



Application Load Balancer

Elastic Load Balancing



Elastic Load Balancing: Application Load Balancer

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

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

Table of Contents

什麼是 Application Load Balancer ?	1
Application Load Balancer 元件	1
Application Load Balancer 概觀	2
從 Classic Load Balancer 遷移的優點	2
相關服務	3
定價	4
開始使用	5
開始之前	5
步驟 1：設定您的目標群組	5
步驟 2：選擇負載平衡器類型	6
步驟 3：設定您的負載平衡器和接聽程式	6
步驟 4：測試您的負載平衡器	7
步驟 5：(選用) 刪除負載平衡器	8
教學課程：使用 AWS CLI 建立 Application Load Balancer	9
開始之前	9
建立負載平衡器	9
新增 HTTPS 接聽程式	11
新增以路徑為基礎的路由	11
刪除負載平衡器	12
負載平衡器	13
負載平衡器的子網路	14
可用區域子網路	14
Local Zone 子網路	15
Outpost 子網路	15
負載平衡器安全群組	16
負載平衡器狀態	16
負載平衡器屬性	17
IP 地址類型	19
負載平衡器資源對應	20
資源對映元件	20
負載平衡器連線	22
連線閒置逾時	22
HTTP 用戶端保持活動時間	22
跨區域負載平衡	24

刪除保護	24
去同步緩解模式	25
主機標頭保留	26
AWS WAF	28
建立負載平衡器	29
步驟 1：設定目標群組	5
步驟 2：註冊目標	31
步驟 3：設定負載平衡器和接聽程式	31
步驟 4：測試負載平衡器	7
更新可用區域	35
更新安全群組	36
建議的規則	36
更新相關聯的安全群組	38
更新地址類型	38
更新標籤	39
刪除負載平衡器	40
區域轉移	41
啟動區域轉移	42
更新區域轉移	43
取消區域轉移	43
接聽程式和規則	45
接聽程式組態	45
接聽程式規則	46
預設規則	46
規則優先順序	47
規則動作	47
規則條件	47
規則動作類型	47
固定回應動作	48
轉送動作	48
重新導向動作	50
規則條件類型	54
HTTP 標頭條件	55
HTTP 請求方法條件	55
主機條件	56
路徑條件	57

查詢字串條件	58
來源 IP 地址條件	59
建立 HTTP 接聽程式	59
必要條件	60
新增 HTTP 接聽程式	60
建立 HTTPS 接聽程式	61
SSL 憑證	61
安全政策	63
新增 HTTPS 接聽程式	86
更新接聽程式規則	88
需求	88
新增規則	88
編輯規則	90
重新排列規則	91
刪除規則	92
更新 HTTPS 接聽程式	93
更換預設憑證	93
將憑證新增至憑證清單	94
從憑證清單中移除憑證	94
更新安全政策	95
使用相互 TLS 驗證	96
開始之前	96
HTTP 標頭	99
設定相互 TLS	101
連線日誌	106
驗證使用者身分	106
準備使用 OIDC 合規 IdP	106
準備使用 Amazon Cognito	107
準備使用 Amazon CloudFront	108
設定使用者身分驗證	109
身分驗證流程	111
使用者宣告編碼和簽章驗證	113
逾時	116
身分驗證登出	117
X-Forwarded 標頭	118
X-Forwarded-For	118

X-Forwarded-Proto	121
X-Forwarded-Port	122
更新標籤	122
更新接聽程式標籤	122
更新規則標籤	123
刪除接聽程式	124
目標群組	125
路由組態	126
Target type (目標類型)	126
IP 地址類型	128
通訊協定版本	128
已登記的目標	129
目標群組屬性	130
路由算法	132
修改目標群組的路由演算法	133
自動目標權重 (ATW)	133
異常偵測	134
異常緩解	135
取消登記的延遲	136
慢速啟動模式	137
建立目標群組	138
設定運作狀態檢查	140
運作狀態檢查設定	140
目標運作狀態	142
運作狀態檢查原因代碼	143
檢查目標的運作狀態	144
修改目標群組的運作狀態檢查設定	145
跨區域負載平衡 (Cross-zone load balancing)	145
關閉跨區域負載平衡	146
開啟跨區域負載平衡	147
目標群組運作狀態	148
運作運作狀態不佳	148
需求和考量事項	149
監控	149
範例	149
修改目標群組運作狀態良好設定	150

針對您的負載平衡器使用 Route 53 DNS 備援	151
登記目標	152
目標安全群組	153
共用子網路	153
登記和取消登記目標	153
黏性工作階段	156
持續時間型粘性	157
應用程式型粘性	159
Lambda 函數作為目標	161
準備 Lambda 函數	162
為 Lambda 函數建立目標群組	155
從負載平衡器接收事件	164
對負載平衡器進行回應	165
多值標頭	165
啟用運作狀態檢查	168
取消註冊 Lambda 函數	169
更新標籤	170
刪除目標群組	171
監控負載平衡器	172
CloudWatch 度量	172
Application Load Balancer 指標	173
Application Load Balancer 的指標維度	190
Application Load Balancer 指標的統計資料	191
CloudWatch 檢視負載平衡器的指標	192
存取日誌	194
存取日誌檔	194
存取日誌項目	196
範例日誌項目	208
處理存取日誌檔	210
啟用存取日誌	211
停用存取日誌	218
連線日誌	218
連線記錄檔	219
連線日誌項目	220
範例日誌項目	223
處理連線記錄檔	224

啟用連線記錄	224
停用連線記錄	230
請求追蹤	231
語法	231
限制	232
CloudTrail 日誌	232
Elastic Load Balancing 資訊 CloudTrail	232
了解 Elastic Load Balancing 日誌檔案項目	233
為您的負載平衡器進行疑難排解	236
已註冊目標處於非服務中狀態	236
用戶端無法連接到面向網際網路的負載平衡器	237
負載平衡器不會收到傳送至自訂域的請求	238
傳送至負載平衡器的 HTTPS 要求會傳回 "NET::ERR_CERT_COMMON_NAME_INVALID"	238
負載平衡器顯示處理時間延長	238
負載平衡器會傳送 000 的回應代碼	239
負載平衡器產生 HTTP 錯誤	239
HTTP 400：錯誤的請求	239
HTTP 401：未經授權	240
HTTP 403：禁止	240
HTTP 405：方法不允許	240
HTTP 408：請求逾時	240
HTTP 413：承載過大	240
HTTP 414：URI 過長	241
HTTP 460	241
HTTP 463	241
HTTP 464	241
HTTP 500：內部伺服器錯誤	241
HTTP 501：未導入	242
HTTP 502：無效的閘道	242
HTTP 503：服務無法使用	242
HTTP 504：閘道逾時	242
HTTP 505：不支援的版本	243
儲存空間不足	243
HTTP 561：未經授權	243
目標產生了 HTTP 錯誤	243
AWS Certificate Manager 憑證無法使用	243

不支援多行標頭	243
使用資源對應疑難排解狀況不良的目標	244
配額	246
文件歷史紀錄	249
.....	ccliv

什麼是 Application Load Balancer ？

Elastic Load Balancing 會自動將傳入流量分配到一或多個可用區域中的多個目標，例如 EC2 執行個體、容器和 IP 地址。其會監控已註冊目標的運作狀態，並且僅將流量路由至運作狀態良好的目標。當傳入流量隨著時間發生變化，Elastic Load Balancing 會擴展您的負載平衡器。他可以自動擴展以因應絕大多數的工作負載。

Elastic Load Balancing 支援下列負載平衡器：Application Load Balancer、Network Load Balancer、Gateway Load Balancer 和 Classic Load Balancer。您可以選取最符合您需要的負載平衡器類型。本指南主要介紹 Application Load Balancer。如需有關其他負載平衡器的詳細資訊，請參閱 [User Guide for Network Load Balancers](#)、[User Guide for Gateway Load Balancers](#) 和 《[Classic Load Balancer 使用者指南](#)》。

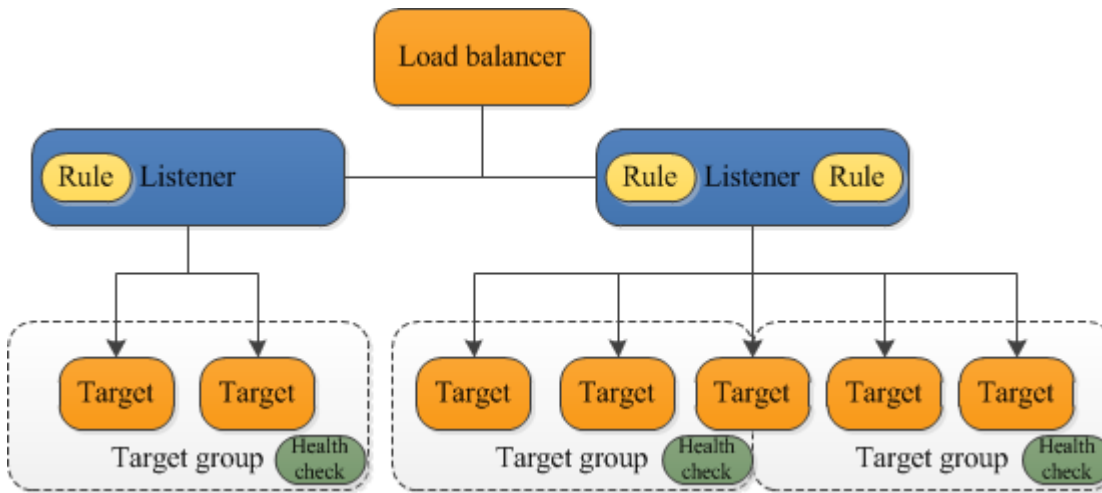
Application Load Balancer 元件

負載平衡器做為用戶端的單一聯絡點。負載平衡器會將傳入的應用程式流量分散到多個可用區域中的多個目標，例如 EC2 執行個體。這會提高您應用程式的可用性。您要為負載平衡器添加一個或多個接聽程式。

接聽程式會使用您所設定的通訊協定與連接埠，檢查來自用戶端的連線請求。您為接聽程式定義的規則，將決定負載平衡器路由請求到已登錄的目標的方法。每個規則由優先順序、一或多個動作及一或多個條件組成。滿足規則的條件時，即會執行它的動作。您必須為每個接聽程式定義預設規則，也可以選擇性地定義額外的規則。

每個目標群組會使用您指定的通訊協定和連接埠號碼，將請求路由至一個或多個已註冊的目標，例如 EC2 執行個體。您可以向多個目標群組註冊任一目標。您可以針對每個目標群組設定運作狀態檢查。凡已註冊至負載平衡器的接聽程式規則中指定之目標群組的所有目標，系統將對其執行運作狀態檢查。

下圖說明基本元件。請注意，每個接聽程式包含預設規則，而一個接聽程式包含另一個規則，可將請求路由至不同的目標群組。一個目標向兩個目標群組註冊。



如需詳細資訊，請參閱下列文件：

- [負載平衡器](#)
- [接聽程式](#)
- [目標群組](#)

Application Load Balancer 概觀

Application Load Balancer 在應用程式層 (開放系統互連 (OSI) 模型的第七層) 運作。當負載平衡器收到請求後，它會依優先順序評估接聽程式規則，以決定要套用的規則，然後從目標群組中選取規則動作的目標。您可以設定接聽程式規則，以根據應用程式流量的內容，將請求路由到不同的目標群組。即使一個目標向多個目標群組註冊，每個目標群組的路由都是獨立運作。您可以設定在目標群組層級上使用的路由演算法。預設路由演算法是循環式；或者，您可以指定最少未完成的請求路由演算法。

您可以依據需求變更，為負載平衡器新增和移除目標，而不會中斷應用程式整體的請求流程。當應用程式的流量隨著時間發生變化，Elastic Load Balancing 會擴展您的負載平衡器。Elastic Load Balancing 能夠自動擴展以因應絕大多數的工作負載。

您可以設定運作狀態檢查，用於監控已註冊目標的運作狀態，使負載平衡器只能傳送請求至運作狀態良好的目標。

如需詳細資訊，請參閱 Elastic Load Balancing 使用者指南中的 [Elastic Load Balancing 的運作方式](#)。

從 Classic Load Balancer 遷移的優點

使用 Application Load Balancer (而非 Classic Load Balancer) 具有下列優點：

- 支援 [路徑條件](#)。您可以為接聽程式設定規則，以根據請求中的 URL 來轉送請求。這可讓您將應用程式建構成較小的服務，以根據 URL 的內容，將請求路由傳送到正確的服務。
- 支援 [主機條件](#)。您可以為接聽程式設定規則，以根據 HTTP 標頭中的主機欄位來轉送請求。這可讓您使用單一負載平衡器，將請求路由至多個網域。
- 支援根據請求中的欄位 (例如 [HTTP 標頭條件](#) 和方法、查詢參數及來源 IP 地址) 來路由。
- 支援將請求路由至單一 EC2 執行個體上的多個應用程式。您可將一個執行個體或 IP 地址註冊到多個目標群組，每個目標群組都在不同的連接埠上。
- 支援將請求從一個 URL 重新導向另一個 URL。
- 支援傳回自訂 HTTP 回應。
- 支援透過 IP 地址註冊目標，包括位於負載平衡器的 VPC 外部的目標。
- 支援註冊 Lambda 函數做為目標。
- 支援負載平衡器在路由請求之前，透過企業或社交身分來驗證應用程式的使用者。
- 支援容器化的應用程式。Amazon Elastic Container Service (Amazon ECS) 可在排程任務時選取未使用的連接埠，並使用此連接埠向目標群組註冊該任務。這使您得以有效利用您的叢集。
- Support 獨立監視每個服務的健全狀況，因為健全狀況檢查是在目標群組層級定義，而且會在目標群組層級報告許多測 CloudWatch 量結果。將目標群組連接到 Auto Scaling 群組令您能夠隨需動態擴展各項服務。
- 存取日誌包含其他資訊，且以壓縮格式存放。
- 提升負載平衡器的效能。

如需各種負載平衡器類型支援的功能詳細資訊，請參閱 Elastic Load Balancing [產品比較](#)。

相關服務

Elastic Load Balancing 適用以下服務，可改善應用程式的可用性和可擴展性。

- Amazon EC2 – 在雲端執行應用程式的虛擬伺服器。您可以設定負載平衡器，將流量路由到 EC2 執行個體。
- Amazon EC2 Auto Scaling – 確保您正在執行所需數量的執行個體 (即使其中某個執行個體處於故障狀態)，並可讓您隨著執行個體需求的變更，自動增加或減少執行個體的數量。如果您啟用具有 Elastic Load Balancing 的 Auto Scaling，Auto Scaling 啟動的執行個體會自動在目標群組中註冊，而由 Auto Scaling 終止的執行個體會自動從目標群組中取消註冊。
- AWS Certificate Manager – 建立 HTTPS 接聽程式時，可以指定 ACM 所提供的憑證。負載平衡器會使用此憑證來終止連線，並解密來自用戶端的請求。如需詳細資訊，請參閱 [SSL 憑證](#)。

- Amazon CloudWatch — 可讓您監控負載平衡器，並視需要採取行動。如需詳細資訊，請參閱 [CloudWatch Application Load Balancer 的指標](#)。
- Amazon ECS – 可讓您在 EC2 執行個體叢集上執行、停止和管理 Docker 容器。您可以設定負載平衡器，將流量路由到容器。如需詳細資訊，請參閱 Amazon Elastic Container Service Developer Guide 中的 [Service load balancing](#)。
- AWS Global Accelerator – 改善應用程式的可用性和效能。使用加速器將流量分散到一個或多個 AWS 區域中的多個負載平衡器。如需詳細資訊，請參閱 [《AWS Global Accelerator 開發人員指南》](#)。
- Route 53 – 透過將網域名稱 (如 `www.example.com`) 轉換為電腦用來互相連線的數字 IP 地址 (例如 `192.0.2.1`)，提供可靠且經濟實惠的方式來將訪客路由至網站。AWS 會將 URL 指派至負載平衡器等資源。不過，您可能需要能讓使用者輕鬆記住的 URL。例如，您可以將網域名稱映射至負載平衡器。如需詳細資訊，請參閱《Amazon Route 53 開發人員指南》中的 [將流量路由到 ELB 負載平衡器](#)。
- AWS WAF – 您可以使用 AWS WAF 搭配 Application Load Balancer，以根據 Web 存取控制清單 (Web ACL) 中的規則來允許或封鎖請求。如需詳細資訊，請參閱 [應用程式負載平衡器和 AWS WAF](#)。

若要檢視與負載平衡器整合的服務資訊，請在 AWS Management Console 中選取您的負載平衡器，並選擇 Integrated services (整合服務) 索引標籤。

定價

使用負載平衡器時，您只需按實際用量付費。如需詳細資訊，請參閱 [Elastic Load Balancing 定價](#)。

Application Load Balancer 入門

本教學課程透過 Web 型介面提供應用程式負載平衡器的 AWS Management Console 實際操作簡介。若要建立第一個 Application Load Balancer，請完成以下步驟。

任務

- [開始之前](#)
- [步驟 1：設定您的目標群組](#)
- [步驟 2：選擇負載平衡器類型](#)
- [步驟 3：設定您的負載平衡器和接聽程式](#)
- [步驟 4：測試您的負載平衡器](#)
- [步驟 5：\(選用\) 刪除負載平衡器](#)

如需常見負載平衡器組態的示範，請參閱 [Elastic Load Balancing Demos](#)。

開始之前

- 決定您要用於 EC2 執行個體的兩個可用區域。在各個可用區域內設定至少包含一個公有子網路的 Virtual Private Cloud (VPC)。這些公有子網路將用於設定負載平衡器。您可以改為在上述可用區域的其他子網路中啟動您的 EC2 執行個體。
- 在各個可用區域內啟動至少一個 EC2 執行個體。請務必在每個 EC2 執行個體上安裝 Web 伺服器，例如 Apache 或 Internet Information Services (IIS)。確保這些執行個體的安全群組在連接埠 80 上允許 HTTP 存取。

步驟 1：設定您的目標群組

建立目標群組以用於請求路由。接聽程式的預設規則會將請求路由傳送至此目標群組中的已註冊目標。負載平衡器會使用您為目標群組定義的運作狀態檢查設定，檢查此目標群組中各目標的運作狀態。

使用主控台設定目標群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡中，選擇目標群組。

3. 選擇 Create target group (建立目標群組)。
4. 在基本組態下，將目標類型保留為執行個體。
5. 在目標群組名稱中，輸入新的目標群組名稱。
6. 保留預設通訊協定 (HTTP) 和連接埠 (80)。
7. 選取內含執行個體的 VPC。將通訊協定版本保留為 HTTP1。
8. 針對 Health checks (運作狀態檢查)，保留預設設定。
9. 選擇下一步。
10. 在註冊目標頁面上，完成以下步驟。這是建立負載平衡器的選用步驟。不過，如果您想要測試負載平衡器，並確保其會將流量路由到此目標，則必須註冊此目標。
 - a. 在可用執行個體中，選取一個或多個執行個體。
 - b. 保留預設連接埠 80，並選擇包含為下方待處理項目。
11. 選擇 Create target group (建立目標群組)。

