和日本的請求會在歐盟處理。如需功能特定的路由詳細資訊，請參閱跨區域推論表。
- 在亞太區域（孟買）– ap-south-1、亞太區域（新加坡）– ap-southeast-1和南美洲（聖保羅）– sa-east-1中，AWS 安全代理程式使用全球跨區域推論。我們可能會在任何商業 AWS 區域中處理輸入提示和輸出結果。

在所有情況下，您的資料只會存放在發出請求的區域中。跨區域操作期間傳輸的所有資料都會保留在 AWS 網路上，而不會周遊公有網際網路。我們會加密 AWS 區域之間傳輸中的資料。

跨區域推論一律啟用，且無法選擇退出。如需哪些區域處理每個功能請求的詳細資訊，請參閱 中的跨區域推論一節the section called “安全最佳實務”。

資料刪除

當您從 AWS Security Agent 刪除資料時：

- 資料會標記為刪除，且無法再透過 服務存取。
- 資料會在 30 天內從所有 AWS Security Agent 系統刪除。

刪除您的資料

1. 在 AWS 主控台中，導覽至 AWS Security Agent。

2. 選擇您要刪除的資料（安全性檢閱、問題清單或自訂需求）。
3. 選擇刪除並確認刪除。

AWS Security Agent 的身分和存取管理

AWS Identity and Access Management (IAM) 是 AWS 服務 ，可協助管理員安全地控制對 AWS resources 的存取。 IAM 管理員可控制誰可以經過身分驗證 （登入） 和授權 （具有許可） 來使用 AWS Security Agent 資源。 IAM 是您可以免費使用 AWS 服務 的 。

目標對象

使用方式 AWS Identity and Access Management (IAM) 會有所不同，取決於您在 AWS Security Agent 中執行的工作。

服務使用者 – 如果您使用 AWS Security Agent 服務來執行任務，您的管理員會為您提供所需的登入資料和許可。當您使用更多 AWS Security Agent 功能來執行工作時，您可能需要額外的許可。了解存取許可的管理方式可協助您向管理員請求正確的許可。

如果您無法存取 AWS Security Agent 中的功能，請參閱 the section called “故障診斷 AWS Security Agent 身分和存取”。

服務管理員 - 如果您在公司負責 AWS Security Agent 資源，您可能擁有 AWS Security Agent 的完整存取權。您的任務是判斷服務使用者應存取的 AWS Security Agent 功能和資源。然後，您必須向 IAM 管理員提交請求，以變更服務使用者的許可。檢閱此頁面上的資訊，以了解 的基本概念 IAM。若要進一步了解貴公司如何 IAM 搭配 AWS Security Agent 使用 ，請參閱 the section called “AWS Security Agent 如何使用 IAM”。

IAM 管理員 - 如果您是 IAM 管理員，建議您進一步了解如何撰寫政策以管理對 AWS Security Agent 的存取。若要檢視您可以在 中使用的 AWS Security Agent 身分型政策範例 IAM，請參閱 the section called “AWS Security Agent 身分型政策範例”。

使用身分驗證

身分驗證是您 AWS 使用身分憑證登入 的方式。您必須擔任 IAM 角色，以 AWS 帳戶根使用者 IAM 使用者、 或 身分進行身分驗證 （登入 AWS)。AWS 建議使用 IAM 角色，並避免直接使用根使用者。

您可以使用透過身分來源提供的登入資料，以聯合身分 AWS 形式登入 。 AWS IAM Identity Center (IAM Identity Center) 使用者、您公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料，都是聯合身分的範例。當您以聯合身分身分登入時，您的管理員先前會使用 IAM 角色設定聯合身分。當您使用聯合 AWS 身分存取 時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需登入的詳細資訊 AWS，請參閱《AWS 登入使用者指南》中的如何登入您的 AWS 帳戶。

如果您以 AWS 程式設計方式存取 ， AWS 會提供軟體開發套件 (SDK) 和命令列界面 (CLI)，以使用您的 憑證以密碼編譯方式簽署您的請求。如果您不使用 AWS 工具，則必須自行簽署請求。如需使用建議方法自行簽署請求的詳細資訊，請參閱 AWS 一般參考中的 Signature 第 4 版簽署程序。

無論您使用何種身分驗證方法，您可能還需要提供額外的安全性資訊。例如， AWS 建議您使用多重驗證 (MFA) 來提高帳戶的安全性。若要進一步了解，請參閱《AWS IAM Identity Center (AWS Single Sign-On 的後續） 使用者指南》中的多重要素驗證，以及《IAM 使用者指南》中的在 AWS 中使用多重要素驗證 (MFA)。

AWS 帳戶根使用者

當您第一次建立 時 AWS 帳戶，您會從單一登入身分開始，該身分可完整存取帳戶中的所有 AWS 服務和資源。此身分稱為 AWS 帳戶「根使用者」，是藉由您用來建立帳戶的電子郵件地址和密碼以登入並存取。強烈建議您不要以根使用者處理日常作業。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需需要您以根使用者身分登入的任務完整清單，請參閱《帳戶管理參考指南》中的需要根使用者憑證的任務。

聯合身分

最佳實務是， 要求人類使用者，包括需要管理員存取權的使用者，使用臨時 AWS 服務 登入資料與身分提供者聯合來存取 。

聯合身分是來自您的企業使用者目錄、Web 身分提供者、 AWS Directory Service、 Identity Center 目錄或任何使用透過身分來源提供的登入資料 AWS 服務 存取的使用者。當聯合身分存取時 AWS 帳戶，它們會擔任 角色，而角色會提供臨時登入資料。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連接並同步到您自己的身分來源中的一組使用者 AWS 帳戶 和群組，以便在所有 和應用程式中使用。如需 IAM Identity Center 的資訊，請參閱什麼是 IAM Identity Center？《AWS IAM Identity Center (AWS Single Sign-On 的後續版本） 使用者指南》中的 。

IAM 使用者 和 群組

IAM 使用者 是您 中的身分 AWS 帳戶 ，具有單一人員或應用程式的特定許可。如果可能，我們建議依賴臨時登入資料，而不是建立擁有密碼和存取金鑰等長期登入資料 IAM 使用者 的人員。不過，如果您有特定的使用案例需要使用長期憑證 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱《IAM 使用者指南》中的為需要長期憑證的使用案例定期輪換存取金鑰。

IAM 群組是指定 集合的身分 IAM 使用者。您無法以群組身分登入。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有名為 IAMAdmins 的群組，並授予該群組管理 IAM 資源的許可。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供臨時憑證。若要進一步了解，請參閱《IAM 使用者指南》中的何時建立 IAM 使用者 （ 而非角色 ）。

IAM 角色

IAM 角色是 中具有特定許可 AWS 帳戶 的身分。它類似於 IAM 使用者，但不與特定人員相關聯。您可以 AWS Management Console 切換 IAM 角色，暫時在 中擔任 角色。您可以透過呼叫 AWS CLI 或 AWS API 操作或使用自訂 URL 來擔任角色。如需使用角色方法的詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM 角色。