步驟 2：選擇負載平衡器類型

Elastic Load Balancing 支援多種不同類型的負載平衡器。本教學課程旨在建立 Application Load Balancer。

使用主控台建立應用程式負載平衡器

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 於導覽列上，為負載平衡器選擇一個區域。請務必選擇您用於 EC2 執行個體的另一區域。
3. 在導覽窗格的 Load Balancing (負載平衡) 下方，選擇 Load Balancers (負載平衡器)。
4. 選擇 Create Load Balancer (建立負載平衡器)。
5. 針對 Application Load Balancer (應用程式負載平衡器)，選擇 Create (建立)。

步驟 3：設定您的負載平衡器和接聽程式

若要建立 Application Load Balancer，必須先提供負載平衡器的基本組態資訊，例如名稱、機制和 IP 地址類型。然後提供網路和一個或多個接聽程式的相關資訊。接聽程式是檢查連線請求的程序。使用通訊協定以及連接埠為用戶端與負載平衡器間的連線進行設定。如需受支援的通訊協定與連接埠之詳細資訊，請參閱[接聽程式組態](#)。

設定負載平衡器和接聽程式

1. 針對 Load balancer name (負載平衡器名稱)，輸入負載平衡器的名稱。例如 my-alb。
2. 對於 Scheme (機制) 和 IP address type (IP 地址類型)，保留預設值。
3. 在網路映射中，選取您用於 EC2 執行個體的 VPC。選取至少兩個可用區域，且每個區域至少選取一個子網路。針對用於啟動 EC2 執行個體的各個可用區域，先選取可用區域，接著選取該可用區域的一個公有子網路。
4. 對於安全群組，我們會為您在上一步中選取的 VPC 選取預設安全群組。您可以改為選擇其他安全群組。安全群組包含的規則必須允許負載平衡器與已註冊的目標在接聽程式連接埠和運作狀態檢查連接埠上通訊。如需詳細資訊，請參閱[安全群組規則](#)。
5. 對於接聽程式和路由，請保留預設通訊協定和連接埠，然後從清單中選取目標群組。如此設定的接聽程式，會在連接埠 80 上接受 HTTP 流量，並依預設將流量轉送至選取的目标群組。在此教學課程中，您不會建立 HTTPS 接聽程式。
6. 在預設動作中，選取您在「步驟 1：設定目標群組」中建立與註冊的目標群組。
7. (選用) 新增標籤以便對負載平衡器進行分類。每個負載平衡器的標籤索引鍵必須是唯一的。允許的字元包括英文字母、空格、數字 (UTF-8 格式) 以及以下特殊字元：+ - =。 _ : / @。不可使用結尾或前方空格。標籤值區分大小寫。
8. 複查您的組態，然後選擇 Create load balancer (建立負載平衡器)。一些預設屬性會在建立期間套用至負載平衡器。您可以在建立負載平衡器之後檢視和編輯這些屬性。如需詳細資訊，請參閱[負載平衡器屬性](#)。

步驟 4：測試您的負載平衡器

建立負載平衡器之後，確認其是否會將流量傳送到您的 EC2 執行個體。

測試您的負載平衡器

1. 系統通知您已成功建立負載平衡器之後，選擇 Close (關閉)。
2. 在導覽窗格的 LOAD BALANCING (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選取新建立的目標群組。
4. 選擇 Targets (目標) 並確認您的執行個體已就緒。若執行個體的狀態為 `initial`，原因可能是執行個體仍在進行註冊，或者未通過可視為運作狀態良好的運作狀態檢查次數下限。當至少有一個執行個體處於 `healthy` 狀態後，您即可測試您的負載平衡器。
5. 在導覽窗格的 Load Balancing (負載平衡) 下方，選擇 Load Balancers (負載平衡器)。
6. 選取新建立的負載平衡器。

7. 選擇 [說明] 並複製負載平衡器的 DNS 名稱 (例如, my-load-balancer-1234567890abcdef.cn)。將此 DNS 名稱貼至已連接網際網路的 web 瀏覽器的網址欄位。如果一切正常, 瀏覽器會顯示您的伺服器的預設頁面。
8. (選用) 若要定義額外的接聽程式規則, 請參閱[新增規則](#)。

步驟 5 : (選用) 刪除負載平衡器

在您的負載平衡器可用後, 將會根據持續執行時間收取一小時或不足一小時的費用。當您已不再需要負載平衡器時, 便可將其刪除。刪除負載平衡器後, 便會停止收取費用。請注意, 刪除負載平衡器並不會影響已向該負載平衡器註冊的目標。例如, EC2 執行個體會在刪除根據本指南建立的負載平衡器後繼續執行。

使用主控台刪除負載平衡器

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 下方, 選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器的核取方塊, 然後選擇動作、刪除。
4. 出現確認提示時, 選擇 Yes, Delete (是, 刪除)。

教學課程：使用 AWS CLI 建立 Application Load Balancer

本教學課程透過 AWS CLI

開始之前

- 使用以下命令來確認您執行的 AWS CLI 版本是否支援 Application Load Balancer。

```
aws elbv2 help
```

如果您收到錯誤訊息，指出 elbv2 不是有效的選擇，請更新 AWS CLI。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[安裝 AWS Command Line Interface](#)。

- 在虛擬私有雲端 (VPC) 中啟動您的 EC2 執行個體。確保這些執行個體的安全群組允許在接聽程式連接埠和運作狀態檢查連接埠上進行存取。如需詳細資訊，請參閱[目標安全群組](#)。
- 決定是建立 IPv4 負載平衡器還是雙堆疊負載平衡器。如果您想要用戶端僅使用 IPv4 地址來與負載平衡器通訊，請使用 IPv4。如果您想要用戶端同時使用 IPv4 和 IPv6 地址來與負載平衡器通訊，請選擇雙堆疊。也可以選擇雙堆疊使用 IPv6 與後端目標 (例如 IPv6 應用程式或雙堆疊子網路) 進行通訊。
- 請務必在每個 EC2 執行個體上安裝 Web 伺服器，例如 Apache 或 Internet Information Services (IIS)。確保這些執行個體的安全群組在連接埠 80 上允許 HTTP 存取。

建立負載平衡器

完成以下步驟，建立您的第一個負載平衡器。

建立負載平衡器

- 使用指[create-load-balancer](#)令建立負載平衡器。您必須指定並非來自相同可用區域的兩個子網路。

```
aws elbv2 create-load-balancer --name my-load-balancer \  
--subnets subnet-0e3f5cac72EXAMPLE subnet-081ec835f3EXAMPLE --security-groups  
sg-07e8ffd50fEXAMPLE
```

使用指[create-load-balancer](#)令建立 **dualstack** 負載平衡器。

```
aws elbv2 create-load-balancer --name my-load-balancer \  
--subnets subnet-0e3f5cac72EXAMPLE subnet-081ec835f3EXAMPLE --security-groups  
sg-07e8ffd50fEXAMPLE --ip-address-type dualstack
```

其輸出將包含負載平衡器的 Amazon Resource Name (ARN)，格式如下：

```
arn:aws:elasticloadbalancing:us-east-2:123456789012:loadbalancer/app/my-load-  
balancer/1234567890123456
```

2. 使用命 [create-target-group](#) 令建立目標群組，並指定用於 EC2 執行個體的相同 VPC。

可以建立 IPv4 和 IPv6 目標群組，以便與雙堆疊負載平衡器建立關聯。目標群組的 IP 地址類型會決定負載平衡器用於與後端目標通訊並檢查目標運作狀態的 IP 版本。

```
aws elbv2 create-target-group --name my-targets --protocol HTTP --port 80 \  
--vpc-id vpc-0598c7d356EXAMPLE --ip-address-type [ipv4 or ipv6]
```

其輸出將包含目標群組的 ARN，格式如下：

```
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-  
targets/1234567890123456
```

3. 使用 [register-targets](#) 命令向目標群組註冊您的執行個體：

```
aws elbv2 register-targets --target-group-arn targetgroup-arn \  
--targets Id=i-0abcdef1234567890 Id=i-1234567890abcdef0
```

4. 使用 [create-listener](#) 命令為您的負載平衡器建立具有預設規則以轉送請求至目標群組的接聽程式：

```
aws elbv2 create-listener --load-balancer-arn loadbalancer-arn \  
--protocol HTTP --port 80 \  
--default-actions Type=forward,TargetGroupArn=targetgroup-arn
```

其輸出將包含接聽程式的 ARN，格式如下：

```
arn:aws:elasticloadbalancing:us-east-2:123456789012:listener/app/my-load-  
balancer/1234567890123456/1234567890123456
```

5. (選擇性) 您可以使用下 [describe-target-health](#) 列命令驗證目標群組的已註冊目標的健全狀況：

```
aws elbv2 describe-target-health --target-group-arn targetgroup-arn
```

新增 HTTPS 接聽程式

如果您有使用 HTTP 接聽程式的負載平衡器，您可以如下所示新增 HTTPS 接聽程式。

將 HTTPS 接聽程式新增至您的負載平衡器

1. 使用以下其中一個方法來建立 SSL 憑證以與您的負載平衡器搭配使用：
 - 使用 AWS Certificate Manager (ACM) 建立或匯入憑證。如需詳細資訊，請參閱《AWS Certificate Manager 使用者指南》中的[請求憑證](#)或[匯入憑證](#)。
 - 使用 AWS Identity and Access Management (IAM) 上傳憑證。如需詳細資訊，請參閱 IAM 使用者指南中的[使用伺服器憑證](#)。
2. 使用 [create-listener](#) 命令來建立具有預設規則以將請求轉送至目標群組的接聽程式。建立 HTTPS 接聽程式時必須指定 SSL 憑證。請注意，您可以使用 `--ssl-policy` 選項來指定非預設的 SSL 政策。

```
aws elbv2 create-listener --load-balancer-arn loadbalancer-arn \  
--protocol HTTPS --port 443 \  
--certificates CertificateArn=certificate-arn \  
--default-actions Type=forward,TargetGroupArn=targetgroup-arn
```

新增以路徑為基礎的路由

如果您的接聽程式具有會將請求轉送至一個目標群組的預設規則，則您可以新增規則來根據 URL，將請求轉送至另一個目標群組。例如，您可以將一般請求路由到一個目標群組和請求，以向其他目標群組顯示影像。

將規則新增至具有路徑模式的接聽程式

1. 使用[create-target-group](#)指令建立目標群組：

```
aws elbv2 create-target-group --name my-targets --protocol HTTP --port 80 \  
--vpc-id vpc-0598c7d356EXAMPLE
```

2. 使用 [register-targets](#) 命令向目標群組註冊您的執行個體：

```
aws elbv2 register-targets --target-group-arn targetgroup-arn \  
--targets Id=i-0abcdef1234567890 Id=i-1234567890abcdef0
```

3. 使用 [create-rule](#) 命令，在 URL 包含指定的模式時，將規則新增至會將請求轉送至目標群組的接聽程式：

```
aws elbv2 create-rule --listener-arn listener-arn --priority 10 \  
--conditions Field=path-pattern,Values='/img/*' \  
--actions Type=forward,TargetGroupArn=targetgroup-arn
```

刪除負載平衡器

當您已不再需要負載平衡器和目標群組時，便可將其刪除，如下所示：

```
aws elbv2 delete-load-balancer --load-balancer-arn loadbalancer-arn  
aws elbv2 delete-target-group --target-group-arn targetgroup-arn
```

Application Load Balancer

負載平衡器做為用戶端的單一聯絡點。用戶端將請求傳送到負載平衡器，而負載平衡器將請求傳送到如 EC2 執行個體等的目標。若要設定您的負載平衡器，您需要建立 [目標群組](#)，然後使用您的目標群組來登錄目標。您也可以建立 [接聽程式](#)，以檢查來自用戶端的連線請求，並建立接聽程式規則，將來自用戶端的請求路由到一或多個目標群組中的目標。

如需詳細資訊，請參閱 Elastic Load Balancing 使用者指南中的 [Elastic Load Balancing 的運作方式](#)。

目錄

- [負載平衡器的子網路](#)
- [負載平衡器安全群組](#)
- [負載平衡器狀態](#)
- [負載平衡器屬性](#)
- [IP 地址類型](#)
- [Application Load Balancer 資源對應](#)
- [負載平衡器連線](#)
- [跨區域負載平衡](#)
- [刪除保護](#)
- [去同步緩解模式](#)
- [主機標頭保留](#)
- [應用程式負載平衡器和 AWS WAF](#)
- [建立 Application Load Balancer](#)
- [Application Load Balancer 的可用區域](#)
- [Application Load Balancer 的安全群組](#)
- [Application Load Balancer 的 IP 地址類型](#)
- [Application Load Balancer 的標籤](#)
- [刪除 Application Load Balancer](#)
- [區域轉移](#)

負載平衡器的子網路

建立 Application Load Balancer 時，必須啟用內含目標的區域。若要啟用區域，請在該區域中指定子網路。Elastic Load Balancing 會在您指定的每個區域建立負載平衡器節點。

考量事項

- 每個已啟用的區域擁有至少一個已註冊的目標時，負載平衡器的效率最高。
- 如果您在某個區域內註冊目標但未啟用該區域，這些已註冊的目標便不會接收來自負載平衡器的流量。
- 如果您為負載平衡器啟用多個區域，這些區域必須是相同類型。例如，您無法同時啟用可用區域和 Local Zone。
- 您可以指定與您共用的子網路。

Application Load Balancers 支援以下子網路類型。

子網類型

- [可用區域子網路](#)
- [Local Zone 子網路](#)
- [Outpost 子網路](#)

可用區域子網路

您必須選取至少兩個可用區域子網路。將適用以下限制：

- 每個子網路都必須來自不同的可用區域。
- 為了確保負載平衡器可以適當調整規模，請確認負載平衡器的每個可用區域子網路有一個 CIDR 區塊，並具有至少一個 /27 位元遮罩 (例如，10.0.0.0/27)，且每個子網路至少有八個可用 IP 地址。需要八個可用 IP 地址，才能讓負載平衡器視需要橫向擴展。負載平衡器會使用這些 IP 地址與目標建立連線。如果沒有這些地址，Application Load Balancer 可能會在嘗試取代節點時遇到困難，而導致其變成失敗狀態。

注意：如果嘗試擴展時，Application Load Balancer 子網路用完可用的 IP 地址，Application Load Balancer 將在容量不足的情況下執行。在此期間，舊節點將繼續為流量提供服務，但是在嘗試建立連線時，延遲的擴展嘗試可能會導致 5xx 錯誤或逾時。

Local Zone 子網路

您可以指定一個或多個 Local Zone 子網路。將適用以下限制：

- 您無法搭 AWS WAF 配負載平衡器使用。
- 您無法將 Lambda 函數做為目標使用。
- 您無法使用黏性工作階段或應用程式黏性。

Outpost 子網路

您可以指定單一的 Outpost 子網路。將適用以下限制：

- 內部部署資料中心必須已安裝和設定 Outpost。Outpost 與其 AWS 區域之間必須有可靠的網路連線。如需詳細資訊，請參閱 [AWS Outposts 使用者指南](#)。
- 負載平衡器在負載平衡器節點的 Outpost 上需要兩個 large 執行個體。下表所示是支援的執行個體類型。負載平衡器會視需要擴展，一次調整一個節點的大小 (從 large 到 xlarge，然後從 xlarge 到 2xlarge，接著從 2xlarge 到 4xlarge)。將節點擴展至最大的執行個體大小之後，如果您需要額外的容量，負載平衡器會將 4xlarge 執行個體新增為負載平衡器節點。如果您沒有足夠的執行個體容量或可用的 IP 地址來擴展負載平衡器，負載平衡器會向 [AWS Health Dashboard](#) 報告事件，且負載平衡器狀態會變為 active_impaired。
- 您可以依執行個體 ID 或 IP 地址來註冊目標。如果您在該 AWS 地區註冊前哨站的目標，則不會使用它們。
- 無法使用以下功能：作為目標的 Lambda 函數、AWS WAF 整合、粘性會話、身分驗證支援以及與 AWS Global Accelerator 整合。

Application Load Balancer 可部署在 Outpost 的 c5/c5d、m5/m5d 或 r5/r5d 執行個體上。下表顯示負載平衡器可在 Outpost 上使用的每個執行個體類型的大小和 EBS 磁碟區：

執行個體類型和大小	EBS 磁碟區 (GB)
c5/c5d	
大型	50
xlarge	50

執行個體類型和大小	EBS 磁碟區 (GB)
2xlarge	50
4xlarge	100
m5/m5d	
大型	50
xlarge	50
2xlarge	100
4xlarge	100
r5/r5d	
大型	50
xlarge	100
2xlarge	100
4xlarge	100

負載平衡器安全群組

安全群組扮演防火牆的角色，可控制允許進出負載平衡器的流量。您可以選擇連接埠和通訊協定，以同時允許傳入和傳出流量。

與負載平衡器相關聯之安全群組的規則，在接聽程式連接埠和運作狀態檢查連接埠上都必須允許這兩個方向的流量。當您將接聽程式新增至負載平衡器，或更新目標群組的運作狀態檢查連接埠時，您必須檢閱安全群組規則，以確保它們在新的連接埠上同時允許這兩個方向的流量。如需詳細資訊，請參閱 [建議的規則](#)。

負載平衡器狀態

負載平衡器可以是以下其中一個狀態：

provisioning

正在設定負載平衡器。

active

負載平衡器已設定完成並準備好路由流量。

active_impaired

負載平衡器正在路由流量，但不具備擴展所需的資源。

failed

無法設定的負載平衡器。

負載平衡器屬性

以下是負載平衡器屬性：

access_logs.s3.enabled

指出在 Amazon S3 中存放的存取日誌是否啟用。預設值為 `false`。

access_logs.s3.bucket

存取日誌的 Amazon S3 儲存貯體名稱。如果啟用存取日誌，則此為必要屬性。如需詳細資訊，請參閱 [啟用存取日誌](#)。

access_logs.s3.prefix

Amazon S3 儲存貯體中的位置字首。

client_keep_alive.seconds

用戶端保持活動的值，以秒為單位。預設值為 3600 秒。

deletion_protection.enabled

表示是否已啟用刪除保護。預設值為 `false`。

idle_timeout.timeout_seconds

閒置逾時值 (以秒為單位)。預設值為 60 秒。

ipv6.deny_all_igw_traffic

封鎖網際網路閘道 (IGW) 對負載平衡器的存取，以防止透過網際網路閘道對內部負載平衡器進行非預期存取。如果是面向網際網路的負載平衡器，設為 `false`，如果是內部負載平衡器，則設為 `true`。此屬性不會阻止非 IGW 網際網路存取 (例如透過對等互連、Transit Gateway 或)。AWS Direct Connect AWS VPN

routing.http.desync_mitigation_mode

決定負載平衡器如何處理可能對應用程式造成安全風險的請求。可能的值為 `monitor`、`defensive` 和 `strictest`。預設值為 `defensive`。

routing.http.drop_invalid_header_fields.enabled

指出具有無效標頭欄位的 HTTP 標頭是否已被負載平衡器 (`true`) 移除或已路由至目標 (`false`)。預設值為 `false`。如 HTTP 欄位名稱登錄檔中所述，Elastic Load Balancing 會要求有效的 HTTP 標頭名稱符合規則運算式 `[-A-Za-z0-9]+`。每個名稱由英數字元或連字號組成。如果要從請求中移除不符合此模式的 HTTP 標頭，請選取 `true`。

routing.http.preserve_host_header.enabled

指示 Application Load Balancer 是否應保留 HTTP 請求中的 Host 標頭，並將該標頭傳送到目標而不進行任何變更。可能的值為 `true` 和 `false`。預設值為 `false`。

routing.http.x_amzn_tls_version_and_cipher_suite.enabled

指示在向目標傳送用戶端請求前，是否在用戶端請求中新增這兩個標頭 (`x-amzn-tls-version` 和 `x-amzn-tls-cipher-suite`) (內含與交涉 TLS 版本和密碼套件相關的資訊)。 `x-amzn-tls-version` 標頭包含有關與用戶端交涉的 TLS 通訊協定版本的資訊，而 `x-amzn-tls-cipher-suite` 標頭包含有關與用戶端交涉的密碼套件的資訊。兩個標頭都採用 OpenSSL 格式。此屬性的可能值為 `true` 和 `false`。預設值為 `false`。

routing.http.xff_client_port.enabled

指示 X-Forwarded-For 標頭是否應該保留用戶端用來連線到負載平衡器的來源連接埠。可能的值為 `true` 和 `false`。預設值為 `false`。

routing.http.xff_header_processing.mode

可讓您在 Application Load Balancer 將 HTTP 請求傳送至目標之前修改、保留或移除該請求中的 X-Forward-For 標頭。可能的值為 `append`、`preserve` 和 `remove`。預設值為 `append`。