IAM 具有臨時登入資料的 角色在下列情況下非常有用：

- 聯合身分使用者存取 – 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱 IAM 使用者指南中的為第三方身分提供者建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM中的角色相關聯。如需有關許可集的資訊，請參閱《AWS IAM Identity Center (AWS Single Sign-On 的後續） 使用者指南》中的許可集。
- 暫時 IAM 使用者 許可 – IAM 使用者 可以擔任 IAM 角色，暫時接受特定任務的不同許可。
- 跨帳戶存取 – 您可以使用 IAM 角色，允許不同帳戶中的某人 （ 信任的委託人 ） 存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。不過，對於某些 AWS 服務，您可以將政策直接連接到資源 （ 而不是使用角色做為代理 ）。若要了解跨帳戶存取的角色和資源型政策之間的差異，請參閱《IAM 使用者指南》中的 IAM 角色與資源型政策的差異。
- 跨服務存取 – 有些 AWS 服務 使用其他 中的功能 AWS 服務。例如，當您在服務中呼叫 時，該服務通常會在 中執行應用程式 Amazon EC2 或將物件存放在其中 Amazon S3。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 委託人許可 – 當您使用 IAM 使用者 或 角色在 中執行動作時 AWS，您會被視為委託人。政策能將許可授予主體。當您使用某些服務時，您可能會執行一個動作，然後在不同的服務中觸發另一個動作。在此情況下，您必須具有執行這兩個動作的許可。
  - 服務角色 – 服務角色是服務擔任以代表您執行動作 IAM 的角色。 IAM 管理員可以從內部建立、修改和刪除服務角色 IAM。如需詳細資訊，請參閱 IAM 使用者指南中的建立角色以委派許可給 AWS 服務服務。

- 服務連結角色 – 服務連結角色是連結至 的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶 ，並由服務擁有。 IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 上執行 Amazon EC2 的應用程式 – 您可以使用 IAM 角色來管理在 Amazon EC2 執行個體上執行之應用程式的臨時登入資料，以及提出 AWS CLI 或 AWS API 請求。最好將存取金鑰存放在 Amazon EC2 執行個體中。若要將 AWS 角色指派給 Amazon EC2 執行個體並將其提供給其所有應用程式，您可以建立連接至執行個體的執行個體描述檔。執行個體描述檔包含 角色，並可讓在 Amazon EC2 執行個體上執行的程式取得臨時登入資料。如需詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM 角色將許可授予在 Amazon EC2 執行個體上執行的應用程式。

若要了解如何使用 IAM 角色，請參閱《IAM 使用者指南》中的何時建立 IAM 角色 （ 而非使用者 ）。

使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策是 中的物件， AWS 當與身分或資源建立關聯時， 會定義其許可。當委託人 （ 使用者、根使用者或角色工作階段 ） 發出請求時， 會 AWS 評估這些政策。政策中的許可決定是否允許或拒絕請求。大多數政策會以 JSON 文件 AWS 形式存放在 中。如需 JSON 政策文件結構和內容的詳細資訊，請參閱《IAM 使用者指南》中的 JSON 政策概觀。

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

每個 IAM 實體 （ 使用者或角色 ） 都從沒有許可開始。根據預設，使用者什麼都不做，甚至無法變更自己的密碼。若要授予使用者執行動作的許可，管理員必須將許可政策附加到使用者。或者，管理員可以將使用者新增到具備預定許可的群組。當管理員將許可授予群組時，該群組中的所有使用者都會獲得這些許可。

IAM 無論您用來執行操作的方法為何， 政策都會定義動作的許可。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該政策的使用者可以從 AWS Management Console AWS CLI、 或 AWS API 取得角色資訊。

身分型政策

身分型政策是您可以連接到身分的 JSON 許可政策文件，例如 IAM 使用者、 角色或 群組。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策直接嵌入單一使用者、群組或角色。受管政策是獨立的政策，您可以連接到 中的多個使用者、群組和角色 AWS 帳戶。受管政策包括 AWS 受管政策和客戶受管政策。如需了解如何在受管政策及內嵌政策間選擇，請參閱 IAM 使用者指南中的在受管政策和內嵌政策間選擇。

資源型政策

以資源為基礎的政策是您連接到資源的 JSON 政策文件，例如 儲存 Amazon S3 貯體。服務管理員可使用這些政策來定義指定委託人 (帳戶成員、使用者或角色) 可以在什麼情況下對該資源執行什麼動作。資源型政策是內嵌政策。不存在受管的資源型政策。

存取控制清單 (ACL)

存取控制清單 (ACL) 是可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源之許可的政策類型。ACLs 類似於以資源為基礎的政策，雖然它們不使用 JSON 政策文件格式。 Amazon S3 AWS WAF， 和 Amazon VPC 是支援 ACLs的服務範例。如需進一步瞭解 ACL，請參閱《Amazon Simple Storage Service 開發人員指南》中的存取控制清單 (ACL) 概觀。

其他政策類型

AWS 支援其他較不常見的政策類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 許可界限 – 許可界限是一種進階功能，您可以在其中設定身分型政策可授予 IAM 實體 (IAM 使用者或角色) 的最大許可。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 `Principal` 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。任何這些政策中的明確拒絕都會覆寫允許。如需許可界限的詳細資訊，請參閱《IAM 使用者指南》中的IAM 實體的許可界限。
- 服務控制政策 SCPs) – SCPs是 JSON 政策，可指定 中組織或組織單位 (OU) 的最大許可 AWS Organizations。 AWS Organizations 是一種用於分組和集中管理您企業擁有 AWS 帳戶 的多個 的服務。若您啟用組織中的所有功能，您可以將服務控制政策 (SCP) 套用到任何或所有帳戶。SCP 會限制成員帳戶中實體的許可，包括每個 AWS 帳戶根使用者。如需 Organizations 和 SCPs 的詳細資訊，請參閱《AWS Organizations 使用者指南》中的 SCPs 的運作方式。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過撰寫程式的方式建立角色或聯合身分使用者的暫時工作階段時，做為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參《IAM 使用者指南》中的工作階段政策。

多種政策類型

當多種類型的政策適用於請求時，產生的許可會更複雜而無法理解。若要了解如何 AWS 在涉及多個政策類型時決定是否允許請求，請參閱《IAM 使用者指南》中的政策評估邏輯。

AWS Security Agent 如何使用 IAM

在您使用 IAM 管理對 AWS Security Agent 的存取之前，請先了解哪些 IAM 功能可與 AWS Security Agent 搭配使用。

IAM 功能	AWS Security Agent 支援
the section called “AWS Security Agent 的身分型政策”	是
the section called “AWS Security Agent 中的資源型政策”	否
the section called “AWS Security Agent 的政策動作”	是
the section called “AWS Security Agent 的政策資源”	部分
the section called “AWS Security Agent 的政策條件索引鍵”	是
the section called “AWS Security Agent 中的存取控制清單 (ACLs)”	否
the section called “使用 AWS Security Agent 的屬性型存取控制 (ABAC)”	否
the section called “搭配 AWS Security Agent 使用臨時登入資料”	是
the section called “轉送 AWS Security Agent 的存取工作階段”	是

IAM 功能	AWS Security Agent 支援
the section called “AWS Security Agent 的服務角色”	否
the section called “AWS Security Agent 的服務連結角色”	是

若要全面了解 AWS Security Agent 和其他 AWS 服務 的使用方式 IAM，請參閱《IAM 使用者指南》中的 AWS 服務 使用 IAM 。

AWS Security Agent 的身分型政策

支援身分型政策：是

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的透過客戶管理政策定義自訂 IAM 許可。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及允許或拒絕動作的條件。您無法在身分型政策中指定委託人，因為它適用於其連接的 使用者或角色。若要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素參考。

AWS Security Agent 的身分型政策範例

若要檢視 AWS Security Agent 身分型政策的範例，請參閱 the section called “AWS Security Agent 身分型政策範例”。

AWS Security Agent 中的資源型政策

支援資源型政策：否

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。委託人可以包括帳戶、使用者、角色、聯合身分使用者或 AWS Services。

如需啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，做為資源型政策的主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當委託人和資源位於不同的 AWS 帳戶中

時，信任帳戶中的 IAM 管理員也必須授予委託人實體（使用者或角色）存取資源的許可。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的跨帳戶資源存取。

AWS Security Agent 的政策動作

支援動作 是

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

IAM 身分型政策的 `Action`元素說明政策允許或拒絕的特定動作。政策動作通常具有與相關聯 AWS API 操作相同的名稱。政策會使用動作來授予執行相關聯操作的許可。

AWS Security Agent 中的政策動作在動作之前使用下列字首：`securityagent:`。例如，若要授予某人使用 AWS Security Agent `CreateEnvironment` API 操作建立環境的許可，請在其政策中包含 `securityagent:CreateEnvironment`動作。政策陳述式必須包含 `Action` 或 `NotAction` 元素。AWS Security Agent 會定義自己的一組動作，描述您可以使用此服務執行的任務。

若要在單一陳述式中指定多個動作，請用逗號分隔，如下所示：

```
"Action": [
      "securityagent:action1",
      "securityagent:action2"
```

您也可以使用萬用字元 (*) 來指定多個動作。例如，若要指定開頭是 `List` 文字的所有動作，請包含以下動作：

```
"Action": "securityagent:List*"
```

AWS Security Agent 的政策資源

支援政策資源：部分

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

`Resource` JSON 政策元素可指定要套用動作的物件。陳述式必須包含 `Resource` 或 `NotResource` 元素。最佳實務是使用其 Amazon Resource Name (ARN) 來指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (*) 來表示陳述式適用於所有資源。

```
"Resource": "*"
```

有些 AWS Security Agent API 動作支援多個資源。例如，呼叫 ListEnvironments API 動作時，可以參考多個環境。若要在單一陳述式中指定多項資源，請使用逗號分隔 ARN。

```
"Resource": [
      "EXAMPLE-RESOURCE-1",
      "EXAMPLE-RESOURCE-2"
```

例如，AWS Security Agent 環境資源具有下列 ARN：

```
arn:${Partition}:securityagent:${Region}:${Account}:environment/${EnvironmentId}
```

若要在陳述式my-environment-2中指定環境my-environment-1和 ，請使用下列範例 ARNs：

```
"Resource": [
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-1",
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-2"
```

若要指定屬於特定帳戶的所有環境，請使用萬用字元 (*)：

```
"Resource": "arn:aws:securityagent:us-east-1:123456789012:environment/*"
```

AWS Security Agent 的政策條件索引鍵

支援服務特定政策條件金鑰：是

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 （或Condition區塊） 可讓您指定陳述式生效的條件。Condition 元素是選用項目。您可以建立使用條件運算子的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵， AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值， 會使用邏輯OR操作 AWS 評估條件。在授予陳述式的許可之前，必須符合所有條件。

您也可以在指定條件時使用預留位置變數。例如，只有在資源以其 IAM 使用者 名稱加上標籤時，您才能授予存取資源的 IAM 使用者 許可。如需詳細資訊，請參閱《IAM 使用者指南》中的IAM 政策元素：變數和標籤。

AWS Security Agent 會定義自己的一組條件金鑰，也支援使用一些全域條件金鑰。若要查看所有 AWS 全域條件索引鍵，請參閱《IAM 使用者指南》中的AWS 全域條件內容索引鍵。

AWS Security Agent 中的存取控制清單 (ACLs)

支援 ACL：否

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

使用 AWS Security Agent 的屬性型存取控制 (ABAC)

支援 ABAC (政策中的標籤)：否

搭配 AWS Security Agent 使用臨時登入資料

支援臨時憑證：是

當您使用臨時登入資料登入時，某些 AWS 服務無法運作。如需詳細資訊，包括哪些 AWS Services 使用臨時登入資料，請參閱《IAM 使用者指南》中的使用 IAM 的 AWS 服務。

如果您使用使用者名稱和密碼以外的任何方法登入 AWS 管理主控台，則會使用臨時登入資料。例如，當您使用公司的單一登入 (SSO) 連結存取 AWS 時，該程序會自動建立臨時登入資料。當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需切換角色的詳細資訊，請參閱《IAM 使用者指南》中的從使用者切換至 IAM 角色 (主控台)。

您可以使用 AWS CLI 或 AWS API 手動建立臨時登入資料。然後，您可以使用這些臨時登入資料來存取 AWS。AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱 IAM 中的暫時性安全憑證。

轉送 AWS Security Agent 的存取工作階段

支援轉寄存取工作階段 (FAS)：是

當您使用 IAM 使用者或角色在 AWS 中執行動作時，您會被視為委託人。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 使用呼叫 AWS 服務的委託人許可，結合請求 AWS 服務向下游服務提出請求。只有在服務收到需要與其他 AWS 服務或資源互動才能完成的請求時，才會提出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱轉發存取工作階段。

AWS Security Agent 的服務角色

支援服務角色：否

服務角色是服務擔任的 IAM 角色，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的建立角色以將許可委派給 AWS 服務。

AWS Security Agent 的服務連結角色

支援服務連結角色：是

服務連結角色是連結至 AWS 服務的一種服務角色。服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的 AWS 帳戶中，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

AWS Security Agent 身分型政策範例

根據預設， IAM 使用者 和 角色沒有建立或修改 AWS Security Agent 資源的許可。他們也無法使用 AWS Management Console AWS CLI或 AWS API 執行任務。 IAM 管理員必須建立 IAM 政策，授予使用者和角色對所需指定資源執行特定 API 操作的許可。然後，管理員必須將這些政策連接到需要這些許可的 IAM 使用者 或 群組。

若要了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的使用 JSON 編輯器建立政策。

政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 AWS Security Agent 資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管政策並邁向最低權限許可 – 若要開始將許可授予您的使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 中使用 AWS 帳戶。我們建議您定義特定於使用案例 AWS 的客戶受管政策，以進一步減少許可。如需詳細資訊，請參閱《IAM 使用者指南》中的 AWS 受管政策或 任務函數的 AWS 受管政策。
- 套用最低權限許可 – 當您使用 IAM 政策設定許可時，請僅授予執行任務所需的許可。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需使用 IAM 套用許可的詳細資訊，請參閱《IAM 使用者指南》中的 中的政策和許可 IAM。
- 使用 IAM 政策中的條件來進一步限制存取 – 您可以在政策中新增條件，以限制對動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。如果透過特定 例如 使用服務動作 AWS 服務，您也可以使用 條件來授予其存取權 CloudFormation。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素：條件。