- 如果該值為 `append`，Application Load Balancer 會在將 HTTP 請求傳送至目標之前將 HTTP 請求中的 (最近一次跳轉的) 用戶端 IP 地址新增至該請求中的 X-Forward-For 標頭。

- 如果該值為 `preserve`，Application Load Balancer 會保留 HTTP 請求中的 `X-Forward-For` 標頭，並將該請求傳送到目標，而不進行任何變更。
- 如果該值為 `remove`，Application Load Balancer 會在將 HTTP 請求傳送至目標之前移除該請求中的標頭 `X-Forward-For`。

`routing.http2.enabled`

指出是否啟用 HTTP/2。預設值為 `true`。

`waf.fail_open.enabled`

指出在無法將要求轉送至目標時，是否允許已 AWS WAF 啟用的負載平衡器將要求路由至目標 AWS WAF。可能的值為 `true` 和 `false`。預設值為 `false`。

Note

引入了 `routing.http.drop_invalid_header_fields.enabled` 屬性，可提供 HTTP 去同步保護。新增了 `routing.http.desync_mitigation_mode` 屬性，可為應用程式提供更全面的 HTTP 去同步保護。您不需要同時使用這兩個屬性，可選擇其中一個屬性，具體取決於您的應用程式需求。

IP 地址類型

您可以設定 IP 地址類型，用戶端可以使用此類型的 IP 地址存取面向網際網路的負載平衡器和內部負載平衡器。

應用程式負載平衡器支援下列 IP 地址類型：

ipv4

用戶端必須使用 IPv4 地址 (例如，192.0.2.1) 才能連接至負載平衡器

dualstack

用戶端可以使用 IPv4 地址 (例如，192.0.2.1) 和 IPv6 地址 (例如，2001:0db8:85a3:0:0:8a2e:0370:7334) 連接至負載平衡器。

考量事項

- 負載平衡器會根據目標群組的 IP 地址類型與目標進行通訊。

- 當您啟用負載平衡器的雙堆疊模式時，Elastic Load Balancing 會提供負載平衡器的 AAAA DNS 記錄。使用 IPv4 地址與負載平衡器通訊的用戶端可解析 A DNS 記錄。使用 IPv6 地址與負載平衡器通訊的用戶端可解析 AAAA DNS 記錄。
- 透過網際網路閘道存取內部雙堆疊負載平衡器會遭到封鎖，以防止來自網際網路的非預期存取。但是，這並不會阻止非 IGW 網際網路存取 (例如透過對等互連、Transit Gateway 或)。AWS Direct Connect AWS VPN

dualstack-without-public-ipv4

用戶端必須使用 IPv6 位址 (例如，2001:0 資料庫 8:85 和 3:0:8 a2e: 0370:7334) 連線到負載平衡器。

考量事項

- Application Load Balancer 身份驗證僅在連線到身分識別提供者 (IdP) 或 Amazon Cognito 端點時支援 IPv4。如果沒有公用 IPv4 位址，負載平衡器將無法完成驗證程序，進而導致 HTTP 500 錯誤。

如需 IP 位址類型的詳細資訊，請參閱[Application Load Balancer 的 IP 地址類型](#)。

Application Load Balancer 資源對應

應用程式負載平衡器資源對應提供負載平衡器架構的互動式顯示，包括其關聯的接聽程式、規則、目標群組和目標。資源對應也會反白顯示所有資源之間的關係和路由路徑，以視覺化方式呈現負載平衡器的組態。

使用主控台檢視應用程式負載平衡器的資源對應

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 選擇資源對映索引標籤以顯示負載平衡器的資源對映。

資源對映元件

地圖檢視

「應用程式負載平衡器」資源對映中有兩種檢視：「概觀」和「狀態不良的目標對映」。預設情況下會選取「概觀」，並顯示所有負載平衡器的資源。選取「狀況不良的目標對映」檢視只會顯示狀況不良的目標及其相關聯的資源。

狀態不良的目標對映檢視可用來疑難排解未通過健康狀態檢查的目標。如需詳細資訊，請參閱 [使用資源對應疑難排解狀況不良的目標](#)。

資源群組

應用程式負載平衡器資源對映包含四個資源群組，每種資源類型各一個。資源群組為「監聽器」、「規則」、「目標群組」和「目標」。

資源並排

群組中的每個資源都有自己的圖標，其中會顯示有關該特定資源的詳細資訊。

- 將游標暫留在資源圖標上，會反白顯示其與其他資源之間的關係。
- 選取資源圖標會反白該資源與其他資源之間的關係，並顯示有關該資源的其他詳細資訊。
 - 規則條件：每個規則的條件。
 - 目標群組健全狀況摘要：每個健全狀況狀態的已註冊目標數目。
 - 目標健全狀況目標目前的健全狀況狀態和說明。

Note

您可以關閉 [顯示資源詳細資料] 以隱藏資源對映中的其他詳細資料。

- 每個資源圖標都包含一個連結，選取此連結後，會導覽至該資源的詳細資訊頁面。
 - 監聽器-選取偵聽程式通訊協定：連接埠。例如：HTTP:80
 - 規則-選取規則動作。例如：Forward to target group
 - 目標群組-選取目標群組名稱。例如：my-target-group
 - 目標-選取目標 ID。例如：i-1234567890abcdef0

匯出資源對應

選取 [匯出] 可讓您選擇將應用程式負載平衡器資源對映的目前檢視匯出為 PDF。

負載平衡器連線

處理要求時，負載平衡器會維護兩個連線：一個與用戶端的連線，另一個與目標的連線。負載平衡器與用戶端之間的連線也稱為前端連線。負載平衡器與目標之間的連線也稱為後端連線。

連線閒置逾時

連線閒置逾時是指在負載平衡器關閉連線之前，現有的用戶端或目標連線可以保持非作用中狀態，而不會傳送或接收任何資料。

若要確保冗長的作業 (例如檔案上傳) 有時間完成，請在每個閒置逾時期間結束之前傳送至少 1 個位元組的資料，並視需要增加閒置逾時期間的長度。也建議您將應用程式的閒置逾時設定為大於負載平衡器所設定的閒置逾時。否則，如果應用程式不正常地關閉與負載平衡器的 TCP 連線，負載平衡器可能會在收到指示連線已關閉的封包之前傳送請求給應用程式。如果是這種情況，則負載平衡器會向用戶端傳送「HTTP 502 錯誤閘道」錯誤。

根據預設，Elastic Load Balancing 會將負載平衡器的閒置逾時值設定為 60 秒或 1 分鐘。使用下列程序來設定不同的閒置逾時值。

使用主控台更新連線閒置逾時值

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在 [流量組態] 下，輸入 [連線閒置逾時] 的值。有效範圍為 1 到 4000 秒。
6. 選擇儲存變更。

若要使用更新閒置逾時值 AWS CLI

以 [屬性來使用](#) `modify-load-balancer-attributes` `idle_timeout.timeout_seconds` 命令。

HTTP 用戶端保持活動時間

HTTP 用戶端保持活動持續時間是「Application Load Balancer」維持與用戶端之間持續 HTTP 連線的最長時間長度。設定的 HTTP 用戶端 keepalive 持續時間過後，Application Load Balancer 會接受一個要求，並傳回正常關閉連線的回應。

負載平衡器傳送的回應類型取決於用戶端連線所使用的 HTTP 版本。對於使用 HTTP 1.x 連線的用戶端，負載平衡器會傳送包含欄位 `Connection: close` 的 HTTP 標頭。對於使用 HTTP/2 連線的用戶端，負載平衡器會傳送框架。GOAWAY

根據預設，應用程式負載平衡器會將 HTTP 用戶端保持作用持續時間值設定為 3600 秒或 1 小時。HTTP 用戶端保持活動持續時間不能關閉或設定為低於 60 秒的最低值，但是您可以將 HTTP 用戶端保持活動持續時間增加到最長 604800 秒或 7 天。應用程式負載平衡器會在一開始與用戶端建立 HTTP 連線時，開始啟動 HTTP 用戶端持續作用期間。當沒有流量時，持續時間期間會繼續執行，並且在建立新連線之前不會重設。

Note

將應用程式負載平衡器的 IP 位址類型切換至 `dualstack-without-public-ipv4` 負載平衡器時，會等待所有作用中的連線完成。若要欺騙切換應用程式負載平衡器 IP 位址類型所需的時間量，請考慮降低 HTTP 用戶端保持活動時間。

應用程式負載平衡器會在初始連線期間指派 HTTP 用戶端保持作用持續時間一次。當更新 HTTP 用戶端保持活動持續時間時，這可能會導致同時連接與不同的 HTTP 用戶端保持活動持續時間值。現有的連線會保留在其初始連線期間套用的 HTTP 用戶端保持活動持續時間值，而任何新連線都會收到更新的 HTTP 用戶端保持活動持續時間值。

使用主控台更新用戶端保持作用持續時間值

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在流量組態下，輸入 HTTP 用戶端保持作用持續時間的值。有效範圍為 60 到 604800 秒。
6. 選擇儲存變更。

使用更新用戶端保持作用持續時間值 AWS CLI

以 [屬性來使用](#) `modify-load-balancer-attributes` `client_keep_alive.seconds` 命令。

跨區域負載平衡

使用 Application Load Balancer 時，跨區域負載平衡依預設會開啟，而且無法在負載平衡器層級進行變更。如需詳細資訊，請參閱 Elastic Load Balancing User Guide 中的 [Cross-zone load balancing](#) 章節。

可以在目標群組層級關閉跨區域負載平衡。如需詳細資訊，請參閱 [the section called “關閉跨區域負載平衡”](#)。

刪除保護

為避免您的負載平衡器上遭意外刪除，您可以啟用刪除保護。您的負載平衡器的刪除保護預設為停用。

如果您為負載平衡器啟用刪除保護，則必須先停用才可刪除負載平衡器。

使用主控台來啟用刪除保護

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在組態下方，開啟刪除保護。
6. 選擇儲存變更。

使用主控台來停用刪除保護

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在組態頁面下方，關閉刪除保護。
6. 選擇儲存變更。

啟用或停用刪除保護 AWS CLI

以 [屬性來使用](#) `modify-load-balancer-attributesdeletion_protection.enabled` 命令。

去同步緩解模式

去同步緩解模式可保護應用程式免於因 HTTP 去同步而發生問題。負載平衡器會根據其威脅層級對每個要求進行分類，允許安全要求，然後根據您指定的緩和模式來降低風險。非同步緩和模式分為監控、防禦性和最嚴格。預設值為防禦模式，可針對 HTTP 非同步提供持久的緩和措施，同時維持應用程式的可用性。您可以切換至最嚴格模式，以確保應用程式只接收符合 [RFC 7230](#) 的請求。

`http_desync_guardian` 程式庫會分析 HTTP 請求，以防止 HTTP 去同步攻擊。如需詳細資訊，請參閱 [GitHub](#) 啟 [HTTP 不同步守護](#) 功能。

分類

分類如下：

- 合規 — 要求符合 RFC 7230，不會造成任何已知的安全威脅。
- 可接受 — 要求不符合 RFC 7230，但不會造成已知的安全威脅。
- 不明確 — 要求不符合 RFC 7230，但造成風險，因為各種 Web 伺服器 and 代理的處理方式不同。
- 嚴重 — 要求造成高安全性風險。負載平衡器會封鎖要求，傳送提供 400 回應至用戶端，並關閉用戶端連線。

如果要求不符合 RFC 7230，負載平衡器會增加

`DesyncMitigationMode_NonCompliant_Request_Count` 指標。如需詳細資訊，請參閱 [Application Load Balancer 指標](#)。

每個請求的分類都包含在負載平衡器存取日誌中。如果請求不符合規定，存取日誌會包含分類原因代碼。如需詳細資訊，請參閱 [分類原因](#)。

模式

下表說明 Application Load Balancer 如何根據模式和分類處理請求。

分類	監控模式	防禦性模式	最嚴格模式
合規	允許	已允許	允許
可接受	允許	允許	封鎖

分類	監控模式	防禦性模式	最嚴格模式
不明確	允許	允許 ¹	封鎖
嚴重	允許	封鎖	封鎖

¹ 路由傳送要求，但關閉用戶端和目標連接。如果負載平衡器在防禦模式下收到大量「不明確」請求，則可能會產生額外費用。這是因為每秒增加的新連線數目會提高每小時使用的負載平衡器容量單位 (LCU)。您可以使用 `NewConnectionCount` 指標，來比較負載平衡器在監控模式和防禦模式下建立新連線的方式。

使用主控台更新非同步緩和模式

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在封包處理下，在去同步緩解模式中，選擇防禦、最嚴格或監控。
6. 選擇儲存變更。

若要使用更新不同步緩和模式 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令，同時將 `routing.http.desync_mitigation_mode` 屬性設為 `monitor`、`defensive` 或 `strictest`。

主機標頭保留

啟用保留主機標頭屬性後，Application Load Balancer 會保留 HTTP 請求中的 Host 標頭，並將標頭傳送到目標，而不進行任何修改。如果 Application Load Balancer 收到多個 Host 標頭，則會全部保留。只會將接聽程式規則套用至收到的第一個 Host 標頭。

依預設，如果未啟用保留主機標頭屬性，則 Application Load Balancer 會以下列方式修改 Host 標頭：

未啟用主機標頭保留，且接聽程式連接埠不是預設連接埠時：未使用預設連接埠 (連接埠 80 或 443) 時，如果用戶端尚未附加連接埠號碼，則我們會將該號碼附加至主機標頭。例如，如果接聽程式連接埠

不是預設連接埠 (例如 8080)，則內含 Host: www.example.com 之 HTTP 請求中的 Host 標頭會修改為 Host: www.example.com:8080。

未啟用主機標頭保留，且接聽程式連接埠為預設連接埠 (連接埠 80 或 443) 時：對於預設的接聽程式連接埠 (連接埠 80 或 443)，我們不會將連接埠號碼附加至傳出主機標頭。已存在於傳入主機標頭中的任何連接埠號碼都會遭到移除。

下表顯示更多範例，說明 Application Load Balancer 如何根據接聽程式連接埠處理 HTTP 請求中的主機標頭。

接聽程式連接埠	範例請求	請求中的主機標頭	主機標頭保留已停用 (預設行為)	主機標頭保留已啟用
在預設的 HTTP/HTTPS 接聽程式上傳送請求。	GET / index.html HTTP/1.1 Host: example.com	example.com	example.com	example.com
請求在默認 HTTP 接聽程序上發送，並且主機標頭具有端口 (例如，80 或 443)。	GET / index.html HTTP/1.1 Host: example.com:80	example.com:80	example.com	example.com:80
請求具有絕對路徑。	GET https:// dns_name/ index.html HTTP/1.1 Host: example.com	example.com	dns_name	example.com
要求會在非預設接聽程式連接埠上傳送 (例如 8080)	GET / index.html HTTP/1.1 Host: example.com	example.com	example.com:8080	example.com

接聽程式連接埠	範例請求	請求中的主機標頭	主機標頭保留已停用 (預設行為)	主機標頭保留已啟用
在非預設接聽程式連接埠上傳送請求，並且主機標頭具有連接埠 (例如 8080)。	GET / index.html HTTP/1.1 Host: example.com:8080	example.com:8080	example.com:8080	example.com:8080

使用控制台啟用主機標頭保留

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在封包處理下方，開啟保留主機標頭。
6. 選擇儲存變更。

若要使用啟用主機標頭保留 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令，同時將 `routing.http.preserve_host_header.enabled` 屬性設為 `true`。

應用程式負載平衡器和 AWS WAF

您可以 AWS WAF 與應用程式負載平衡器搭配使用，根據 Web 存取控制清單 (Web ACL) 中的規則來允許或封鎖要求。如需詳細資訊，請參閱 AWS WAF Developer Guide 中的 [Working with web ACLs](#)。

根據預設，如果負載平衡器無法從中取得回應 AWS WAF，則會傳回 HTTP 500 錯誤，而且不會轉寄要求。如果您需要負載平衡器將要求轉送至目標，即使目標無法聯絡 AWS WAF，您可以啟用 AWS WAF 整合。若要檢查您的負載平衡器是否與整合 AWS WAF，請在中選取您的負載平衡器，AWS Management Console 然後選擇整合式服務索引標籤。

預先定義的網頁 ACL

啟用 AWS WAF 整合時，您可以選擇使用預先定義的規則自動建立新 Web ACL。預先定義的 Web ACL 包含三種 AWS 受管規則，可提供針對最常見安全威脅的保護。

- [AWSManagedRulesAmazonIpReputationList](#)-Amazon IP 信譽清單規則群組會封鎖通常與機器人或其他威脅相關聯的 IP 地址。如需詳細資訊，請參閱AWS WAF 開發人員指南中的 [Amazon IP 信譽清單受管規則群組](#)。
- [AWSManagedRulesCommonRuleSet](#)-[核心規則集 \(CRS\) 規則群組](#)針對各種漏洞的利用提供保護，包括 [OWASP Top 10](#) 等 OWASP 出版物中描述的一些高風險和常見漏洞。如需詳細資訊，請參閱AWS WAF 開發人員指南中的[核心規則集 \(CRS\) 受管規則群組](#)。
- [AWSManagedRulesKnownBadInputsRuleSet](#)-「已知錯誤輸入」規則群組會封鎖已知無效且與惡意利用或發現弱點相關聯的要求模式。如需詳細資訊，請參閱AWS WAF 開發人員指南中的[已知錯誤輸入受管理規則群組](#)。

啟用 AWS WAF 使用控制台

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在整合索引標籤上，展開 AWS Web 應用程式防火牆 (WAF)，然後選擇關聯 WAF Web ACL。
5. 在「Web ACL」下，選擇「自動建立預先定義的 Web ACL」，或選取現有的 Web ACL。
6. 在「規則動作」下，選擇「封鎖」或「計數」。
7. 選擇確認。

若要啟用 AWS WAF 失敗開啟，請使用 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令，同時將 `waf.fail_open.enabled` 屬性設為 `true`。

建立 Application Load Balancer

負載平衡器會從用戶端取得請求，然後將請求分發到目標群組中的目標。

開始之前，請確保虛擬私有雲端 (VPC) 在目標使用的每個區域中至少有一個公有子網路。如需詳細資訊，請參閱 [the section called “負載平衡器的子網路”](#)。

若要使用建立負載平衡器 AWS CLI，請參閱[教學課程：使用 AWS CLI 建立 Application Load Balancer](#)。

若要使用建立負載平衡器 AWS Management Console，請完成下列工作。

任務

- [步驟 1：設定目標群組](#)
- [步驟 2：註冊目標](#)
- [步驟 3：設定負載平衡器和接聽程式](#)
- [步驟 4：測試負載平衡器](#)

步驟 1：設定目標群組

設定目標群組可讓您註冊 EC2 執行個體等目標。設定負載平衡器時，在此步驟中設定的目標群組會作為接聽程式規則中的目標群組使用。如需詳細資訊，請參閱 [Application Load Balancer 的目標群組](#)。

使用主控台設定目標群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Target Groups (目標群組)。
3. 選擇 Create target group (建立目標群組)。
4. 在基本組態區段中，設定下列參數：
 - a. 在選擇目標類型中，選取執行個體來依執行個體 ID 指定目標，或選取 IP 地址來僅依 IP 地址指定目標。如果目標類型為 Lambda 函數，您可以選取運作狀態檢查區段中的啟用來啟用運作狀態檢查。
 - b. 在目標群組名稱中，輸入目標群組的名稱。
 - c. 視需要修改連接埠和通訊協定。
 - d. 如果目標類型為執行個體或 IP 地址，請選擇 IPv4 或 IPv6 作為 IP 地址類型，否則請跳至下一個步驟。

請注意，只有具有所選 IP 地址類型的目標才能包含在此目標群組中。建立目標群組後，便無法變更 IP 地址類型。
 - e. 對於 VPC，請選取虛擬私有雲端 (VPC)，內含要包含在目標群組中的目標。
 - f. 對於通訊協定版本，如果請求通訊協定為 HTTP/1.1 或 HTTP/2，則選取 HTTP1；如果請求通訊協定為 HTTP/2 或 gRPC，則選取 HTTP2；如果請求通訊協定為 gRPC，則選取 gRPC。
5. 在運作狀態檢查區段中，視需要修改預設設定。對於進階運作狀態檢查設定，請選擇運作狀態檢查連接埠、計數、逾時、間隔，並指定成功代碼。如果運作狀態檢查連續超過運作狀態不佳閾值的次

數，負載平衡器會停用該目標。當運作狀態檢查連續超過運作狀態不佳閾值次數時，負載平衡器會重新啟用該目標。如需詳細資訊，請參閱 [目標群組運作狀態檢查](#)。

6. (選用) 新增一個或多個標籤，如下所示：
 - a. 展開 Tags (標籤) 區段。
 - b. 選擇 Add tag (新增標籤)。
 - c. 輸入標籤索引鍵和標籤值。允許的字元包括 UTF-8 格式的英文字母、空格、數字，以及以下特殊字元：+ - = . _ : / @。不可使用結尾或前方空格。標籤值區分大小寫。
7. 選擇下一步。

步驟 2：註冊目標

您可以在目標群組中將 EC2 執行個體、IP 地址或 Lambda 函數註冊為目標。這是建立負載平衡器的選用步驟。不過，您必須註冊目標，才能確保負載平衡器會將流量路由至其中。

1. 在註冊目標頁面中，如下所示，新增一個或多個目標：
 - 如果目標類型為執行個體，請選取一個或多個執行個體，輸入一個或多個連接埠，然後選擇包含為下方待處理項目。
 - 如果目標類型是 IP 地址，請執行下列動作：
 - a. 從清單中選取網路 VPC，或選擇其他私人 IP 地址。
 - b. 手動輸入 IP 地址，或使用執行個體詳細資料尋找 IP 地址。一次最多可輸入五個 IP 地址。
 - c. 輸入用於將流量路由到指定 IP 地址的連接埠。
 - d. 選擇包含為下方待處理項目。
 - 如果目標類型為 Lambda，請選取 Lambda 函數，或輸入 Lambda 函數 ARN，然後選擇包含為下方待處理項目。
2. 選擇 Create target group (建立目標群組)。

步驟 3：設定負載平衡器和接聽程式

若要建立 Application Load Balancer，必須先提供負載平衡器的基本組態資訊，例如名稱、機制和 IP 地址類型。然後提供網路和一個或多個接聽程式的相關資訊。接聽程式是檢查連線請求的程序。使用通訊協定以及連接埠為用戶端與負載平衡器間的連線進行設定。如需受支援的通訊協定與連接埠之詳細資訊，請參閱 [接聽程式組態](#)。

使用主控台設定負載平衡器和接聽程式

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
 2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
 3. 選擇 Create Load Balancer (建立負載平衡器)。
 4. 在 Application Load Balancer (應用程式負載平衡器) 下，選擇 Create (建立)。
 5. 基本組態
 - a. 針對 Load balancer name (負載平衡器名稱)，輸入負載平衡器的名稱。例如 **my-alb**。Application Load Balancer 的名稱必須在該區域的 Application Load Balancer 和 Network Load Balancer 集內是唯一的。名稱最多可包含 32 個字元，而且只能包含英數字元和連字號。名稱開頭或結尾不得為連字號或 `internal-`。建立 Application Load Balancer 之後，就無法變更其名稱。
 - b. 針對 Scheme (機制)，選擇 Internet-facing (面對網際網路) 或 internal (內部)。面對網際網路的負載平衡器會透過網際網路將用戶端的請求路由至目標。內部負載平衡器會使用私有 IP 地址將請求路由至目標。
 - c. 針對 IP 位址類型，請選擇 IPv4、雙堆疊或不含公用 IPv4 的雙堆疊。如果您的用戶端使用 IPv4 位址與負載平衡器通訊，請選擇 IPv4。如果您的用戶端同時使用 IPv4 和 IPv6 地址來與負載平衡器通訊，請選擇 Dualstack (雙堆疊)。如果您的用戶端只使用 IPv6 位址與負載平衡器進行通訊，請選擇沒有公用 IPv4 的雙堆疊。
 6. 網路映射
 - a. 針對 VPC，選取您用於 EC2 執行個體的 VPC。如果您對機制選取面向網際網路，則只有具有網際網路閘道的 VPC 可供選擇。
 - b. 對於映射，請依照下列方式選取子網路，以啟用負載平衡器的區域：
 - 來自兩個或更多可用區域的子網路
 - 來自一個或多個 Local Zone 的子網路
 - 一個 Outpost 子網路
- 如需詳細資訊，請參閱 [the section called “負載平衡器的子網路”](#)。
- 如果是內部負載平衡器，會從子網路 CIDR 指派 IPv4 和 IPv6 地址。
- 如果您為負載平衡器啟用雙堆疊模式，請選取同時具有 IPv4 和 IPv6 CIDR 區塊的子網路。
7. 針對 Security groups (安全群組)，選取現有的安全群組，或建立新的安全群組。

負載平衡器的安全群組必須允許它與已註冊的目標在接聽程式連接埠和運作狀態檢查連接埠上通訊。主控台可以代替您建立負載平衡器的安全群組，內含允許此通訊的規則。您也可以建立安全群組並選取它。如需詳細資訊，請參閱 [建議的規則](#)。

(選用) 若要為您的負載平衡器建立新的安全群組，請選擇 Create a new security group (建立新的安全群組)。

8. 對於接聽程式和路由，預設接聽程式會在連接埠 80 上接受 HTTP 流量。您可以保留預設的通訊協定和連接埠，或選擇其他通訊協定和連接埠。對於 Default action (預設動作)，選擇您建立的目標群組。您可以選擇 Add listener (新增接聽程式) 以新增另一個接聽程式 (例如，HTTPS 接聽程式)。
9. (選擇性) 如果使用 HTTPS 接聽程式

對於安全政策，建議您一律使用最新的預先定義安全政策。

a. 針對預設 SSL/TLS 憑證，有下列選項可用：

- 如果您使用建立或匯入憑證 AWS Certificate Manager，請選取從 ACM，然後從選取憑證選取憑證選取憑證。
- 如果您使用 IAM 匯入憑證，請選取從 IAM，然後從選取憑證處選取您的憑證。
- 如果您想匯入憑證，但您的區域無法使用 ACM，請依序選取匯入和到 IAM。在憑證名稱欄位輸入憑證名稱。在憑證私有金鑰中，複製並貼上私有金鑰檔案的內容 (PEM 編碼)。在憑證內文中，複製並貼上公有金鑰憑證檔案的內容 (PEM 編碼)。在 Certificate Chain (憑證鏈) 中，將憑證鏈檔案的內容 (PEM 編碼) 複製並貼上，除非您使用的是自我簽署憑證，且不介意瀏覽器隱含地接受憑證。

b. (選擇性) 若要啟用相互驗證，請在用戶端憑證處理下啟用相互驗證 (MTL)。

啟用時，預設的相互 TLS 模式為傳遞。

如果您選取「使用信任存放區驗證」：

- 根據預設，會拒絕具有過期用戶端憑證的連線。若要變更此行為，請展開 [進階 MTL 設定]，然後在 [用戶端憑證到期] 下選取 [允許過期的用戶]
- 在 [信任存放區] 下選擇現有的信任存放區，或選擇 [新增信任存放區]
 - 如果您選擇 [新增信任存放區]，請提供信任存放區名稱、S3 URI 憑證授權單位位置，以及選擇性地提供 S3 URI 憑證撤銷清單位置。

10. (選用) 您可以在建立期間將其他服務與負載平衡器整合，在「使用服務整合最佳化」下方。

- 您可以選擇在現有或自動建立的 Web ACL 中加入負載平衡器的AWS WAF安全性保護。建立之後，即可在[AWS WAF 主控台](#)中管理 Web ACL。如需詳細資訊，請參閱[開發人員指南中的建立 Web ACL 與 AWS 資源的關聯或取消關聯](#)。AWS WAF
- 您可以選擇為您AWS Global Accelerator建立加速器，並將負載平衡器與加速器建立關聯。加速器名稱可以包含下列字元（最多 64 個字元）：a-z、A-Z、0-9。（句號）和-（連字號）。建立加速器之後，您可以在[AWS Global Accelerator 主控台](#)中對其進行管理。如需詳細資訊，請參閱AWS Global Accelerator 開發人員指南中的[建立負載平衡器時新增加速器](#)。

11. 標記和建立

- a. （選用）新增標籤以便對負載平衡器進行分類。每個負載平衡器的標籤索引鍵必須是唯一的。允許的字元包括英文字母、空格、數字 (UTF-8 格式) 以及以下特殊字元：+ - =。 _ : / @。不可使用結尾或前方空格。標籤值區分大小寫。
- b. 複查您的組態，然後選擇 Create load balancer (建立負載平衡器)。一些預設屬性會在建立期間套用至負載平衡器。您可以在建立負載平衡器之後檢視和編輯這些屬性。如需詳細資訊，請參閱 [負載平衡器屬性](#)。

步驟 4：測試負載平衡器

建立負載平衡器後，請確認 EC2 執行個體已通過初始運作狀態檢查。然後，您可以檢查負載平衡器是否正在向 EC2 執行個體傳送流量。若要刪除負載平衡器，請參閱[刪除 Application Load Balancer](#)。

若要測試負載平衡器

1. 建立網路負載平衡器之後，選擇 Close (關閉)。
2. 在導覽窗格中，選擇 Target Groups (目標群組)。
3. 選取新建立的目標群組。
4. 選擇 Targets (目標) 並確認您的執行個體已就緒。如果執行個體的狀態為 `initial`，通常是因為執行個體仍在註冊中。此狀態也可能表示執行個體尚未通過最低數量的運作狀態檢查，無法視為運作狀態良好。至少有一個執行個體的運作狀態為健康之後，您可以測試您的負載平衡器。如需詳細資訊，請參閱 [目標運作狀態](#)。
5. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
6. 選取新建立的負載平衡器。
7. 選擇 [說明]，然後複製網際網路對向或內部負載平衡器的 DNS 名稱 (例如，my-load-balancer-1234567890abcdef.eu)。

- 對於面向網際網路的負載平衡器，將 DNS 名稱貼至已連接網際網路的 Web 瀏覽器的網址欄位。
- 對於內部負載平衡器，請將 DNS 名稱貼至具有 VPC 私人連線的 Web 瀏覽器的網址欄位中。

如果一切設定都正常，瀏覽器會顯示伺服器的預設頁面。

8. 如果網頁未顯示，請參閱下列文件以取得其他組態說明和疑難排解步驟。

- 對於 DNS 相關問題，請參閱《Amazon Route 53 開發人員指南》中的[將流量路由到 ELB 負載平衡器](#)。
- 如需有關負載平衡器問題的資訊，請參閱[為 Application Load Balancer 進行疑難排解](#)。

Application Load Balancer 的可用區域

您可以隨時為您的負載平衡器啟用或停用可用區域。當您啟用可用區域之後，負載平衡器會開始將請求路由到該可用區域內已註冊的目標。如果您確認每個已啟用的可用區域擁有至少一個登錄的目標，您的負載平衡器會展現最高效率。

當您停用可用區域之後，該可用區域內的目標仍註冊到負載平衡器，但負載平衡器不會將請求路由給它們。

使用主控台更新可用區域

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在網路映射索引標籤中，選擇編輯子網路。
5. 若要啟用可用區域，請選取其核取方塊並選取子網路。如果可用的子網路只有一個，則會選取該子網路。
6. 若要變更已啟用可用區域的子網路，請從清單中選擇其中一個其他的子網路。
7. 若要停用可用區域，請清除其核取方塊。
8. 選擇儲存變更。

若要使用更新可用區域 AWS CLI

使用 [set-subnets](#) 命令。

Application Load Balancer 的安全群組

Application Load Balancer 的安全群組會控制允許到達和離開負載平衡器的流量。您必須確保負載平衡器可以在接聽程式連接埠和運作狀態檢查連接埠上與已註冊的目標通訊。當您將接聽程式新增到負載平衡器，或針對目標群組更新負載平衡器用來路由請求的運作狀態檢查連接埠時，您必須確認與負載平衡器相關聯的安全群組在新的連接埠上允許這兩個方向的流量。如果不是如此，您可以編輯目前相關聯之安全群組的規則，或將其他安全群組與負載平衡器建立關聯。您可以選擇要允許的連接埠和通訊協定。例如，您可以開放網際網路控制訊息通訊協定 (ICMP) 連線負載平衡器回應 ping 請求 (不過，ping 請求不會轉發到任何執行個體)。

建議的規則

對於面向網際網路的負載平衡器，建議您使用以下規則。

Inbound

Source	Port Range	Comment
0.0.0.0/0	####	在負載平衡器接聽程式連接埠上允許所有傳入流量

Outbound

Destination	Port Range	Comment
#####	#####	在執行個體接聽程式連接埠上允許流向執行個體的傳出流量
#####	#####	在運作狀態檢查連接埠上允許流向執行個體的傳出流量

對於內部負載平衡器，建議您使用以下規則。

Inbound

Source	Port Range	Comment
--------	------------	---------

<i>VPC CIDR</i>	<i>####</i>	在負載平衡器接聽程式連接埠上允許來自 VPC CIDR 的傳入流量
Outbound		
Destination	Port Range	Comment
<i>#####</i>	<i>#####</i>	在執行個體接聽程式連接埠上允許流向執行個體的傳出流量
<i>#####</i>	<i>#####</i>	在運作狀態檢查連接埠上允許流向執行個體的傳出流量

對於用作 Network Load Balancer 目標的 Application Load Balancer，建議您使用以下規則。

Inbound		
Source	Port Range	Comment
<i>### IP ##/CIDR</i>	<i>alb ####</i>	允許負載平衡器接聽程式連接埠上的傳入用戶端流量
<i>VPC CIDR</i>	<i>alb ####</i>	透 AWS PrivateLink 過負載平衡器接聽程式連接埠允許輸入用戶端流量
<i>VPC CIDR</i>	<i>alb ####</i>	允許來自 Network Load Balancer 的傳入運作狀態檢查流量
Outbound		
Destination	Port Range	Comment
<i>#####</i>	<i>#####</i>	在執行個體接聽程式連接埠上允許流向執行個體的傳出流量

#####

#####

在運作狀態檢查連接埠上允許
流向執行個體的傳出流量

請注意，Application Load Balancer 的安全群組使用連線追蹤，來追蹤來自 Network Load Balancer 的流量相關資訊。無論為 Application Load Balancer 設定的安全群組規則為何，都會發生此情況。若要進一步了解 Amazon EC2 連線追蹤，請參閱 Amazon EC2 使用者指南中的[安全群組連線追蹤](#)。

若要確保您的目標僅接收來自負載平衡器的流量，請限制與目標相關聯的安全群組，以僅接受負載平衡器的流量。這可以透過將負載平衡器的安全性群組設定為目標安全性群組的輸入規則中的來源來達成。

我們也建議您允許傳入 ICMP 流量，以支援路徑 MTU 探索。如需詳細資訊，請參閱 Amazon EC2 使用者指南中的[路徑 MTU 探索](#)。

更新相關聯的安全群組

您可以隨時更新與負載平衡器相關聯的安全群組。

使用主控台更新安全群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在安全性索引標籤中，選擇編輯。
5. 若要將安全群組與負載平衡器建立關聯，請選取安全群組。若要移除安全群組關聯，請選擇安全群組的 X 圖示。
6. 選擇儲存變更。

若要使用更新安全群組 AWS CLI

使用 [set-security-groups](#) 命令。

Application Load Balancer 的 IP 地址類型

您可以設定 Application Load Balancer，讓用戶端只能使用 IPv4 地址，或既可使用 IPv4 又可使用 IPv6 地址 (雙堆疊)，來與負載平衡器通訊。負載平衡器會根據目標群組的 IP 地址類型與目標進行通訊。如需詳細資訊，請參閱 [IP 地址類型](#)。

雙堆疊要求

- 您可以在建立負載平衡器時設定 IP 地址類型，並隨時更新它。
- 您為負載平衡器指定的 Virtual Private Cloud (VPC) 和子網路必須具有相關聯的 IPv6 CIDR 區塊。如需詳細資訊，請參閱《Amazon EC2 使用者指南》中的 [IPv6 地址](#)。
- 負載平衡器子網路的路由表必須路由 IPv6 流量。
- 負載平衡器的安全群組必須允許 IPv6 流量。
- 負載平衡器子網路的網路 ACL 必須允許 IPv6 流量。

在建立時設定 IP 地址類型

按照 [???](#) 的說明進行設定。

使用主控台更新 IP 地址類型

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在網路映射索引標籤上，選擇編輯 IP 地址類型。
5. 對於 IP 位址類型，請選擇 IPv4 以僅支援 IPv4 位址，選擇雙堆疊同時支援 IPv4 和 IPv6 位址，或選擇不含公用 IPv4 的雙堆以僅支援 IPv6 位址。
6. 選擇儲存變更。