- 使用 IAM Access Analyzer 驗證您的 IAM 政策以確保安全和功能許可 – IAM Access Analyzer 驗證新的和現有的政策，以便政策遵守 IAM 政策語言 (JSON) 和 IAM 最佳實務。 IAM Access Analyzer 提供超過 100 個政策檢查和可行的建議，以協助您撰寫安全和功能政策。如需詳細資訊，請參閱《IAM 使用者指南》中的IAM Access Analyzer 政策驗證。
- 需要多重要素驗證 (MFA) – 如果您的案例需要帳戶中的 IAM 使用者 或根使用者，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需更多資訊，請參閱 IAM 使用者指南中的設定 MFA 保護的 API 存取。

使用 AWS Security Agent 主控台

若要存取 AWS Security Agent 主控台，IAM 主體必須擁有一組最低許可。這些許可必須允許委託人列出和檢視您 中 AWS Security Agent 資源的詳細資訊 AWS 帳戶。如果您建立比最低必要許可更嚴格的身分型政策，則對於具有該政策的主體而言，主控台將無法如預期運作。

為了確保您的 IAM 主體仍然可以使用 AWS Security Agent 主控台，請使用您自己的唯一名稱建立政策。將政策連接至主體。如需詳細資訊，請參閱《IAM 使用者指南》中的新增許可到使用者。

如需政策詳細資訊，請參閱 the section called “AWS 受管政策：SecurityAgentWebAppAPIPolicy”。

對於僅呼叫 AWS CLI 或 AWS API 的使用者，您不需要允許最低主控台許可。反之，只需允許存取符合您嘗試執行之 API 作業的動作就可以了。

故障診斷 AWS Security Agent 身分和存取

使用以下資訊來協助您診斷和修正使用 AWS Security Agent 和 時可能遇到的常見問題 IAM。

AccessDeniedException

如果您在呼叫 AWS API 操作`AccessDeniedException`時收到 ，則您使用的 IAM 主體憑證沒有發出該呼叫所需的許可。

```
An error occurred (AccessDeniedException) when calling the CreateEnvironment operation:
User: arn:aws:iam::111122223333:user/user_name is not authorized to perform:
securityagent:CreateEnvironment on resource:
 arn:aws:securityagent:region:111122223333:environment/my-env
```

在先前的範例訊息中，使用者沒有呼叫 AWS Security Agent `CreateEnvironment` API 操作的許可。若要提供 AWS Security Agent 管理員許可給 IAM 主體，請參閱 the section called “AWS Security Agent 身分型政策範例”。

如需 IAM 的一般資訊，請參閱《IAM 使用者指南》中的使用政策控制對 AWS 資源的存取。

我想要允許 以外的人員 AWS 帳戶 存取我的 AWS Security Agent 資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解 AWS Security Agent 是否支援這些功能，請參閱 the section called “AWS Security Agent 如何使用 IAM”。
- 若要了解如何提供您擁有之 資源 AWS 帳戶 的存取權，請參閱《IAM 使用者指南》 IAM 使用者 中的在您擁有 AWS 帳戶 的另一個 中提供 的存取權。
- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱《IAM 使用者指南》中的將存取權提供給第三方 AWS 帳戶 擁有。
- 若要了解如何透過聯合身分提供存取權，請參閱《IAM 使用者指南》中的提供存取權給外部驗證的使用者（聯合身分）。
- 若要了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 角色與資源型政策的差異。

事件回應

AWS Security Agent 是一項主動式安全測試服務，旨在識別並防止漏洞遭到利用。此服務著重於透過設計審查、程式碼分析和滲透測試進行預防性安全驗證，而不是被動事件偵測或回應。

事件偵測

AWS Security Agent 不提供事件偵測功能。此服務可做為主動式安全測試工具運作，在開發期間和部署之前驗證應用程式是否有漏洞。對於執行時間安全監控和事件偵測，請使用 Amazon GuardDuty、AWS Security Hub 或 Amazon CloudWatch 等服務。

事件提醒

AWS Security Agent 不會為安全事件產生即時提醒。在完成設計檢閱、程式碼分析或滲透測試業務之後，此服務會透過 AWS 主控台交付安全性問題清單。這些調查結果代表測試期間發現的潛在漏洞，而不是作用中的安全事件。

事件修復

AWS Security Agent 不提供自動事件修復。服務可識別安全漏洞並提供修補指引，包括：

- 已識別漏洞的詳細說明
- 已驗證問題清單的可重現入侵路徑
- 特定程式碼修正和實作指引
- 已發現問題的影響分析

開發和安全團隊使用此指南，在漏洞到達生產環境之前手動解決漏洞。

支援事件回應活動

雖然 AWS Security Agent 並非針對事件回應而設計，但安全團隊可以使用 服務來支援事件後活動：

漏洞驗證

發生安全事件後，請使用 AWS Security Agent 來測試其他應用程式或環境中是否存在類似的漏洞。

安全狀態評估

執行滲透測試，以驗證在事件修復過程中實作的安全改進。

根本原因分析

使用程式碼安全檢閱功能來識別其他程式碼庫中可能存在的類似漏洞。

AWS Security Agent 的合規驗證

若要了解 AWS 服務是否在特定合規計劃的範圍內，請參閱合規計劃範圍內的 AWS 服務，然後選擇您感興趣的合規計劃。如需一般資訊，請參閱 AWS 合規計劃。

您可以使用 AWS Artifact 下載第三方稽核報告。如需詳細資訊，請參閱在 AWS Artifact 中下載報告。

您在使用 AWS 服務時的合規責任取決於資料的機密性、您公司的合規目標，以及適用的法律和法規。AWS 提供下列資源，以協助合規：

- 安全合規與治理 - 這些解決方案實作指南內容討論了架構考量，並提供部署安全與合規功能的步驟。
- HIPAA 合格服務參考 – 列出 HIPAA 合格服務。並非所有 AWS 服務都符合 HIPAA 資格。

- AWS 合規資源 - 此系列工作手冊和指南可能適用於您的產業和地點。
- AWS 客戶合規指南 – 透過合規的角度了解共同責任模型。本指南摘要說明保護 AWS 服務的最佳實務，並將指引映射至跨多個架構的安全控制（包括國家標準技術研究所 (NIST)、支付卡產業安全標準委員會 (PCI) 和國際標準化組織 (ISO)）。
- 《AWS Config 開發人員指南》中的使用規則評估資源 – AWS Config 服務會評估資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub – 此 AWS 服務提供 AWS 內安全狀態的完整檢視。Security Hub 使用安全控制來評估您的 AWS 資源，並檢查是否符合安全產業標準和最佳實務。如需支援的服務和控制清單，請參閱 Security Hub 控制參考。
- Amazon GuardDuty – 此 AWS 服務透過監控您的環境是否有可疑和惡意活動，偵測對 AWS 帳戶、工作負載、容器和資料的潛在威脅。GuardDuty 可滿足特定合規架構所規定的入侵偵測需求，以協助您因應 PCI DSS 等各種不同的合規需求。
- AWS Audit Manager – 此 AWS 服務可協助您持續稽核 AWS 使用情況，以簡化管理風險與法規與業界標準的法規遵循方式。