若要使用更新 IP 位址類型 AWS CLI

使用 [set-ip-address-type](#) 命令。

Application Load Balancer 的標籤

標籤可幫助您以不同的方式來將負載平衡器分類，例如，根據目的、擁有者或環境。

您可以在每個負載平衡器中加入多個標籤。如果所新增的標籤，其索引鍵已經與負載平衡器相關聯，則此動作會更新該標籤的值。

使用標籤完成負載平衡器使用後，可將其自負載平衡器中移除。

限制

- 每一資源標籤數上限：50
- 索引鍵長度上限：127 個 Unicode 字元
- 數值長度上限：255 個 Unicode 字元
- 標籤鍵與值皆區分大小寫。允許的字元包括可用 UTF-8 表示的英文字母、空格和數字，還有以下特殊字元：`+ - = . _ : / @`。不可使用結尾或前方空格。
- 請勿在標籤名稱或值中使用 `aws:` 前置詞，因為它已保留供 AWS 使用。您不可編輯或刪除具此字首的標籤名稱或值。具此字首的標籤，不算在受資源限制的標籤計數內。

使用主控台來更新負載平衡器的標籤

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在標籤索引標籤上，選擇管理標籤，並執行下列一個或多個動作：
 - a. 若要更新標籤，請編輯 Key (索引鍵) 和 Value (值) 的值。
 - b. 若要新增標籤，請選擇新增標籤，然後輸入索引鍵和值的值。
 - c. 若要刪除標籤，請選擇標籤旁的 Remove (移除) 按鈕。
5. 完成標籤的更新作業後，請選擇儲存變更。

若要使用更新負載平衡器的標籤 AWS CLI

使用 [add-tags](#) 和 [remove-tags](#) 指令。

刪除 Application Load Balancer

在您的負載平衡器可用後，將會根據持續執行時間收取一小時或不足一小時的費用。當您不再需要負載平衡器時，可以將它刪除。刪除負載平衡器後，便會停止收取費用。

如果已啟用刪除保護，則無法刪除負載平衡器。如需詳細資訊，請參閱 [刪除保護](#)。

請注意，刪除負載平衡器不會影響其登錄目標。例如，您的 EC2 執行個體將繼續執行，且仍會登錄到他們的目標群組。若要刪除您的目標群組，請參閱 [刪除目標群組](#)。

使用主控台來刪除負載平衡器

1. 如果您網域有指向負載平衡器的 DNS 記錄，請將其指向新的位置，並等待 DNS 變生效，然後刪除負載平衡器。

範例：

- 如果記錄是存留時間 (TTL) 為 300 秒的 CNAME 記錄，請等待至少 300 秒，然後再繼續執行下一個步驟。
 - 如果記錄是 Route 53 別名 (A) 記錄，請至少等待 60 秒。
 - 如果使用 Route 53，則記錄變更需要 60 秒才能傳播到所有全域 Route 53 名稱伺服器。將此時間新增至正在更新之記錄的 TTL 值。
2. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
 3. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
 4. 選取負載平衡器，然後選擇動作、刪除負載平衡器。
 5. 出現確認提示時，請輸入 **confirm**，然後選擇 Delete (刪除)。

若要使用刪除負載平衡器 AWS CLI

使用 [delete-load-balancer](#) 指令。

區域轉移

區域轉移是 Amazon Route 53 應用程式復原控制器 (Route 53 ARC) 中的一個功能。使用區域轉移功能，您可以透過單一動作，將負載平衡器資源從受損的可用區域轉移。如此一來，您就可以繼續從 AWS 區域中其他運作狀態良好的可用區域進行操作。

啟動區域轉移後，負載平衡器會停止將資源的流量傳送至受影響的可用區域。Route 53 ARC 會立即建立區域轉移。不過，可能需要一點時間 (通常最多幾分鐘)，才能完成受影響可用區域中現有正在進行的連線。如需詳細資訊，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [How a zonal shift works: health checks and zonal IP addresses](#)。

只有在跨區域負載平衡關閉的情況下，Application Load Balancer 和 Network Load Balancer 才支援區域轉移。如果開啟跨區域負載平衡，就無法啟動區域轉移。如需詳細資訊，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [Resources supported for zonal shifts](#)。

在使用區域轉移之前，請檢閱以下內容：

- 區域轉移不支援跨區域負載平衡。必須關閉跨區域負載平衡才能使用此功能。
- 在 AWS Global Accelerator 中將 Application Load Balancer 作為加速器端點使用時，不支援區域轉移。
- 只能針對單一可用區域，啟動特定負載平衡器的區域轉移。無法針對多個可用區域啟動區域轉移。
- 當出現多個基礎設施問題影響服務時，AWS 會主動從 DNS 中移除區域負載平衡器 IP 地址。在啟動區域轉移之前，請務必檢查目前的可用區域容量。如果您的負載平衡器已關閉跨區域負載平衡，而且您使用了區域轉移來移除區域負載平衡器 IP 地址，則受區域轉移影響的可用區域也會失去目標容量。
- 如果 Application Load Balancer 是 Network Load Balancer 的目標，請務必從 Network Load Balancer 啟動區域轉移。如果從 Application Load Balancer 啟動區域轉移，Network Load Balancer 將無法辨識轉移，並會繼續將流量傳送至 Application Load Balancer。

如需詳細指南和資訊，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [Best practices with Route 53 ARC zonal shifts](#)。

啟動區域轉移

本程序中的步驟說明如何使用 Amazon EC2 主控台來啟動區域轉移。如需使用 Route 53 ARC 主控台啟動區域轉移的步驟，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [Starting a zonal shift](#)。

使用主控台啟動區域轉移

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡下，選擇負載平衡器。
3. 選取負載平衡器名稱。
4. 在整合索引標籤中的 Route 53 應用程式復原控制器下，選擇啟動區域轉移。
5. 選取要將流量移出的可用區域。
6. 選擇或輸入區域轉移的到期時間。區域轉移到期時間最初可設定為 1 分鐘至三天 (72 小時)。

所有區域轉移都是暫時的。您必須設定到期時間，但您可以稍後更新作用中的轉移以設定新的到期時間。

7. 輸入註解。如需要，可以稍後更新區域轉移以編輯註解。
8. 選取此核取方塊以確認啟動區域轉移會將流量從可用區域轉移，以減少應用程式的容量。
9. 選擇啟動。

使用 AWS CLI 啟動區域轉移

若要以程式設計方式使用區域轉移，請參閱 [Zonal Shift API Reference Guide](#)。

更新區域轉移

本程序中的步驟說明如何使用 Amazon EC2 主控台更新區域轉移。如需有關使用 Amazon Route 53 應用程式復原控制器主控台更新區域轉移的步驟，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [Updating a zonal shift](#)。

使用主控台更新區域轉移

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡下，選擇負載平衡器。
3. 選取具有作用中區域轉移的負載平衡器名稱。
4. 在整合索引標籤中的 Route 53 應用程式復原控制器下，選擇更新區域轉移。

這會開啟 Route 53 ARC 主控台以繼續更新。

5. 在設定區域轉移到期時間中，選擇性選取或輸入到期時間。
6. 在註解中，可編輯現有註解或輸入新註解。
7. 選擇更新。

使用 AWS CLI 更新區域轉移

若要以程式設計方式使用區域轉移，請參閱 [Zonal Shift API Reference Guide](#)。

取消區域轉移

本程序中的步驟說明如何使用 Amazon EC2 主控台取消區域轉移。如需使用 Amazon Route 53 應用程式復原控制器主控台取消區域轉移的步驟，請參閱 Amazon Route 53 Application Recovery Controller Developer Guide 中的 [Canceling a zonal shift](#)。

使用主控台取消區域轉移

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡下，選擇負載平衡器。
3. 選取具有作用中區域轉移的負載平衡器名稱。
4. 在整合索引標籤中的 Route 53 應用程式復原控制器下，選擇取消區域轉移。

這會開啟 Route 53 ARC 主控台以繼續取消。

5. 選擇取消區域轉移。
6. 在確認對話上，選擇繼續。

使用 AWS CLI 取消區域轉移

若要以程式設計方式使用區域轉移，請參閱 [Zonal Shift API Reference Guide](#)。

Application Load Balancer 的接聽程式

接聽程式是檢查連線請求的程序，必須使用您已設定的通訊協定與連接埠。開始使用 Application Load Balancer 之前，必須新增至少一個接聽程式。如果負載平衡器沒有接聽程式，就無法接收來自用戶端的流量。您為接聽程式定義的規則，將決定負載平衡器將請求路由到已註冊目標 (例如 EC2 執行個體) 的方法。

目錄

- [接聽程式組態](#)
- [接聽程式規則](#)
- [規則動作類型](#)
- [規則條件類型](#)
- [為 Application Load Balancer 建立 HTTP 接聽程式](#)
- [為 Application Load Balancer 建立 HTTPS 接聽程式](#)
- [Application Load Balancer 的接聽程式規則](#)
- [為 Application Load Balancer 更新 HTTPS 接聽程式](#)
- [Application Load Balancer 中的 TLS 相互驗證](#)
- [使用 Application Load Balancer 來驗證使用者身分](#)
- [HTTP 標頭和 Application Load Balancer](#)
- [接聽程式和規則的標籤](#)
- [刪除 Application Load Balancer 的接聽程式](#)

接聽程式組態

接聽程式支援下列通訊協定與連接埠：

- Protocols (通訊協定) : HTTP、HTTPS
- Ports (連接埠) : 1-65535

您可以使用 HTTPS 接聽程式來將加密和解密的工作卸載到您的負載平衡器，使得您的應用程式可以專注在商業邏輯上。如果接聽程式通訊協定是 HTTPS，則必須在接聽程式上至少部署一個 SSL 伺服器憑證。如需詳細資訊，請參閱 [為 Application Load Balancer 建立 HTTPS 接聽程式](#)。

如果您必須確保目標解密的是 HTTPS 流量 (而不是負載平衡器), 則可以建立在連接埠 443 上具有 TCP 接聽程式的 Network Load Balancer。使用 TCP 接聽程式時, 負載平衡器會將加密的流量傳遞給目標, 而不需要對流量進行解密。如需詳細資訊, 請參閱 [《Network Load Balancer 使用者指南》](#)。

應用程式負載平衡器提供的原生支援 WebSockets。您可以使用 HTTP 連線升級, 將現有的 HTTP/1.1 連線升級為 WebSocket (ws或wss) 連線。升級時, 用於要求 (與負載平衡器以及目標) 的 TCP WebSocket 連線會成為用戶端透過負載平衡器與目標之間的持續連線。您可以同時使 WebSockets 用 HTTP 和 HTTPS 接聽程式。您為監聽器選擇的選項會套用至 WebSocket 連線以及 HTTP 流量。[如需詳細資訊, 請參閱 Amazon CloudFront 開發人員指南中的通訊 WebSocket 協定如何運作。](#)

Application Load Balancer 對使用 HTTPS 接聽程式的 HTTP/2 提供原生支援。您可以使用 HTTP/2 連線平行傳送高達 128 個請求。您可以使用通訊協定版本, 使用 HTTP/2 將請求傳送至目標。如需詳細資訊, 請參閱 [通訊協定版本](#)。由於 HTTP/2 可更有效率地使用前端連線, 您可能會注意到用戶端和負載平衡器之間的連線數較少。您無法使用 HTTP/2 的伺服器推送功能。

如需詳細資訊, 請參閱 Elastic Load Balancing User Guide 中的 [Request routing](#)。


接聽程式規則

每個接聽程式都有預設動作, 也稱為預設規則。無法刪除預設規則, 且最後執行的一律是預設規則。您可以建立其他規則, 這些規則會由優先順序、一個或多個動作及一個或多個條件組成。您可以隨時新增或編輯規則。如需詳細資訊, 請參閱 [編輯規則](#)。

預設規則

建立接聽程式時, 您會定義預設規則的預設動作。預設規則不能有條件。如果沒有符合任何接聽程式規則的條件, 則會執行預設規則的動作。

以下顯示主控台中預設規則的範例：

Priority	Conditions (If)	Actions (Then) 
Last (default)	<i>If no other rule applies</i>	Forward to target group <ul style="list-style-type: none"> my-targets: 1 (100%) Group-level stickiness: Off

規則優先順序

每個規則具有優先順序。依優先順序評估規則，從最低值到最高值。預設規則最後評估。您可以隨時變更非預設規則的優先順序。您無法變更預設規則的優先順序。如需詳細資訊，請參閱 [更新規則優先順序](#)。

規則動作

每個規則動作都包含類型、優先順序和執行動作所需的資訊。如需詳細資訊，請參閱 [規則動作類型](#)。

規則條件

每個規則條件具有類型和組態資訊。滿足規則的條件時，即會執行它的動作。如需詳細資訊，請參閱 [規則條件類型](#)。

規則動作類型

以下是接聽程式規則支援的動作類型：

authenticate-cognito

[HTTPS 接聽程式] 使用 Amazon Cognito 來驗證使用者身分。如需詳細資訊，請參閱 [使用 Application Load Balancer 來驗證使用者身分](#)。

authenticate-oidc

[HTTPS 接聽程式] 使用與 OpenID Connect (OIDC) 相容的身分提供者來驗證使用者。

fixed-response

傳回自訂的 HTTP 回應。如需詳細資訊，請參閱 [固定回應動作](#)。

forward

將請求轉送到指定的目標群組。如需詳細資訊，請參閱 [轉送動作](#)。

redirect

將請求從一個 URL 重新導向到另一個 URL。如需詳細資訊，請參閱 [重新導向動作](#)。

優先順序最低的動作會先執行。每個規則必須包含剛好以下其中一個動作：forward、redirect 或 fixed-response，而且必須是最後要執行的動作。

如果通訊協定版本為 gRPC 或 HTTP/2，則唯一支援的動作是 forward 動作。

固定回應動作

您可以使用 fixed-response 動作來捨棄用戶端請求，並傳回自訂 HTTP 回應。您可以使用此動作來傳回 2XX、4XX 或 5XX 回應代碼和選用的訊息。

採取 fixed-response 動作時，會在存取日誌中記錄重新導向目標的動作和 URL。如需詳細資訊，請參閱 [存取日誌項目](#)。會在 HTTP_Fixed_Response_Count 指標中報告成功的 fixed-response 動作計數。如需詳細資訊，請參閱 [Application Load Balancer 指標](#)。

Example 範例固定回應動作 AWS CLI

您可以在建立或修改規則時指定動作。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。下列動作傳送包含指定的狀態碼和訊息本文的固定回應。

```
[
  {
    "Type": "fixed-response",
    "FixedResponseConfig": {
      "StatusCode": "200",
      "ContentType": "text/plain",
      "MessageBody": "Hello world"
    }
  }
]
```

轉送動作

您可以使用 forward 動作將請求路由傳送到一或多個目標群組。如果您為一個 forward 動作指定多個目標群組，則必須為每個目標群組指定加權。每個目標群組權重為介於 0 到 999 之間的值。符合加權目標群組之監聽程式規則的請求，會根據其權重分配到這些目標群組。例如，如果您指定兩個目標群組，每個目標群組的權重為 10，則每個目標群組都會收到一半的請求。如果您指定兩個目標群組，一個權重為 10，另一個權重為 20，則權重為 20 的目標群組接收的請求數量是另一個目標群組的兩倍。

根據預設，在加權的目標群組之間分配流量設定規則，並不保證可以接受黏性工作階段。若要確保接受黏性工作階段，請啟用目標群組的黏性規則。當負載平衡器首次將要求路由至加權目標群組時，會產生一個名為的 Cookie，AWSALBTG 該 Cookie 會對所選目標群組的相關資訊進行編碼、加密 Cookie，並在對用戶端的回應中包含 Cookie。用戶端應該包含負載平衡器後續請求中接收的 cookie。當負載平衡器收到符合已啟用目標群組粘性的規則的請求並包含 cookie 時，會將請求路由至 cookie 中指定的目標群組。

Application Load Balancer 不支援 URL 編碼的 Cookie 值。

透過 CORS (跨來源資源共享) 請求，有些瀏覽器需要 SameSite=None; Secure 來啟用綁定。在此情況下，Elastic Load Balancing 會產生第二個 Cookie AWSALBTGCORS，其中包含與原始黏性 Cookie 以及此SameSite屬性相同的資訊。用戶端會同時收到這兩個 Cookie。

Example 具有一個目標群組的轉送動作範例

您可以在建立或修改規則時指定動作。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。下列動作將請求轉送到指定的目標群組。

```
[
  {
    "Type": "forward",
    "ForwardConfig": {
      "TargetGroups": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067"
        }
      ]
    }
  }
]
```

Example 具有兩個加權目標群組的轉送動作範例

下列動作會根據每個目標群組的權重，將要求轉送至兩個指定的目標群組。

```
[
  {
    "Type": "forward",
    "ForwardConfig": {
      "TargetGroups": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/blue-targets/73e2d6bc24d8a067",
          "Weight": 10
        },
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/green-targets/09966783158cda59",
          "Weight": 20
        }
      ]
    }
  }
]
```

```

    }
  ]
}
]

```

Example 已啟用粘性的轉送動作範例

如果您有一個轉送動作涉及多個目標群組，且一個或多個目標群組已啟用[粘性會話](#)，則您必須啟用目標群組粘性。

下列動作會將請求轉送至兩個指定的目標群組，搭配啟用目標群組黏性。不包含該綁定 Cookie 的請求會根據每個目標群組的權重進行路由。

```

[
  {
    "Type": "forward",
    "ForwardConfig": {
      "TargetGroups": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/blue-targets/73e2d6bc24d8a067",
          "Weight": 10
        },
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/green-targets/09966783158cda59",
          "Weight": 20
        }
      ],
      "TargetGroupStickinessConfig": {
        "Enabled": true,
        "DurationSeconds": 1000
      }
    }
  }
]

```

重新導向動作

您可以使用 `redirect` 動作將用戶端請求從一個 URL 重新導向到另一個。您可以根據您的需求，將重新導向設定為暫時 (HTTP 302) 或永久 (HTTP 301)。

URI 包含以下元件：

```
protocol://hostname:port/path?query
```

您必須修改以下至少一個元件，以避免重新導向迴圈：protocol、hostname、port 或 path。未修改的任何元件會維持其原始值。

protocol

通訊協定 (HTTP 或 HTTPS)。您可以重新導向 HTTP 到 HTTP、HTTP 到 HTTPS，以及 HTTPS 到 HTTPS。您不能重新導向 HTTPS 到 HTTP。

hostname

主機名稱。主機名稱不區分大小寫、長度最多 128 個字元，由英數字元、萬用字元 (* 和 ?) 和連字號 (-) 組成。

port

連接埠 (1 到 65535)。

路徑

絕對路徑，開頭為前置字元 "/"。路徑區分大小寫、長度最多 128 個字元，由英數字元、萬用字元 (* 和 ?)、& (使用 &#x26;) 和下列特殊字元組成：_-.\$/~"@:+

query

查詢參數。長度上限為 128 個字元。

您可以使用以下預留關鍵字，來在目標 URL 中重複使用原始 URL 的 URI 元件：

- `{protocol}` - 保留通訊協定。用於通訊協定和查詢元件。
- `{host}` - 保留網域。用於主機名稱、路徑和查詢元件。
- `{port}` - 保留連接埠。用於連接埠、路徑和查詢元件。
- `{path}` - 保留路徑。用於路徑和查詢元件。
- `{query}` - 保留查詢參數。用於查詢元件。

採取 `redirect` 動作時，會將動作記錄於存取日誌中。如需詳細資訊，請參閱 [存取日誌項目](#)。會在 `HTTP_Redirect_Count` 指標中報告成功的 `redirect` 動作計數。如需詳細資訊，請參閱 [Application Load Balancer 指標](#)。

Example 使用主控台的重新導向動作範例

以下規則會設定使用 HTTPS 通訊協定和指定連接埠 (40443) 永久重新導向到 URL，但會保留原始主機名稱、路徑和查詢參數。這個畫面等同於 "https://{host}:40443/{path}?{query}"。

Action types

Forward to target groups Redirect to URL Return fixed response

Redirect to URL [Info](#)

Redirect client requests from one URL to another. You cannot redirect HTTPS to HTTP. To avoid a redirect loop, you must modify at least one of the following components: protocol, port, hostname or path. Components that you do not modify retain their original values.