AWS Security Agent 中的彈性

> **Note**
>
> AWS 安全代理程式適用於下列區域：美國東部（維吉尼亞北部）– us-east-1、美國西部（奧勒岡）– us-west-2、亞太區域（孟買）– ap-south-1、亞太區域（新加坡）– ap-southeast-1、亞太區域（雪梨）– ap-southeast-2、亞太區域（東京）– ap-northeast-1、歐洲（法蘭克福）– eu-central-1、歐洲（愛爾蘭）– eu-west-1和南美洲（聖保羅）– sa-east-1。它使用跨區域推論。在亞太區域（孟買）、亞太區域（新加坡）和南美洲（聖保羅），AWS 安全代理程式會使用全球跨區域推論，將推論請求路由到任何商業 AWS 區域。在所有其他區域中，它使用地理跨區域推論，將資料處理保持在特定的地理邊界內。AWS Security Agent 是開發應用程式時使用的工具，不應部署為關鍵或面向客戶的基礎設施。

AWS 全球基礎設施以 AWS 區域與可用區域為中心建置。AWS 區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援聯網相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域和可用區域的詳細資訊，請參閱 AWS 全球基礎設施。

AWS Security Agent 中的基礎設施安全

作為受管服務，AWS Security Agent 受到 AWS 全球網路安全的保護。如需有關 AWS 安全服務和 AWS 如何保護基礎設施的資訊，請參閱 AWS Cloud Security。若要使用基礎設施安全性的最佳實務來設計您的 AWS 環境，請參閱 AWS Well-Architected Framework 中的安全性。

網路隔離

AWS Security Agent 是一種全受管服務，可透過 AWS 主控台和 AWS Security Agent Web 應用程式存取。對服務的存取是透過 AWS Identity and Access Management (IAM) 或 AWS IAM Identity Center 控制，可與您的身分提供者整合。

AWS Security Agent 支援 AWS PrivateLink 支援的介面 VPC 端點，可讓您從 VPC 私下存取服務 APIs，而無需周遊公有網際網路。如需詳細資訊，請參閱the section called “介面 VPC 端點”。

AWS Security Agent 需要網際網路存取權，才能對目標應用程式和控制平面操作執行滲透測試。此服務使用 AWS 受管基礎設施進行測試，不會在您帳戶中佈建 EC2 執行個體或其他運算資源。如果您設定 VPC 存取進行滲透測試，安全代理程式會在子網路中建立 ENI，但此 ENI 沒有公有 IP 地址。如需詳細資訊，請參閱the section called “將代理程式連線至私有 VPC 資源”。

多租戶和資源隔離

AWS Security Agent 是一項多租戶服務。安全審查、調查結果和客戶資料會隔離到個別 AWS 帳戶，並靜態加密。AWS 會套用標準基礎設施隔離控制，以確保某個客戶的安全測試活動不會影響另一個客戶的效能或機密性。

AWS 安全代理程式和界面 VPC 端點 (AWS PrivateLink)

您可以使用 AWS PrivateLink 在 VPC 和 AWS 安全代理程式之間建立私有連線。您可以像在 VPC 中一樣存取安全代理程式，無需使用網際網路閘道、NAT 裝置、VPN 連接或 Direct Connect 連接。VPC 中的執行個體不需要公有 IP 地址即可存取 安全代理程式。

您可以透過建立由 AWS PrivateLink 提供支援的介面端點來建立此私有連線。我們會在您為介面端點啟用的每個子網中建立端點網路介面。這些是申請者管理的網路介面，可做為目的地為 Security Agent 之流量的進入點。

如需詳細資訊，請參閱《 AWS PrivateLink 指南》中的透過 PrivateLink 存取 AWS 服務。 AWS PrivateLink

安全代理程式的考量事項

在您設定 Security Agent 的介面端點之前，請檢閱 AWS PrivateLink 指南中的考量事項。

Security Agent 支援從您的 VPC 呼叫其所有 API 動作，包括管理代理程式空間、滲透測試和安全調查結果的操作。

您可以在建立端點時選取 IPv4、IPv6 或雙堆疊。

Security Agent 支援 VPC 端點政策。根據預設，允許透過介面端點完整存取 Security Agent。您可以將端點政策連接至介面端點，或建立安全群組與端點網路介面的關聯，以控制存取。

所有對 Security Agent 的 API 呼叫都會使用 TLS 1.2 或更高版本加密，包括透過 VPC 端點進行的呼叫。如需資料保護的詳細資訊，請參閱 the section called “資料保護”。Security Agent API 呼叫會記錄在 AWS CloudTrail 中。如需詳細資訊，請參閱the section called “範例： AWS 安全代理程式日誌檔案項目”。

建立 Security Agent 的介面端點

您可以使用 Amazon VPC 主控台或 AWS 命令列界面 (AWS CLI) 來建立 Security Agent 的界面端點。如需詳細資訊，請參閱 AWS PrivateLink 指南中的建立介面端點。

使用下列服務名稱建立 Security Agent 的介面端點：

```
com.amazonaws.<region>.securityagent
```

若要使用 CLI AWS 建立介面端點，請執行下列命令。將區域、VPC ID、子網路 IDs和安全群組 ID 取代為您自己的值。

```
aws ec2 create-vpc-endpoint \
    --vpc-id vpc-1a2b3c4d \
    --vpc-endpoint-type Interface \
    --service-name com.amazonaws.<region>.securityagent \
    --subnet-ids subnet-1a2b3c4d subnet-5e6f7a8b \
    --security-group-ids sg-1a2b3c4d \
    --private-dns-enabled
```

> **Important**
>
> 必須為介面端點啟用私有 DNS。安全代理程式要求您使用預設的區域 DNS 名稱（例如，`securityagent.us-east-1.api.aws`) 透過端點提出 API 請求。不支援直接呼叫端點特定的 VPCE URL。

若要使用私有 DNS，您必須將 VPC `true`的 `enableDnsSupport`和 `enableDnsHostnames` 屬性同時設定為 。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 VPC 的 DNS 屬性。

設定介面端點的安全群組

建立介面端點時，您可以將安全群組與端點網路介面建立關聯，以控制連至 Security Agent 的流量。建立安全群組，允許來自 VPC 中需要與安全代理程式通訊之資源的傳入 HTTPS 流量（連接埠 443)。

安全群組應包含下列傳入規則：

- 類型：HTTPS
- Protocol (通訊協定)：TCP
- 連接埠範圍：443
- 來源：您的 VPC CIDR 區塊（例如 `10.0.0.0/16`) 或需要存取 Security Agent 的資源的安全群組 IDs

如需詳細資訊，請參閱 AWS PrivateLink 指南中的安全群組。

為您的介面端點建立端點政策

端點政策為 IAM 資源，您可將其連接至介面端點。預設端點政策允許透過介面端點完整存取 Security Agent。若要控制允許從 VPC 存取安全代理程式，請將自訂端點政策連接至介面端點。

端點政策會指定以下資訊：

- 可執行動作的主體 (AWS 帳戶、IAM 使用者和 IAM 角色）。
- 可執行的動作。
- 可供執行動作的資源。

如需詳細資訊，請參閱《 AWS PrivateLink 指南》中的使用端點政策控制對 服務的存取。

範例： AWS Security Agent 動作的 VPC 端點政策

以下是 AWS Security Agent 的端點政策範例。連接至 端點時，此政策會授予所有資源上所有委託人的所列 AWS Security Agent 動作的存取權。

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": [
            "securityagent:CreatePentest",
            "securityagent:ListPentests",
            "securityagent:BatchGetPentests",
            "securityagent:StartPentestExecution",
            "securityagent:StopPentestExecution",
            "securityagent:ListFindings",
            "securityagent:BatchGetFindings"
         ],
         "Resource": "*"
      }
   ]
}
```

範例：拒絕來自指定 AWS 帳戶的所有存取的 VPC 端點政策

下列 VPC 端點政策拒絕 AWS 使用端點帳戶對資源123456789012的所有存取。此政策允許來自其他帳戶的所有動作。

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "securityagent:*",
         "Resource": "*"
      },
      {
         "Effect": "Deny",
         "Principal": {
            "AWS": "123456789012"
         },
         "Action": "securityagent:*",
         "Resource": "*"
      }
```

```
    ]
}
```

疑難排解

呼叫 Security Agent API 時連線逾時

- 確認 VPC 端點狀態為 `available`：

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
    --query "VpcEndpoints[].State"
```

- 確認與端點相關聯的安全群組允許來自 VPC CIDR 或來源安全群組的連接埠 443 上的傳入 TCP 流量。
- 檢查您的 VPC 路由表不會將 Security Agent 流量路由到網際網路閘道或 NAT 閘道，而不是端點。

的 DNS 解析失敗 **`securityagent.<region>.api.aws`**

- 確認已在端點上啟用私有 DNS：

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
    --query "VpcEndpoints[].PrivateDnsEnabled"
```

- 確認您的 VPC `true` 上的 `enableDnsSupport`和 `enableDnsHostnames` 都設定為 ：

```
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsSupport
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsHostnames
```

`Access denied` 呼叫 Security Agent APIs時發生錯誤

- 檢查 VPC 端點政策，確保它允許您嘗試執行的動作。
- 確認您的 IAM 身分政策授予必要的`securityagent:`許可。
- 如果使用自訂端點政策，請確認端點政策和 IAM 政策都允許請求。

AWS Security Agent 中的組態和漏洞分析

組態和 IT 控制是 AWS 與身為我們客戶的您共同的責任。如需詳細資訊，請參閱 AWS 共同責任模型。

預防跨服務混淆代理人

混淆代理人問題屬於安全性議題，其中沒有執行動作許可的實體可以強制具有更多許可的實體執行該動作。在 AWS 中，跨服務模擬可能會導致混淆代理人問題。在某個服務 (呼叫服務) 呼叫另一個服務 (被呼叫服務) 時，可能會發生跨服務模擬。可以操縱呼叫服務來使用其許可，以其不應有存取許可的方式對其他客戶的資源採取動作。為了防止這種情況，AWS 提供的工具可協助您保護所有 服務的資料，其服務主體已獲得您帳戶中資源的存取權。如需詳細資訊，請參閱《AWS Identity and Access Management 使用者指南》中的預防跨服務混淆代理人。

AWS Security Agent 的安全最佳實務

AWS Security Agent 提供許多安全功能，供您在開發和實作自己的安全政策時加以考量。以下最佳實務為一般準則，並不代表完整的安全解決方案。這些最佳實務可能不適用或無法滿足您的環境需求，因此請將其視為實用建議就好，而不要當作是指示。

使用非生產環境進行滲透測試

AWS Security Agent 使用來自 Kali Linux 發行版本的完整滲透測試工具套件。這些工具旨在識別安全漏洞，並可能執行修改應用程式狀態、資料或系統組態的動作。

最佳實務：針對反映生產設定的非生產環境執行滲透測試。這些測試環境應該：

- 不包含即時客戶資料或敏感的生產資訊
- 與生產系統隔離
- 具有與生產類似的組態和安全控制
- 不使用登入資料來存取生產系統

在生產環境中進行測試可能會導致：

- 資料修改或刪除
- 服務中斷或效能降低
- 意外狀態變更
- 觸發安全提醒或事件回應程序

驗證 AI 產生的安全調查結果

AWS Security Agent 會使用 AI 代理器執行安全分析。由於 AI 系統具有非確定性的性質，滲透測試執行可能會在不同的執行中產生不同的結果。

最佳實務：在採取修復動作之前驗證安全性問題清單：

- 檢閱 AWS Security Agent 為每個問題清單產生的驗證指令碼
- 在測試環境中執行驗證指令碼以確認漏洞
- 考慮執行多個滲透測試，以確保全面涵蓋
- 套用專業安全判斷來評估問題清單嚴重性和可利用性

並非所有已識別的問題都代表在特定部署內容中可利用的漏洞。

檢閱和測試產生的修補程式碼

AWS Security Agent 可以為已識別的漏洞產生程式碼修正和安全性改進。這些 AI 產生的修正需要在部署之前進行驗證。

最佳實務：檢閱所有產生的程式碼變更：

- 檢查提議的修正是否完整和正確
- 在非生產環境中徹底測試修正
- 確認修正不會引入新的漏洞或中斷功能
- 套用修正後使用 AWS Security Agent 重新測試，或執行提供的驗證指令碼
- 遵循組織的程式碼檢閱和核准程序

程式碼儲存庫存取

AWS Security Agent 可以透過提取請求評論和程式碼檢閱整合，提供程式碼變更的安全指引。為了保護敏感安全資訊，此功能在特定限制下運作。

限制：程式碼安全指引僅限於私有儲存庫。這可確保：

- 安全性調查結果會對您的組織保密
- 修補之前不會公開揭露潛在漏洞
- 產生的修正建議不會公開入侵詳細資訊

AWS Security Agent 不會為公有儲存庫提供程式碼安全指導。它不會對公開可見安全性問題清單的公有儲存庫或開放原始碼專案進行評論。

AWS Security Agent 滲透測試可以檢閱和修復您為 pentest 設定的私有和公有儲存庫。如果儲存庫是公有的，修補程式碼將以可下載的 diff 檔案提供，而不是提取請求。

可存取URLs

可存取URLs 指定滲透測試環境在測試期間可存取的其他端點。當您的應用程式依賴第三方身分驗證提供者或 CDNs 等外部服務時，這些是必要的。測試所需的所有網路相依性都必須指定為目標 URLs或可存取URLs。網路會封鎖對任何未指定端點的存取。

安全性影響：未指示 AWS Security Agent 對可存取URLs 執行安全性測試。透過指定可存取URLs，您可以指出對這些相依性的信任。滲透測試資料，包括登入資料，可能會在測試期間傳輸到這些可存取的 URL 端點。

跨區域推論

AWS Security Agent 會自動選取最佳區域來處理推論請求。這可將可用的運算資源、模型可用性最大化，並提供最佳客戶體驗。您的資料只會存放在發出請求的區域中；不過，輸入提示和輸出結果可能會在該區域之外處理。我們會跨 AWS 網路傳輸所有加密的資料。