URI parts | Full URL

Protocol : Port
To retain the original port enter #{port}.

HTTPS ▼ 40443
1-65535

Custom host, path, query
Select to modify host, path and query. If no changes are made, settings from the request URL are retained.

Status code
301 - Permanently moved ▼

以下規則會設定永久重新導向到 URL，保留通訊協定、連接埠、主機名稱和查詢參數，並使用 `{path}` 關鍵字來建立修改的路徑。這個畫面等同於 "`{protocol}://{host}:{port}/new/{path}?{query}`"。

Action types Forward to target groups Redirect to URL Return fixed response**Redirect to URL** | [Info](#)

Redirect client requests from one URL to another. You cannot redirect HTTPS to HTTP. To avoid a redirect loop, you must modify at least one of the following components: protocol, port, hostname or path. Components that you do not modify retain their original values.

URI parts**Full URL****Protocol : Port**

To retain the original port enter #{port}.

#{protocol} ▼

#{port}

1-65535

 Custom host, path, query

Select to modify host, path and query. If no changes are made, settings from the request URL are retained.

Host

Specify a host or retain the original host by using #{host}. Not case sensitive.

#{host}

Maximum 128 characters. Allowed characters are a-z, A-Z, 0-9; the following special characters: -, and wildcards (* and ?). At least one "." is required. Only alphabetical characters are allowed after the final "." character.

Path

Specify a path or retain the original path by using #{path}. Case sensitive.

/new/#{path}

Maximum 128 characters. Allowed characters are a-z, A-Z, 0-9; the following special characters: _-.\$/~'"@:~; & (using &); and wildcards (* and ?).

Query - optional

Specify a query or retain the original query by using #{query}. Not case sensitive.

#{query}

Maximum 128 characters.

Status code

301 - Permanently moved ▼

Example 範例重新導向動作 AWS CLI

您可以在建立或修改規則時指定動作。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。下列動作將 HTTP 請求重新導向到連接埠 443 的 HTTPS 請求，而且使用與 HTTP 請求相同的主機名稱、路徑和查詢字串。

```
[
  {
    "Type": "redirect",
    "RedirectConfig": {
      "Protocol": "HTTPS",
      "Port": "443",
      "Host": "#{host}",
      "Path": "/#{path}",
      "Query": "#{query}",
      "StatusCode": "HTTP_301"
    }
  }
]
```

規則條件類型

以下是規則支援的條件類型：

host-header

根據每個請求的主機名稱來路由傳送。如需詳細資訊，請參閱 [主機條件](#)。

http-header

根據每個請求的 HTTP 標頭來路由傳送。如需詳細資訊，請參閱 [HTTP 標頭條件](#)。

http-request-method

根據每個請求的 HTTP 請求方法來路由傳送。如需詳細資訊，請參閱 [HTTP 請求方法條件](#)。

path-pattern

根據請求 URL 中的路徑模式來路由傳送。如需詳細資訊，請參閱 [路徑條件](#)。

query-string

根據查詢字串中的鍵值組或值來路由傳送。如需詳細資訊，請參閱 [查詢字串條件](#)。

source-ip

根據每個請求的來源 IP 位址來路由傳送。如需詳細資訊，請參閱 [來源 IP 地址條件](#)。

每個規則可以選擇性地包含下列每個條件中的一個：host-header、http-request-method、path-pattern 和 source-ip。每個規則也可以選擇性地包含下列每個條件中的一或多個：http-header 和 query-string。

每個條件最多可以指定三個比對評估。例如，對於每個 `http-header` 條件，您最多可以指定三個字串，以便與請求中的 HTTP 標頭值做比較。如果其中一個字串符合 HTTP 標頭的值，即符合條件。若要求所有字串都要符合，請為每個比對評估建立一個條件。

每個規則最多可以指定五個比對評估。例如，您可以建立含有五個條件的規則，其中每個條件有一個比對評估。

您可以在 `http-header`、`host-header`、`path-pattern` 和 `query-string` 條件的比對評估中包含萬用字元。每個規則以五個萬用字元為限。

規則僅會套用至可見的 ASCII 字元；會排除控制字元 (0x00 到 0x1f 和 0x7f)。

如需示範，請參閱 [Advanced Request Routing](#)。

HTTP 標頭條件

您可以使用 HTTP 標頭條件來設定規則，以根據請求的 HTTP 標頭來路由傳送請求。您可以指定標準或自訂 HTTP 標頭欄位的名稱。標頭名稱和比對評估不區分大小寫。比較字串中支援下列萬用字元：`*` (符合 0 個或多個字元) 和 `?` (確切符合 1 個字元)。標頭名稱中不支援萬用字元。

Example 使用的範例 HTTP 標頭條件 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。當請求的 `User-Agent` 標頭符合其中一個指定的字串時，即滿足下列條件。

```
[
  {
    "Field": "http-header",
    "HTTPHeaderConfig": {
      "HTTPHeaderName": "User-Agent",
      "Values": ["*Chrome*", "*Safari*"]
    }
  }
]
```

HTTP 請求方法條件

您可以使用 HTTP 請求方法條件來設定規則，以根據請求的 HTTP 請求方法來路由傳送請求。您可以指定標準或自訂 HTTP 方法。比對評估區分大小寫。不支援萬用字元；因此，方法名稱必須完全相符。

建議您以相同方式來路由傳送 GET 和 HEAD 請求，因為可快取對 HEAD 請求的回應。

Example 範例 HTTP 方法條件 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。使用指定方法的請求符合下列條件。

```
[
  {
    "Field": "http-request-method",
    "HttpRequestMethodConfig": {
      "Values": ["CUSTOM-METHOD"]
    }
  }
]
```

主機條件

您可以使用主機條件來定義規則，以根據主機標頭中的主機名稱來路由傳送請求 (也稱為以主機為基礎的路由)。這可讓您使用單一負載平衡器來支援多個子網域和不同的頂層網域。

主機名稱不區分大小寫，長度最多可達 128 個字元，而且可以包含下列任何字元：

- A-Z、a-z、0-9
- -
- * (符合 0 個或多個字元)
- ? (確切符合 1 個字元)

您必須至少包含一個 "." 字元。您只可以在最後的 "." 字元之後包含字母字元。

主機名稱範例

- **example.com**
- **test.example.com**
- ***.example.com**

規則 ***.example.com** 會符合 **test.example.com** 但不會符合 **example.com**。

Example 範例主機標頭條件 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。當請求的主機標頭符合指定的字串時，即滿足下列條件。

```
[
  {
    "Field": "host-header",
    "HostHeaderConfig": {
      "Values": ["*.example.com"]
    }
  }
]
```

路徑條件

您可以使用路徑條件來定義規則，以根據請求中的 URL 來路由傳送請求 (也稱為以路徑為基礎的路由)。

系統只會將路徑模式套用到 URL 的路徑，而不會套用到其查詢參數。它僅適用於可見的 ASCII 字符；會排除控制字符 (0x00 到 0x1f 和 0x7f)。

規則評估只會在 URI 規範化發生之後執行。

名稱模式區分大小寫，長度最多可達 128 個字元，而且可以包含下列任何字元。

- A-Z、a-z、0-9
- _ - . \$ / ~ ' " @ : +
- & (使用 &)
- * (符合 0 個或多個字元)
- ? (確切符合 1 個字元)

如果通訊協定版本為 gRPC，則條件可以具體到套件、服務或方法。

HTTP 路徑模式範例

- /img/*
- /img/*/pics

gRPC 路徑模式範例

- /package
- /package.service
- /package.service/method

路徑模式是用於路由請求，但不會修改請求。例如，如果規則具有 /img/* 的路徑模式，則該規則會將 /img/picture.jpg 的請求轉送到指定的目標群組，作為對 /img/picture.jpg 的請求。

Example 範例路徑樣式條件 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。當請求的 URL 包含指定的字串時，即滿足下列條件。

```
[
  {
    "Field": "path-pattern",
    "PathPatternConfig": {
      "Values": ["/img/*"]
    }
  }
]
```

查詢字串條件

您可以使用查詢字串條件來設定規則，以根據查詢字串中的鍵值組或值來路由傳送請求。比對評估不區分大小寫。支援下列萬用字元：* (符合 0 個或多個字元) 和 ? (確切符合 1 個字元)。

Example 查詢字串條件範例 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。當請求的查詢字串包含鍵值組 "version=v1" 或任何設為 "example" 的索引鍵時，即滿足下列條件。

```
[
  {
    "Field": "query-string",
    "QueryStringConfig": {
      "Values": [
        {
          "Key": "version",
          "Value": "v1"
        }
      ]
    }
  }
]
```

```
    },
    {
      "Value": "*example*"
    }
  ]
}
]
```

來源 IP 地址條件

您可以使用來源 IP 地址條件來設定規則，以根據請求的來源 IP 地址來路由傳送請求。必須以 CIDR 格式指定 IP 地址。IPv4 和 IPv6 地址都可用。不支援萬用字元。您無法指定來源 IP 規則條件的 255.255.255.255/32 CIDR。

如果用戶端位在 proxy 後方，則此為 proxy 的 IP 地址，而不是用戶端的 IP 地址。

X-Forwarded-For 標頭中的地址不滿足此條件。若要搜尋 X-Forwarded-For 標頭中的地址，請使用 `http-header` 條件。

Example 範例來源 IP 條件 AWS CLI

您可以在建立或修改規則時指定條件。如需詳細資訊，請參閱 [create-rule](#) 和 [modify-rule](#) 命令。當請求的來源 IP 地址出現在其中一個指定的 CIDR 區塊時，即滿足下列條件。

```
[
  {
    "Field": "source-ip",
    "SourceIpConfig": {
      "Values": ["192.0.2.0/24", "198.51.100.10/32"]
    }
  }
]
```

為 Application Load Balancer 建立 HTTP 接聽程式

接聽程式會檢查連線請求。當您在立負載平衡器時便定義接聽程式，然後可隨時新增接聽程式到您的負載平衡器。

此頁面的資訊協助您為負載平衡器建立 HTTP 接聽程式。若要將 HTTPS 接聽程式新增至您的負載平衡器，請參閱 [為 Application Load Balancer 建立 HTTPS 接聽程式](#)。

必要條件

- 若要新增轉送動作到預設的接聽程式規則，您必須指定可用的目標群組。如需詳細資訊，請參閱 [建立目標群組](#)。
- 您可以在多個接聽程式中指定相同的目標群組，但這些接聽程式必須屬於相同的負載平衡器。若要將目標群組與負載平衡器搭配使用，您必須確認沒有其他負載平衡器的接聽程式使用該目標群組。

新增 HTTP 接聽程式

您使用用戶端與負載平衡器間連線的通訊協定與連接埠來設定接聽程式，並為預設接聽程式規則設定目標群組。如需詳細資訊，請參閱 [接聽程式組態](#)。

使用主控台新增 HTTP 接聽程式

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選擇新增接聽程式。
5. 在通訊協定：連接埠中，選擇 HTTP，並保留預設連接埠或輸入其他連接埠。
6. 在預設動作中，選擇下列其中一項：
 - 轉送至目標群組 – 選擇一個或多個要將流量轉送至其中的目標群組。若要新增目標群組，請選擇新增目標群組。如果使用多個目標群組，請為每個目標群組選取權重，並檢閱相關的百分比。如果您已在一個或多個目標群組上啟用粘性，則必須在規則上啟用群組層級粘性。
 - 重新導向至 URL – 指定將用戶端請求重新導向所至的 URL。這可以透過在 URI 部分索引標籤上單獨輸入每個部分，或在完整 URL 索引標籤上輸入完整地址來完成。若是狀態碼，您可以根據需求，將重新導向設定為暫時 (HTTP 302) 或永久 (HTTP 301)。
 - 傳回固定回應 – 指定回應代碼，此代碼將傳回至遭捨棄的用戶端請求。此外，您可以指定內容類型和回應內文，但這並非必填的資訊。
7. 選擇新增。

若要使用 AWS CLI

使用 [create-listener](#) 命令來建立接聽程式和預設規則，以及 [create-rule](#) 命令來定義額外的接聽程式規則。

為 Application Load Balancer 建立 HTTPS 接聽程式

接聽程式會檢查連線請求。當您在立負載平衡器時便定義接聽程式，然後可隨時新增接聽程式到您的負載平衡器。

若要建立 HTTPS 接聽程式，您必須在負載平衡器上部署至少一個 SSL 伺服器憑證。負載平衡器使用伺服器憑證終止前端連接，然後解密用戶端的請求，再將它們傳送到目標。您還必須指定安全政策，此政策在用戶端和負載平衡器之間交涉安全連線時會用到。

如果您需要將加密流量傳遞給目標，而不需要負載平衡器解密流量，就可以建立在連接埠 443 上具有 TCP 接聽程式的 Network Load Balancer 或 Classic Load Balancer。使用 TCP 接聽程式時，負載平衡器會將加密的流量傳遞給目標，而不需要對流量進行解密。

Application Load Balancer 不支援 ED25519 金鑰。

此頁面的資訊協助您為負載平衡器建立 HTTPS 接聽程式。若要將 HTTP 接聽程式新增至您的負載平衡器，請參閱[為 Application Load Balancer 建立 HTTP 接聽程式](#)。

內容

- [SSL 憑證](#)
 - [預設憑證](#)
 - [憑證清單](#)
 - [憑證續約](#)
- [安全政策](#)
 - [TLS 1.3 安全政策](#)
 - [FIPS 安全性原則](#)
 - [FS 支援的政策](#)
 - [TLS 1.0-1.2 安全性原則](#)
 - [TLS 通訊協定和密碼](#)
- [新增 HTTPS 接聽程式](#)

SSL 憑證

負載平衡器需要 X.509 憑證 (SSL/TLS 伺服器憑證)。憑證為憑證授權機構 (CA) 發出的數位形式身分證明。憑證包含識別資訊、有效期間、公有金鑰、序號和發行者的數位簽章。

建立憑證以搭配您的負載平衡器使用時，您必須指定網域名稱。憑證上的網域名稱必須與自訂網域名稱記錄相符，如此我們就可以確認 TLS 連線。如果其不相符，就不會加密流量。

您必須為憑證指定完整網域名稱 (FQDN)，例如 `www.example.com`；或者指定 apex 網域名稱 (FQDN)，例如 `example.com`。您也可以使用星號 (*) 做為萬用字元，以保護相同網域中的多個網站名稱。請求萬用字元憑證時，星號 (*) 必須在網域名稱的最左方，而且僅能保護一個子網域層級。例如，`*.example.com` 保護 `corp.example.com` 和 `images.example.com`，但它無法保護 `test.login.example.com`。另請注意，`*.example.com` 只可以保護 `example.com` 的子網域，無法保護 bare 或 apex 網域 (`example.com`)。萬用字元名稱會顯示於憑證的主體欄位和主體別名延伸。如需公有憑證的詳細資訊，請參閱 [AWS Certificate Manager 使用者指南](#) 中的 [請求公有憑證](#)。

建議您使用 [AWS Certificate Manager \(ACM\)](#) 為負載平衡器建立憑證。ACM 支援具有 2048、3072 和 4096 位元金鑰長度的 RSA 憑證，以及所有 ECDSA 憑證。ACM 會與 Elastic Load Balancing 整合，以便您在負載平衡器上部署憑證。如需詳細資訊，請參閱 [AWS Certificate Manager 使用者指南](#)。

或者，您可以使用 SSL/TLS 工具建立憑證簽署要求 (CSR)，然後取得 CA 簽署的 CSR 以產生憑證，然後將憑證匯入 ACM 或將憑證上傳至 AWS Identity and Access Management (IAM)。如需有關將憑證匯入 ACM 的詳細資訊，請參閱《AWS Certificate Manager 使用者指南》中的 [匯入憑證](#)。如需上傳憑證至 IAM 的詳細資訊，請參閱 IAM 使用者指南中的 [使用伺服器憑證](#)。

預設憑證

建立 HTTPS 接聽程式時，您必須指定剛好一個憑證。此憑證稱為預設憑證。您可以在建立 HTTPS 接聽程式之後取代預設憑證。如需詳細資訊，請參閱 [更換預設憑證](#)。

如果您在 [憑證清單](#) 中指定額外憑證，只有當用戶端連接時未使用伺服器名稱指示 (SNI) 通訊協定來指定主機名稱，或憑證清單中沒有相符的憑證時，才會使用預設憑證。

如果您不指定額外憑證，但需要透過單一負載平衡器來託管多個安全應用程式，您可以使用萬用字元憑證，或將每個額外網域的主體別名 (SAN) 新增至憑證。

憑證清單

HTTPS 接聽程式建立之後具有預設憑證和空的憑證清單。您可以選擇性將憑證新增至接聽程式的憑證清單。使用憑證清單可讓負載平衡器在相同連接埠上支援多個網域，並為每個網域提供不同的憑證。如需詳細資訊，請參閱 [將憑證新增至憑證清單](#)。

負載平衡器使用支援 SNI 的智慧憑證選擇演算法。如果用戶端提供的主機名稱符合憑證清單中的單一憑證，負載平衡器會選取此憑證。如果用戶端提供的主機名稱符合憑證清單中的多個憑證，負載平衡器會選取用戶端可支援的最佳憑證。憑證選擇是根據採用下列順序的以下條件：

- 公有金鑰演算法 (ECDSA 優於 RSA)
- 雜湊演算法 (SHA 優於 MD5)
- 金鑰長度 (最好是最大)
- 有效期間

負載平衡器存取日誌項目會指出用戶端指定的主機名稱和向用戶端出示的憑證。如需詳細資訊，請參閱[存取日誌項目](#)。

憑證續約

每個憑證均附帶有效期間。您必須確保在有效期間結束之前，續約或更換負載平衡器的每個憑證。這包括預設憑證和憑證清單中的憑證。續約或更換憑證不會影響負載平衡器節點收到並且等待路由到運作狀態良好目標的傳輸中請求。續約憑證之後，新請求會使用續約的憑證。更換憑證之後，新請求會使用新的憑證。

您可以如下所示管理憑證續約和更換：

- 負載平衡器提供 AWS Certificate Manager 並部署在負載平衡器上的憑證可以自動續約。ACM 會在憑證過期之前嘗試續約。如需詳細資訊，請參閱 AWS Certificate Manager 使用者指南中的[受管續約](#)。
- 如果您將憑證匯入至 ACM，則必須監控憑證的過期日期，並在憑證過期之前續約。如需詳細資訊，請參閱 AWS Certificate Manager 使用者指南中的[匯入憑證](#)。
- 如果您將憑證匯入至 IAM，則必須建立新的憑證、將新的憑證匯入至 ACM 或 IAM、將新憑證新增至負載平衡器，並從負載平衡器移除過期的憑證。