AWS Security Agent 會根據區域使用兩種跨區域推論類型：

- 地理跨區域推論 – 將資料處理保持在大多數功能的特定地理邊界（例如美國、歐洲、澳洲或日本）內。對於程式碼修復，來自澳洲和日本的請求會在歐盟處理。用於美國東部（維吉尼亞北部）– us-east-1、美國西部（奧勒岡）– us-west-2、亞太區域（雪梨）– ap-southeast-2、亞太區域（東京）– ap-northeast-1、歐洲（法蘭克福）– eu-central-1和歐洲（愛爾蘭）– eu-west-1。
- 全球跨區域推論 – 將推論請求路由到任何商業 AWS 區域，最佳化可用資源並實現更高的模型輸送量。用於亞太區域（孟買）– ap-south-1、亞太區域（新加坡）– ap-southeast-1和南美洲（聖保羅）– sa-east-1。

對於使用全域跨區域推論的區域，輸入提示和輸出結果可能會在任何商業 AWS 區域中處理。跨區域操作期間傳輸的所有資料都會保留在 AWS 網路上，而不會周遊公有網際網路。我們會加密 AWS 區域之間傳輸中的資料。

下表說明根據發出請求的區域和使用的功能來處理推論請求的位置。

請求原始伺服器	程式碼修補以外的所有功能	程式碼修復
美國 — 美國東部（維吉尼亞北部）– us-east-1 、美國西部（奧勒岡）– us-west-2	美國	美國
歐盟 – 歐洲（愛爾蘭）– eu-west-1 、歐洲（法蘭克福）– eu-central-1	歐盟	歐盟
澳洲 — 亞太區域（雪梨）— ap-southeast-2	澳洲	歐盟
日本 — 亞太區域（東京）— ap-northeast-1	日本	歐盟
南美洲 – 南美洲（聖保羅）– sa-east-1	任何商業 AWS 區域	任何商業 AWS 區域
印度 — 亞太區域（孟買）— ap-south-1	任何商業 AWS 區域	任何商業 AWS 區域
東南亞 — 亞太區域（新加坡）— ap-southeast-1	任何商業 AWS 區域	任何商業 AWS 區域

跨區域推論一律啟用，且無法選擇退出。跨區域推論不受服務控制政策 (SCPs) 或 AWS Control Tower 中的客戶政策影響，這些政策會將客戶內容限制在特定區域。如需 AWS Security Agent 如何在跨區域處理期間保護資料的詳細資訊，請參閱跨區域資料處理。

IAM 動作和資源遷移

AWS Security Agent 是一種前端代理程式，可在所有環境的開發生命週期中主動保護您的應用程式。如果您在 2026 年 2 月 9 日之前加入 AWS 安全代理程式，您將受到 2026 年 3 月 9 日現有代理程式執行個體資源和 AWS 安全代理程式 IAM 動作的近期變更的影響。為了準備發佈公有 API/SDK 支援，正在將 代理程式執行個體資源重新命名為 代理程式空間，並且正在重新命名特定 IAM 動作。這些變更將影響您在 2026 年 3 月 9 日之前建立的任何應用程式或代理程式執行個體 IAM 角色。為了避免在 2026 年 3 月 9 日之後看到身分驗證問題，您將需要遵循準備遷移中的步驟。

> **Note**
>
> 如果您在 2026 年 2 月 9 日之後建立任何新的 代理程式執行個體，新的 代理程式執行個體將建立為 代理程式空間，而且不需要任何遷移步驟。

計劃變更

AWS Security Agent 正在將代理程式執行個體資源重新命名為 Agent Space：`arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*`重新命名為 `arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*`。此外，以下 IAM 動作正在重新命名：

- `securityagent:ListAgentInstances` 重新命名為 `securityagent:ListAgentSpaces`
- `securityagent:ListControls` 重新命名為 `securityagent:ListSecurityRequirements`
- `securityagent:BatchGetAgentInstances` 重新命名為 `securityagent:BatchGetAgentSpaces`
- `securityagent:BatchGetSecurityTestContentMetadata` 重新命名為 `securityagent:BatchGetPentestJobContentMetadata`
- `securityagent:BatchGetTasks` 重新命名為 `securityagent:BatchGetPentestJobTasks`
- `securityagent:CreateDocumentReview` 重新命名為 `securityagent:CreateDesignReview`
- `securityagent:GetDocumentReview` 重新命名為 `securityagent:GetDesignReview`
- `securityagent:GetDocumentReviewArtifact` 重新命名為 `securityagent:GetDesignReviewArtifact`
- `securityagent:ListDocumentReviews` 重新命名為 `securityagent:ListDesignReviews`
- `securityagent:ListDocumentReviewComments` 重新命名為 `securityagent:ListDesignReviewComments`
- `securityagent:ListTasks` 重新命名為 `securityagent:ListPentestJobTasks`
- `securityagent:StartPentestExecution` 重新命名為 `securityagent:StartPentestJob`
- `securityagent:StopPentestExecution` 重新命名為 `securityagent:StopPentestJob`
- `securityagent:DeleteDocumentReview` 重新命名為 `securityagent:DeleteDesignReview`

準備遷移

為了避免在 2026 年 3 月 9 日之前繼續使用 AWS Security Agent 時，在 2026 年 3 月 9 日之前發現問題，您將需要信任 IAM 角色/政策中的新舊資源和 IAM 動作，直到 2026 年 3 月 9 日為止。以下說明將提供遷移至新資源和動作格式的指南：

1. 登入您的 AWS 帳戶並導覽至 AWS Security Agent 主控台
2. 在左側面板中，選取設定，然後在服務角色下選擇角色
3. 在關聯角色的 IAM 主控台中，選取新增許可和連接政策
4. 選取 AWSSecurityAgentWebAppPolicy，然後選擇新增許可
   a. 重要注意事項：確認您已選取 AWSSecurityAgentWebAppPolicy 做為新政策，而非 SecurityAgentWebAppAPIPolicy
5. 在許可政策下，確認您的 IAM 角色現在同時具有 AWSSecurityAgentWebAppPolicy 和 SecurityAgentWebAppAPIPolicy
6. 在相同的 IAM 角色主控台中，選取信任關係，然後選取編輯信任政策
7. 將您的信任政策更新為下列格式，以您的 AWS 帳戶 ID 取代 {{accountId}}

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    },
```

```
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:SetContext",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ForAllValues:ArnEquals": {
          "sts:RequestContextProviders": "arn:aws:iam::aws:contextProvider/
IdentityCenter"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    }
  ]
}
```

8. 導覽回 AWS Security Agent 主控台。從左側面板中，選取客服人員空間
9. 針對啟用滲透測試的每個客服人員空間，執行下列步驟
   a. 導覽至客服人員空間，然後選取滲透測試
   b. 在服務存取下，選擇服務角色名稱下的角色
   c. 在關聯角色的 IAM 主控台中，選取信任關係，然後選取編輯信任政策
   d. 將您的信任政策更新為下列格式，以您的 AWS 帳戶 ID 取代 {{accountId}}

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
```

```
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    }
  ]
}
```

使用 AWS CloudTrail 記錄 AWS 安全代理程式 API 呼叫

AWS Security Agent 已與 AWS CloudTrail 整合，CloudTrail 是一種服務，可提供使用者、角色或 AWS Security Agent 中 AWS 服務所採取動作的記錄。CloudTrail 會將 AWS Security Agent 的所有 API 呼叫擷取為事件。擷取的呼叫包括來自 AWS Security Agent 主控台的呼叫，以及對 AWS Security Agent API 操作的程式碼呼叫。如果您建立線索，則可以將 CloudTrail 事件持續交付至 Amazon S3 儲存貯體，包括 AWS Security Agent 的事件。即使您未設定追蹤，依然可以透過 CloudTrail 主控台中的事件歷史記錄檢視最新事件。您可以使用 CloudTrail 所收集的資訊，判斷向 AWS Security Agent 提出的請求、提出請求的 IP 地址、提出請求的人員、提出請求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱《AWS CloudTrail 使用者指南》。

AWS CloudTrail 中的安全代理程式資訊

當您建立 AWS 帳戶時，會在您的帳戶上啟用 CloudTrail。在 AWS 安全代理程式中發生活動時，該活動會與事件歷史記錄中的其他 AWS 服務事件一起記錄在 CloudTrail 事件中。您可以在 AWS 帳戶中檢視、搜尋和下載最近的事件。如需詳細資訊，請參閱「使用 CloudTrail 事件歷史記錄檢視事件」。

若要持續記錄您 AWS 帳戶中的事件，包括 AWS 安全代理程式的事件，請建立追蹤。線索能讓 CloudTrail 將日誌檔案交付至 Amazon S3 儲存貯體。根據預設，當您在主控台中建立線索時，線索會套用至所有 AWS 區域。線索會記錄 AWS 分割區中所有區域的事件，並將日誌檔案傳送到您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中收集的事件資料。如需詳細資訊，請參閱下列內容：

- 建立追蹤的概觀
- CloudTrail 支援的服務和整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 接收多個區域的 CloudTrail 日誌檔案和接收多個帳戶的 CloudTrail 日誌檔案

CloudTrail 會記錄所有 AWS Security Agent 動作。例如，對 `CreatePentest`、`BatchGetPentestJobs` 以及 `ListFindings` 動作發出的呼叫會在 CloudTrail 日誌檔案中產生項目。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 是否使用根或 AWS Identity and Access Management (IAM) 使用者登入資料提出請求。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱 CloudTrail userIdentity 元素。

範例： AWS 安全代理程式日誌檔案項目

了解 AWS Security Agent 日誌檔案項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付到您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌專案。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。CloudTrail 日誌檔並非依公有 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

以下範例顯示的是展示 `CreatePentest` 動作的 CloudTrail 日誌項目：

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Alice"
    },
```

```
    "eventTime": "2025-01-15T10:30:00Z",
    "eventSource": "securityagent.amazonaws.com",
    "eventName": "CreatePentest",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "aws-cli/2.13.0",
    "requestParameters": {
        "pentestName": "WebApp-Security-Test",
        "targetUrl": "https://example.com",
        "testScope": "OWASP-Top-10"
    },
    "responseElements": {
        "pentestId": "pt-1234567890abcdef0",
        "pentestArn": "arn:aws:securityagent:us-east-1:123456789012:pentest/
pt-1234567890abcdef0"
    },
    "requestID": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
    "eventID": "12345678-1234-1234-1234-123456789012",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Service Quotas

AWS Security Agent 具有配額，可限制您可以建立的資源數量，以及您可以執行操作的速率。透過 AWS Support 提交請求，即可增加標記為可調整的配額。如需詳細資訊，請參閱「建立支援案例和案例管理」。

操作配額

操作配額會限制安全性測試的每月用量，並檢閱功能以協助管理服務容量。

資源	Scope (範圍)	配額	可調整
設計檢閱	每個區域每個帳戶每月	200	是
PR 程式碼檢閱	每個區域每個帳戶每月	1,000	是

組態配額

組態配額會限制您可以在 AWS Security Agent 環境中設定的資源和設定數目。

資源	Scope (範圍)	配額	可調整
客服人員空間	每個區域每個帳戶	100	是
整合	每個區域每個帳戶	20	否
每個整合的整合資源	每次整合	50	否
自訂安全需求	每個區域每個帳戶	20	否
Pentest 專案	每個區域每個帳戶	1,000	是
並行 pentest 執行	每個區域每個帳戶	5	是
程式碼檢閱專案	每個區域每個帳戶	1,000	是
並行程式碼檢閱執行	每個區域每個帳戶	5	是

文件歷史紀錄

下表說明 AWS Security Agents 使用者指南的一些主要更新和新功能。

變更	描述	日期
私有連線（預覽）	新增私有連線的文件。這項新功能可讓 AWS Security Agent 使用 Amazon VPC Lattice 連線到私有網路中執行的來源控制系統，而無需將您的系統暴露在公有網際網路。	2026 年 6 月 16 日
威脅建模（預覽）	新增了威脅建模的文件。這項新功能會從設計文件（範圍文件）、原始程式碼（來源）或兩者，為您的應用程式建置威	2026 年 6 月 8 日

	脅模型。範圍文件是定義分析重點的特徵設計文件，而原始程式碼提供現有系統的內容。如果您不提供範圍文件，代理程式只會從原始碼產生威脅模型。每次執行都會產生系統概觀，以及一組依 STRIDE 類別分類且具有嚴重性評分和建議的威脅。	
完整儲存庫程式碼檢閱（預覽版）	新增完整儲存庫程式碼檢閱的文件。這項新功能會執行整個程式碼庫的內容感知安全分析，並為問題清單產生程式碼修復。	2026 年 5 月 12 日
更新區域可用性以尋找修補	已更新 AWS Security Agent Web 應用程式中滲透測試問題清單修復的區域可用性。	2026 年 4 月 16 日
更新了啟用問題清單修復的區域可用性	已更新在 AWS 管理主控台中啟用滲透測試問題清單修復的區域可用性。	2026 年 4 月 16 日
客戶自管金鑰支援	新增客戶受管金鑰 (CMK) 支援的文件。您現在可以在建立客服人員空間和整合以使用您控制的金鑰加密資料時，指定客戶受管 KMS 金鑰。	2026 年 3 月 31 日
AWS 受管政策更新	針對新的 TargetDomain 和 DesignReviewFeedback 資源類型新增 AWSSecurityAgentWebAppPolicy 受管政策。	2026 年 3 月 31 日

AWS 受管政策更新	針對新的 AgentSpace 資源類型和 IAM 動作名稱變更新增 AWSSecurityAgentWebAppPolicy 受管政策。	2026 年 2 月 9 日
AWS 受管政策更新	更新 SecurityAgentWebAppAPIPolicy 以允許客戶刪除設計檢閱。	2026 年 1 月 28 日
AWS 受管政策更新	更新 SecurityAgentWebAppAPIPolicy，讓客戶能夠開始針對安全性問題清單進行自動程式碼修復。	2026 年 1 月 20 日
AWS 受管政策更新	更新至 SecurityAgentWebAppAPIPolicy，以允許客戶在主控台中檢視映像。	2025 年 12 月 5 日
AWS Security Agents 初始版本	服務啟動的初始文件	2025 年 12 月 2 日

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。