安全政策

Elastic Load Balancing 使用 Secure Sockets Layer (SSL) 交涉組態 (稱為安全政策)，在用戶端與負載平衡器之間交涉 SSL 連線。安全政策為通訊協定與加密的組合。通訊協定會在用戶端與伺服器之間建立安全連線，並確保在用戶端與負載平衡器之間傳遞的所有資料為私有。隨碼是一項加密演算法，使用加密金鑰來建立編碼的訊息。通訊協定使用多個加密來加密透過網際網路的資料。在連線交涉程序期間，用戶端與負載平衡器會出示它們分別支援的加密和通訊協定的清單 (以偏好的順序)。在預設情況下，將針對安全連線選取伺服器清單上符合任何用戶端加密的第一個加密。

考量：

- Application Load Balancers 僅支援目標連線的 SSL 重新交涉。

- Application Load Balancers 不支援自訂安全政策。
- 此原ELBSecurityPolicy-TLS13-1-2-2021-06則是使用建立的 HTTPS 接聽程式的預設安全性原則 AWS Management Console。
- 此原ELBSecurityPolicy-2016-08則是使用建立的 HTTPS 接聽程式的預設安全性原則 AWS CLI。
- 建立 HTTPS 接聽程式時，需要選取安全性原則。
 - 我們建議使用ELBSecurityPolicy-TLS13-1-2-2021-06安全性原則，其中包含 TLS 1.3，並且向後相容於 TLS 1.2。
- 您可以選擇用於前端連線的安全性原則，但不能選擇後端連線。
 - 對於後端連線，如果 HTTPS 接聽程式使用的是 TLS 1.3 安全政策，則會使用 ELBSecurityPolicy-TLS13-1-0-2021-06 安全政策。否則會將 ELBSecurityPolicy-2016-08 安全政策用於後端連線。
- 若要符合需要停用特定 TLS 通訊協定版本的合規性和安全性標準，或是支援需要取代加密的舊版用戶端，您可以使用其中一個ELBSecurityPolicy-TLS-安全性原則。若要檢視對應用 Application Load Balancer 衡器要求的 TLS 通訊協定版本，請啟用負載平衡器的存取記錄，並檢查對應的存取記錄項目。如需詳細資訊，請參閱 [Application Load Balancer 的存取記錄](#)。
- 您可以分別使用 IAM AWS 帳戶 和 AWS Organizations 服務控制政策 (SCP) 中的 [Elastic Load Balancing 條件金鑰](#)，來限制您的使用者可以使用哪些安全政策。如需詳細資訊，請參閱AWS Organizations 使用指南中的 [服務控制原則 \(SCP\)](#)

TLS 1.3 安全政策

Elastic Load Balancing 為應用程式負載平衡器提供下列 TLS 1.3 安全性原則：

- ELBSecurityPolicy-TLS13-1-2-2021-06(推薦)
- ELBSecurityPolicy-TLS13-1-2-Res-2021-06
- ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06
- ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06
- ELBSecurityPolicy-TLS13-1-1-2021-06
- ELBSecurityPolicy-TLS13-1-0-2021-06
- ELBSecurityPolicy-TLS13-1-3-2021-06

FIPS 安全性原則

Important

連結至「Application Load Balancer」的所有安全接聽程式都必須使用 FIPS 安全性原則或非 FIPS 安全性原則；它們不能混合使用。如果現有的 Application Load Balancer 有兩個或多個使用非 FIP 原則的接聽程式，而且您希望接聽程式改用 FIPS 安全性原則，請移除所有接聽程式，直到只有一個偵聽器為止。將監聽器的安全性原則變更為 FIPS，然後使用 FIPS 安全性原則建立其他接聽程式。或者，您也可以僅使用 FIPS 安全性原則，建立含有新接聽程式的新應用程式負載平衡器。

聯邦資訊處理標準 (FIPS) 是美國和加拿大政府的一項標準，針對保護敏感資訊的加密模組指定安全性要求。若要深入了解，請參閱AWS 雲端安全性合規頁面上的[聯邦資訊處理標準 \(FIPS\) 140](#)。

所有 FIPS 原則均利用經過 AWS-LC FIPS 驗證的密碼編譯模組。若要深入了解，請參閱 NIST 密碼編譯模組驗證程式網站上的 AWS-LC 密碼編譯[模組](#)頁面。

Elastic Load Balancing 為應用程式負載平衡器提供下列 FIPS 安全性原則：

- ELBSecurityPolicy-TLS13-1-3-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-2-Res-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04(推薦)
- ELBSecurityPolicy-TLS13-1-2-Ext0-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-2-Ext1-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-2-Ext2-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-1-FIPS-2023-04
- ELBSecurityPolicy-TLS13-1-0-FIPS-2023-04

FS 支援的政策

Elastic Load Balancing 為應用程式負載平衡器提供下列 FS (正向保密) 支援的安全性原則：

- ELBSecurityPolicy-FS-1-2-Res-2020-10
- ELBSecurityPolicy-FS-1-2-Res-2019-08
- ELBSecurityPolicy-FS-1-2-2019-08

- ELBSecurityPolicy-FS-1-1-2019-08
- ELBSecurityPolicy-FS-2018-06

TLS 1.0-1.2 安全性原則

Elastic Load Balancing 為應用程式負載平衡器提供下列 TLS 1.0-1.2 安全性原則：

- ELBSecurityPolicy-TLS-1-2-Ext-2018-06
- ELBSecurityPolicy-TLS-1-2-2017-01
- ELBSecurityPolicy-TLS-1-1-2017-01
- ELBSecurityPolicy-2016-08
- ELBSecurityPolicy-TLS-1-0-2015-04
- ELBSecurityPolicy-2015-05(等同於 **ELBSecurityPolicy-2016-08**)

TLS 通訊協定和密碼

TLS 1.3

下表說明可用 TLS 1.3 安全性原則所支援的 TLS 通訊協定和密碼。

注意：前ELBSecurityPolicy-置碼已從安全性原則列的原則名稱中移除。

範例：安全性原ELBSecurityPolicy-TLS13-1-2-2021-06則會顯示為TLS13-1-2-2021-06。

安全政策	TLS13-1-2-2021-06	TLS13-1-3-2021-06	TLS13-1-2-Res-2021-06	TLS13-1-2-Ext2-2021-06	TLS13-1-2-Ext1-2021-06	TLS13-1-1-2021-06	TLS13-1-0-2021-06
TLS 通訊協定							
Protocol-TLSv1							✓

安全政策	TLS13-1-2-2021-06	TLS13-1-3-2021-06	TLS13-1-2-Res-2021-06	TLS13-1-2-Ext2-2021-06	TLS13-1-2-Ext1-2021-06	TLS13-1-1-2021-06	TLS13-1-0-2021-06
Protocol-TLSv1.1						✓	✓
Protocol-TLSv1.2	✓		✓	✓	✓	✓	✓
通訊協定-TLS1.3	✓	✓	✓	✓	✓	✓	✓
TLS 加密							
TLS_AES_128_GCM_SHA256	✓	✓	✓	✓	✓	✓	✓
TLS_AES_256_GCM_SHA384	✓	✓	✓	✓	✓	✓	✓
TLS_CHACHA20_POLY1305_SHA256	✓	✓	✓	✓	✓	✓	✓
ECDHE-ECDSA-AES128-GCM-SHA256	✓		✓	✓	✓	✓	✓

安全政策	TLS13-1-2-2021-06	TLS13-1-3-2021-06	TLS13-1-2-Res-2021-06	TLS13-1-2-Ext2-2021-06	TLS13-1-2-Ext1-2021-06	TLS13-1-1-2021-06	TLS13-1-0-2021-06
ECDHE- RSA- AES128- GCM- SHA256	✓		✓	✓	✓	✓	✓
ECDHE- ECDSA- AES128- SHA256	✓			✓	✓	✓	✓
ECDHE- RSA- AES128- SHA256	✓			✓	✓	✓	✓
ECDHE- ECDSA- AES128- SHA				✓		✓	✓
ECDHE- RSA- AES128- SHA				✓		✓	✓
ECDHE- ECDSA- AES256 -GCM- SHA384	✓		✓	✓	✓	✓	✓

安全政策	TLS13-1-2-2021-06	TLS13-1-3-2021-06	TLS13-1-2-Res-2021-06	TLS13-1-2-Ext2-2021-06	TLS13-1-2-Ext1-2021-06	TLS13-1-1-2021-06	TLS13-1-0-2021-06
ECDHE- RSA- AES256- GCM- SHA384	✓		✓	✓	✓	✓	✓
ECDHE- ECDSA- AES256- SHA384	✓			✓	✓	✓	✓
ECDHE- RSA- AES256- SHA384	✓			✓	✓	✓	✓
ECDHE- RSA- AES256- SHA				✓		✓	✓
ECDHE- ECDSA- AES256- SHA				✓		✓	✓
AES128- GCM- SHA256				✓	✓	✓	✓

安全政策	TLS13-1-2-2021-06	TLS13-1-3-2021-06	TLS13-1-2-Res-2021-06	TLS13-1-2-Ext2-2021-06	TLS13-1-2-Ext1-2021-06	TLS13-1-1-2021-06	TLS13-1-0-2021-06
AES128- SHA256				✓	✓	✓	✓
AES128- SHA				✓		✓	✓
AES256- GCM- SHA384				✓	✓	✓	✓
AES256- SHA256				✓	✓	✓	✓
AES256- SHA				✓		✓	✓

若要使用 CLI 建立使用 TLS 1.3 原則的 HTTPS 接聽程式

使用 [建立接聽程式](#) 命令搭配任何 [TLS 1.3 安全性](#) 原則。

此範例使用安 ELBSecurityPolicy-TLS13-1-2-2021-06 全性原則。

```
aws elbv2 create-listener --name my-listener \
--protocol HTTPS --port 443 \
--ssl-policy ELBSecurityPolicy-TLS13-1-2-2021-06
```

若要使用 CLI 修改 HTTPS 接聽程式以使用 TLS 1.3 原則

搭配任何 [TLS 1.3 安全性](#) 原則使用 [修改接聽程式](#) 命令。

此範例使用安 ELBSecurityPolicy-TLS13-1-2-2021-06 全性原則。

```
aws elbv2 modify-listener \
```

```
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0 \  
--ssl-policy ELBSecurityPolicy-TLS13-1-2-2021-06
```

檢視監聽器使用 CLI 所使用的安全原則

使用 [描述偵聽程式命令搭配您arn的監聽程式](#)。

```
aws elbv2 describe-listeners \  
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0
```

若要使用 CLI 檢視 TLS 1.3 安全性原則的組態

使用 [描述 SSL 原則命令搭配任何 TLS 1.3 安全性原則](#)。

此範例使用安ELBSecurityPolicy-TLS13-1-2-2021-06全性原則。

```
aws elbv2 describe-ssl-policies \  
--names ELBSecurityPolicy-TLS13-1-2-2021-06
```

FIPS

Important

原則ELBSecurityPolicy-TLS13-1-1-FIPS-2023-04和ELBSecurityPolicy-TLS13-1-0-FIPS-2023-04僅針對舊版相容性而提供。雖然他們使用使用 FIPS140 模組的 FIPS 密碼編譯，但它們可能不符合 TLS 組態的最新 NIST 指南。

下表說明可用 FIPS 安全性原則所支援的 TLS 通訊協定和密碼。

注意：前ELBSecurityPolicy-置碼已從安全性原則列的原則名稱中移除。

範例：安全性原ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04則會顯示為TLS13-1-2-FIPS-2023-04。

安全政策

TLS13-1-3-FIPS-2023-04

TLS13-1-2-Res-FIPS-2023-04

TLS13-1-2-FIPS-2023-04

TLS13-1-2-Ext0-FIPS-2023-04

TLS13-1-2-Ext1-FIPS-2023-04

TLS13-1-2-Ext2-FIPS-2023-04

TLS13-1-1-FIPS-2023-04

TLS13-1-0-FIPS-2023-04

TLS 通訊協定

Protocol-TLSv1

✓

Protocol-TLSv1.1

✓

✓

Protocol-TLSv1.2

✓

✓

✓

✓

✓

✓

✓

通訊協定-TLS1.3

✓

✓

✓

✓

✓

✓

✓

✓

TLS 加密

TLS_AES_128_GCM_SHA256

✓

✓

✓

✓

✓

✓

✓

TLS_AES_256_GCM_SHA384

✓

✓

✓

✓

✓

✓

✓

ECDHE-ECDSA-AES128-GCM-

✓

✓

✓

✓

✓

✓

✓

安全政
策

TLS13-1-3-FIPS-2023-04

TLS13-1-2-Res-FIPS-2023-04

TLS13-1-2-FIPS-2023-04

TLS13-1-2-Ext0-FIPS-2023-04

TLS13-1-2-Ext1-FIPS-2023-04

TLS13-1-2-Ext2-FIPS-2023-04

TLS13-1-1-FIPS-2023-04

TLS13-1-0-FIPS-2023-04

SHA2
56ECDHE-
RSA-
AES128-
GCM-
SHA256

✓

✓

✓

✓

✓

✓

✓

ECDHE-
ECD
SA-
AES128
-
SHA256

✓

✓

✓

✓

✓

✓

ECDHE-
RSA-
AES128-
S
HA256

✓

✓

✓

✓

✓

✓

ECDHE-
ECD
SA-
AES128
-SHA

✓

✓

✓

✓

安全政
策

	TLS13-1-3-FIPS-2023-04	TLS13-1-2-Res-FIPS-2023-04	TLS13-1-2-FIPS-2023-04	TLS13-1-2-Ext0-FIPS-2023-04	TLS13-1-2-Ext1-FIPS-2023-04	TLS13-1-2-Ext2-FIPS-2023-04	TLS13-1-1-FIPS-2023-04	TLS13-1-0-FIPS-2023-04
ECDHE- RSA- AES128- SHA				✓		✓	✓	✓
ECDHE- ECD SA- AES256 -GCM- SHA3 84	✓	✓	✓	✓	✓	✓	✓	✓
ECDHE- RSA- AES256- GCM- SHA384	✓	✓	✓	✓	✓	✓	✓	✓
ECDHE- ECD SA- AES256 - SHA384			✓	✓	✓	✓	✓	✓

安全政
策

	TLS13-1-3-FIPS-2023-04	TLS13-1-2-Res-FIPS-2023-04	TLS13-1-2-FIPS-2023-04	TLS13-1-2-Ext0-FIPS-2023-04	TLS13-1-2-Ext1-FIPS-2023-04	TLS13-1-2-Ext2-FIPS-2023-04	TLS13-1-1-FIPS-2023-04	TLS13-1-0-FIPS-2023-04
ECDHE- RSA- AES256- S HA384			✓	✓	✓	✓	✓	✓
ECDHE- RSA- AES256- SHA				✓		✓	✓	✓
ECDHE- ECD SA- AES256 -SHA				✓		✓	✓	✓
AES128- GCM- SHA256					✓	✓	✓	✓
AES128- SH A256					✓	✓	✓	✓
AES128- SHA						✓	✓	✓

安全政策	TLS13-1-3-FIPS-2023-04	TLS13-1-2-Res-FIPS-2023-04	TLS13-1-2-FIPS-2023-04	TLS13-1-2-Ext0-FIPS-2023-04	TLS13-1-2-Ext1-FIPS-2023-04	TLS13-1-2-Ext2-FIPS-2023-04	TLS13-1-1-FIPS-2023-04	TLS13-1-0-FIPS-2023-04
AES256-GCM-SHA384					✓	✓	✓	✓
AES256-SHA A256				✓	✓	✓	✓	✓
AES256-SHA						✓	✓	✓

若要使用 CLI 建立使用 FIPS 原則的 HTTPS 接聽程式

使用 [建立接聽程式](#) 命令搭配任何 [FIPS](#) 安全性原則。

此範例使用安 `ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04` 全性原則。

```
aws elbv2 create-listener --name my-listener \
--protocol HTTPS --port 443 \
--ssl-policy ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04
```

若要使用 CLI 修改 HTTPS 接聽程式以使用 FIPS 原則

對任何 [FIPS](#) 安全 [性原則使用修改偵聽程式](#) 命令。

此範例使用安 `ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04` 全性原則。

```
aws elbv2 modify-listener \
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0 \
--ssl-policy ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04
```

檢視監聽器使用 CLI 所使用的安全原則

使用[描述偵聽程式命令搭配您arn的監聽程式](#)。

```
aws elbv2 describe-listeners \
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0
```

使用 CLI 檢視 FIPS 安全性原則的組態

使用[描述 SSL 原則命令搭配任何 FIPS 安全性原則](#)。

此範例使用安ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04全性原則。

```
aws elbv2 describe-ssl-policies \
--names ELBSecurityPolicy-TLS13-1-2-FIPS-2023-04
```

FS

下表說明可用 FS 支援的安全性原則所支援的 TLS 通訊協定和密碼。

注意：前ELBSecurityPolicy-置碼已從安全性原則列的原則名稱中移除。

範例：安全性原ELBSecurityPolicy-FS-2018-06則會顯示為FS-2018-06。

安全政策	Default	FS-1-2-Res-2020-10	FS-1-2-Res-2019-08	FS-1-2-2019-08	FS-1-1-2019-08	FS-2018-06
TLS 通訊協定						
Protocol-TLSv1	✓					✓
Protocol-TLSv1.1	✓				✓	✓

安全政策	Default	FS-1-2-Res-2020-10	FS-1-2-Res-2019-08	FS-1-2-2019-08	FS-1-1-2019-08	FS-2018-06
Protocol-TLSv1.2	✓	✓	✓	✓	✓	✓
TLS 加密						
ECDHE-ECDSA-AES128-GCM-SHA256	✓	✓	✓	✓	✓	✓
ECDHE-RSA-AES128-GCM-SHA256	✓	✓	✓	✓	✓	✓
ECDHE-ECDSA-AES128-SHA256	✓		✓	✓	✓	✓
ECDHE-RSA-AES128-SHA256	✓		✓	✓	✓	✓

安全政策	Default	FS-1-2-Res-2020-10	FS-1-2-Res-2019-08	FS-1-2-2019-08	FS-1-1-2019-08	FS-2018-06
ECDHE- ECDSA- AES128- SHA	✓			✓	✓	✓
ECDHE- RSA- AES128-S HA	✓			✓	✓	✓
ECDHE- ECDSA- AES256 -GCM- SHA384	✓	✓	✓	✓	✓	✓
ECDHE- RSA- AES256- GCM- SHA384	✓	✓	✓	✓	✓	✓
ECDHE- ECDSA- AES256- SHA384	✓		✓	✓	✓	✓

安全政策	Default	FS-1-2-Res-2020-10	FS-1-2-Res-2019-08	FS-1-2-2019-08	FS-1-1-2019-08	FS-2018-06
ECDHE-RSA-AES256-SHA384	✓		✓	✓	✓	✓
ECDHE-RSA-AES256-SHA	✓			✓	✓	✓
ECDHE-ECDSA-AES256-SHA	✓			✓	✓	✓
AES128-GCM-SHA256	✓					
AES128-SHA256	✓					
AES128-SHA	✓					
AES256-GCM-SHA384	✓					

安全政策	Default	FS-1-2-Res-2020-10	FS-1-2-Res-2019-08	FS-1-2-2019-08	FS-1-1-2019-08	FS-2018-06
AES256-SHA256	✓					
AES256-SHA	✓					

若要使用 CLI 建立使用 FS 支援原則的 HTTPS 接聽程式

使用[建立偵聽程式命令](#)搭配任何 [FS 支援的安全性原則](#)。

此範例使用安ELBSecurityPolicy-FS-2018-06全性原則。

```
aws elbv2 create-listener --name my-listener \
--protocol HTTPS --port 443 \
--ssl-policy ELBSecurityPolicy-FS-2018-06
```

若要使用 CLI 修改 HTTPS 接聽程式以使用 FS 支援的原則

使用[修改偵聽程式命令](#)搭配任何 [FS 支援的安全性原則](#)。

此範例使用安ELBSecurityPolicy-FS-2018-06全性原則。

```
aws elbv2 modify-listener \
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0 \
--ssl-policy ELBSecurityPolicy-FS-2018-06
```

檢視監聽器使用 CLI 所使用的安全性原則

使用[描述偵聽程式命令](#)搭配您arn的監聽程式。

```
aws elbv2 describe-listeners \
```

```
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0
```

若要使用 CLI 檢視 FS 支援的安全性原則的組態

[使用描述-ssl 原則命令搭配任何 FS 支援的安全性原則。](#)

此範例使用安ELBSecurityPolicy-FS-2018-06全性原則。

```
aws elbv2 describe-ssl-policies \
--names ELBSecurityPolicy-FS-2018-06
```

TLS 1.0 - 1.2

下表說明可用 TLS 1.0-1.2 安全性原則所支援的 TLS 通訊協定和密碼。

注意：前ELBSecurityPolicy-置碼已從安全性原則列的原則名稱中移除。

範例：安全性原ELBSecurityPolicy-TLS-1-2-Ext-2018-06則會顯示為TLS-1-2-Ext-2018-06。

安全政策	Default	TLS-1-2-Ext-2018-06	TLS-1-2-2017-01	TLS-1-1-2017-01	TLS-1-0-2015-04*
TLS 通訊協定					
Protocol-TLSv1	✓				✓
Protocol-TLSv1.1	✓			✓	✓
Protocol-TLSv1.2	✓	✓	✓	✓	✓

安全政策

Default

TLS-1-2-Ext-2018-06

TLS-1-2-2017-01

TLS-1-1-2017-01

TLS-1-0-2015-04*

TLS 加密

ECDHE-ECD
SA-AES128
-GCM-SHA2
56

✓

✓

✓

✓

✓

ECDHE-RSA ✓
-AES128-G
CM-SHA256

✓

✓

✓

✓

ECDHE-ECD ✓
SA-AES128-
SHA256

✓

✓

✓

✓

ECDHE-RSA ✓
-AES128-S
HA256

✓

✓

✓

✓

ECDHE-ECD ✓
SA-AES128-
SHA

✓

✓

✓

✓

ECDHE-RSA ✓
-AES128-S
HA

✓

✓

✓

✓

安全政策	Default	TLS-1-2-Ext-2018-06	TLS-1-2-2017-01	TLS-1-1-2017-01	TLS-1-0-2015-04*
ECDHE-ECD SA-AES256 -GCM-SHA3 84	✓	✓	✓	✓	✓
ECDHE-RSA -AES256-G CM-SHA384	✓	✓	✓	✓	✓
ECDHE-ECD SA-AES256- SHA384	✓	✓	✓	✓	✓
ECDHE-RSA -AES256-S HA384	✓	✓	✓	✓	✓
ECDHE-RSA -AES256-S HA	✓	✓		✓	✓
ECDHE-ECD SA-AES256- SHA	✓	✓		✓	✓
AES128-GC M-SHA256	✓	✓	✓	✓	✓
AES128-SH A256	✓	✓	✓	✓	✓

安全政策	Default	TLS-1-2-Ext-2018-06	TLS-1-2-2017-01	TLS-1-1-2017-01	TLS-1-0-2015-04*
AES128-SH A	✓	✓		✓	✓
AES256-GC M-SHA384	✓	✓	✓	✓	✓
AES256-SH A256	✓	✓	✓	✓	✓
AES256-SH A	✓	✓		✓	✓
DES-CBC3- SHA					✓

* 請勿使用此政策，除非您必須支援需要 DES-CBC3-SHA 加密（一種弱式加密）的傳統用戶端。

若要使用 CLI 建立使用 TLS 1.0-1.2 原則的 HTTPS 接聽程式

使用 [建立接聽程式](#) 命令搭配任何 [TLS 1.0-1.2 支援](#) 的安全性原則。

此範例使用安 ELBSecurityPolicy-2016-08 全性原則。

```
aws elbv2 create-listener --name my-listener \
--protocol HTTPS --port 443 \
--ssl-policy ELBSecurityPolicy-2016-08
```

若要使用 CLI 修改 HTTPS 接聽程式以使用 TLS 1.0-1.2 原則

將 [修改接聽程式命令](#) 與任何 [TLS 1.0-1.2 支援](#) 的安全性原則搭配使用。

此範例使用安ELBSecurityPolicy-2016-08全性原則。

```
aws elbv2 modify-listener \  
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0 \  
--ssl-policy ELBSecurityPolicy-2016-08
```

檢視監聽器使用 CLI 所使用的安全原則

使用[描述偵聽程式命令搭配您arn的監聽程式](#)。

```
aws elbv2 describe-listeners \  
--listener-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/my-load-balancer/abcdef01234567890/1234567890abcdef0
```

使用 CLI 檢視 TLS 1.0-1.2 安全性原則的組態

使用[描述-ssl 原則命令搭配任何 TLS 1.0-1.2 支援的安全性原則](#)。

此範例使用安ELBSecurityPolicy-2016-08全性原則。

```
aws elbv2 describe-ssl-policies \  
--names ELBSecurityPolicy-2016-08
```

新增 HTTPS 接聽程式

您使用用戶端與負載平衡器間連線的通訊協定與連接埠來設定接聽程式，並為預設接聽程式規則設定目標群組。如需詳細資訊，請參閱 [接聽程式組態](#)。

必要條件

- 若要建立 HTTPS 接聽程式，您必須指定憑證和安全政策。負載平衡器會使用憑證來終止連接，然後解密用來自用戶端的請求，之後才將它們路由到目標。負載平衡器會在與用戶端交涉 SSL 連線時使用安全政策。
- 若要新增轉送動作到預設的接聽程式規則，您必須指定可用的目標群組。如需詳細資訊，請參閱 [建立目標群組](#)。
- 您可以在多個接聽程式中指定相同的目標群組，但這些接聽程式必須屬於相同的負載平衡器。若要將目標群組與負載平衡器搭配使用，您必須確認沒有其他負載平衡器的接聽程式使用該目標群組。

使用主控台新增 HTTPS 接聽程式

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選擇新增接聽程式。
5. 對於通訊協定：連接埠，選擇 HTTPS，並保留預設連接埠或輸入其他連接埠。
6. (選用) 若要啟用身分驗證，請在身分驗證下選取使用 OpenID 或 Amazon Cognito，然後提供請求的資訊。如需詳細資訊，請參閱 [使用 Application Load Balancer 來驗證使用者身分](#)。
7. 針對 Default actions (預設動作)，執行下列其中一項作業：
 - 轉送至目標群組 – 選擇一個或多個要將流量轉送至其中的目標群組。若要新增目標群組，請選擇新增目標群組。如果使用多個目標群組，請為每個目標群組選取權重，並檢閱相關的百分比。如果您已在一個或多個目標群組上啟用粘性，則必須在規則上啟用群組層級粘性。
 - 重新導向至 URL – 指定將用戶端請求重新導向所至的 URL。這可以透過在 URI 部分索引標籤上單獨輸入每個部分，或在完整 URL 索引標籤上輸入完整地址來完成。若是狀態碼，您可以根據需求，將重新導向設定為暫時 (HTTP 302) 或永久 (HTTP 301)。
 - 傳回固定回應 – 指定回應代碼，此代碼將傳回至遭捨棄的用戶端請求。此外，您可以指定內容類型和回應內文，但這並非必填的資訊。
8. 對於安全政策，建議您一律使用最新的預先定義安全政策。
9. 針對預設 SSL/TLS 憑證，有下列選項可用：
 - 如果您使用建立或匯入憑證 AWS Certificate Manager，請選取從 ACM，然後從選取憑證選取憑證選取憑證。
 - 如果您使用 IAM 匯入憑證，請選取從 IAM，然後從選取憑證處選取您的憑證。
 - 如果您想匯入憑證，但您的區域無法使用 ACM，請依序選取匯入和到 IAM。在憑證名稱欄位輸入憑證名稱。在憑證私有金鑰中，複製並貼上私有金鑰檔案的內容 (PEM 編碼)。在憑證內文中，複製並貼上公有金鑰憑證檔案的內容 (PEM 編碼)。在 Certificate Chain (憑證鏈) 中，將憑證鏈檔案的內容 (PEM 編碼) 複製並貼上，除非您使用的是自我簽署憑證，且不介意瀏覽器隱含地接受憑證。
10. (選擇性) 若要啟用相互驗證，請在用戶端憑證處理下啟用相互驗證 (MTL)。

啟用時，預設的相互 TLS 模式為傳遞。

如果您選取「使用信任存放區驗證」：

- 根據預設，會拒絕具有過期用戶端憑證的連線。若要變更此行為，請展開 [進階 MTL 設定]，然後在 [用戶端憑證到期] 下選取 [允許過期的用戶]
- 在 [信任存放區] 下選擇現有的信任存放區，或選擇 [新增信任存放區]
 - 如果您選擇 [新增信任存放區]，請提供信任存放區名稱、S3 URI 憑證授權單位位置，以及選擇性地提供 S3 URI 憑證撤銷清單位置。

11. 選擇儲存。

若要使用 AWS CLI

使用 [create-listener](#) 命令來建立接聽程式和預設規則，以及 [create-rule](#) 命令來定義額外的接聽程式規則。

Application Load Balancer 的接聽程式規則

您為接聽程式定義的規則將決定負載平衡器將請求路由到一個或多個目標群組中之目標的方法。

每個規則由優先順序、一或多個動作及一或多個條件組成。如需詳細資訊，請參閱 [接聽程式規則](#)。

需求

- 規則只能附加到安全偵聽器。
- 每個規則必須包含剛好以下其中一個動作：`forward`、`redirect` 或 `fixed-response`，而且必須是最後要執行的動作。
- 每個規則可以包含以下其中一個條件或都不包含：`host-header`、`http-request-method`、`path-pattern` 和 `source-ip`，以及以下其中一個條件或都不包含：`http-header` 和 `query-string`。
- 每個條件最多可指定三個比較字串，每個規則最多可指定五個比較字串。
- `forward` 動作會將請求路由至它的目標群組。新增 `forward` 動作之前，請建立目標群組並將目標新增至群組。如需詳細資訊，請參閱 [建立目標群組](#)。

新增規則

建立接聽程式時您會定義預設規則，並且可以隨機定義額外的非預設規則。

使用主控台新增規則

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器以檢視其詳細資訊。
4. 在接聽程式和規則索引標籤上，執行下列其中一個動作：
 - a. 選取通訊協定：連接埠資料欄中的文字，即可開啟接聽程式的詳細資訊頁面。

在規則索引標籤上選擇新增規則。

- b. 選取您要對其新增規則的接聽程式。

選擇管理規則，然後選擇新增規則。
5. 您可以在名稱和標籤底下指定規則的名稱，但這並非必填的資訊。

若要新增其他標籤，請選取新增其他標籤文字。

6. 選擇下一步。
7. 選擇新增條件。
8. 新增一個或多個以下條件：
 - 主機標頭 – 定義主機標頭。例如：`*.example.com`。若要儲存條件，請選擇確認。

最多 128 個字元。不區分大小寫。允許的字元有 a-z、A-Z、0-9，以及下列特殊字元：`-_;` 和萬用字元 (`*` 和 `?`)。

- 路徑 – 定義路徑。例如：`/item/*`。若要儲存條件，請選擇確認。

最多 128 個字元。區分大小寫。允許的字元有 a-z、A-Z、0-9，以及下列特殊字元：`_.$/~"@:+; &` 和萬用字元 (`*` 和 `?`)。

- HTTP 請求方法 – 定義 HTTP 請求方法。若要儲存條件，請選擇確認。

最多 40 個字元。區分大小寫。允許的字元有 A-Z 以及下列特殊字元：`-_`。不支援萬用字元。

- 來源 IP – 以 CIDR 格式定義來源 IP 地址。若要儲存條件，請選擇確認。

IPv4 和 IPv6 CIDR 都是允許的。不支援萬用字元。

- HTTP 標頭 – 輸入標頭的名稱，並新增一個或多個比較字串。若要儲存條件，請選擇確認。

- HTTP 標頭名稱 – 規則會評估內含此標頭的請求，以確認相符值。

最多 40 個字元。不區分大小寫。允許的字元有 a-z、A-Z、0-9 以及下列特殊字元：*?!#%&'&'+.^_~。不支援萬用字元。

- HTTP 標頭值 – 輸入要與 HTTP 標頭值比較的字串。

最多 128 個字元。不區分大小寫。允許的字元有 a-z、A-Z、0-9、空格，以及下列特殊字元：!"#%&'()+,./:;#=>@[^_`{}~;-; 和萬用字元 (* 和 ?)。

- 查詢字串 – 根據查詢字串中的鍵值組或值來路由傳送請求。若要儲存條件，請選擇確認。

最多 128 個字元。不區分大小寫。允許的字元有 a-z、A-Z、0-9，以及下列特殊字元：_-.\$/~"@:+&(!,;=; 和萬用字元 (* 和 ?)。

9. 選擇下一步。

10. 為規則定義下列動作之一：

- 轉送至目標群組 – 選擇一個或多個要將流量轉送至其中的目標群組。若要新增目標群組，請選擇新增目標群組。如果使用多個目標群組，請為每個目標群組選取權重，並檢閱相關的百分比。如果您已在一個或多個目標群組上啟用粘性，則必須在規則上啟用群組層級粘性。
- 重新導向至 URL – 指定將用戶端請求重新導向所至的 URL。這可以透過在 URI 部分索引標籤上單獨輸入每個部分，或在完整 URL 索引標籤上輸入完整地址來完成。若是狀態碼，您可以根據需求，將重新導向設定為暫時 (HTTP 302) 或永久 (HTTP 301)。
- 傳回固定回應 – 指定回應代碼，此代碼將傳回至遭捨棄的用戶端請求。此外，您可以指定內容類型和回應內文，但這並非必填的資訊。

11. 選擇下一步。

12. 輸入介於 1-50000 之間的值，以指定規則的「優先順序」。

13. 選擇下一步。

14. 檢閱目前為新規則設定的所有詳細資料和設定。如果對選取項目感到滿意，請選擇建立。

若要使用新增規則 AWS CLI

使用 [create-rule](#) 命令來建立規則。使用 [describe-rules](#) 命令來檢視規則的相關資訊。

編輯規則

您可以隨時編輯規則的動作和條件。規則更新不會立即生效，因此在您更新規則後，可以使用先前的規則組態路由傳送請求一段時間。任何進行中的請求都已完成。

使用主控台編輯規則

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，執行下列其中一個動作：
 - 選取通訊協定：連接埠資料欄中的文字，即可開啟接聽程式的詳細資訊頁面。
 - i. 在規則索引標籤的接聽程式規則區段中，在名稱標籤資料欄中選取要編輯之規則的文字。
選擇動作，然後編輯規則。
 - ii. 在規則索引標籤的接聽程式規則區段中，選取要編輯的規則。
選擇動作，然後編輯規則。
5. 視需要修改名稱和標籤。若要新增其他標籤，請選取新增其他標籤文字。
6. 選擇下一步
7. 視需要修改條件。您可以新增、編輯現有條件或刪除條件。
8. 選擇下一步
9. 視需要修改動作。
10. 選擇下一步
11. 視需要修改規則優先順序。您可以輸入介於 1-50000 之間的值。
12. 選擇下一步
13. 檢閱為您的規則設定的所有詳細資料和更新的設定。一旦您對您的選擇感到滿意，請選擇「儲存變更」。

若要使用編輯規則 AWS CLI

使用 [modify-rule](#) 命令。

更新規則優先順序

依優先順序評估規則，從最低值到最高值。預設規則最後評估。您可以隨時變更非預設規則的優先順序。您無法變更預設規則的優先順序。

使用主控台更新規則優先順序

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，執行下列其中一個動作：
 - a. 選取通訊協定：連接埠或規則資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
 - i. 選擇動作，然後選擇重新排定規則的優先順序。
 - ii. 在規則索引標籤的接聽程式規則區段中，選擇動作，然後選擇重新排定規則的優先順序。
 - b. 選取接聽程式。
 - 選擇管理規則，然後選擇重新排定規則的優先順序
5. 在接聽程式規則區段中，優先順序資料欄會顯示目前的規則優先順序。您可以輸入介於 1-50000 之間的值來更新規則優先順序。
6. 如果滿意變更結果，請選擇儲存變更。

若要使用更新規則優先順序 AWS CLI

使用 [set-rule-priorities](#) 命令。

刪除規則

您可以隨時刪除接聽程式的非預設規則。您無法刪除接聽程式的預設規則。刪除接聽程式時，其所有規則將被刪除。

使用主控台刪除規則

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，執行下列其中一個動作：
 - a. 選取通訊協定：連接埠或規則資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
 - i. 選取您要刪除的規則。

- ii. 依序選擇動作、刪除規則
 - iii. 在文字欄位中輸入 `confirm`，然後選擇刪除。
- b. 選取名稱標籤資料欄中的文字，以開啟規則的詳細資訊頁面。
- i. 依序選擇動作、刪除規則。
 - ii. 在文字欄位中輸入 `confirm`，然後選擇刪除。

若要使用刪除規則 AWS CLI

使用 [delete-rule](#) 命令。

為 Application Load Balancer 更新 HTTPS 接聽程式

建立 HTTPS 接聽程式之後，您可以更換預設憑證、更新憑證清單或更換安全政策。

任務

- [更換預設憑證](#)
- [將憑證新增至憑證清單](#)
- [從憑證清單中移除憑證](#)
- [更新安全政策](#)

更換預設憑證

您可以使用以下程序，更換接聽程式的預設憑證。如需詳細資訊，請參閱 [SSL 憑證](#)。

使用主控台變更預設憑證

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選擇通訊協定：連接埠資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
5. 在憑證索引標籤上，選擇變更預設值。
6. 在 ACM 和 IAM 憑證資料表中，選取新的預設憑證。
7. 選擇儲存為預設。

若要使用變更預設憑證 AWS CLI

使用 [modify-listener](#) 命令。

將憑證新增至憑證清單

您可以使用以下程序，將憑證新增至接聽程式的憑證清單。當您最初建立 HTTPS 接聽程式時，憑證清單是空的。您可以新增一或多個憑證。您可以選擇性新增預設憑證，以確保此憑證即使更換為預設憑證，也會搭配 SNI 通訊協定一起使用。如需詳細資訊，請參閱 [SSL 憑證](#)。

使用主控台變更預設憑證

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選擇通訊協定：連接埠資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
5. 在憑證索引標籤上，選擇新增憑證。
6. 在 ACM 和 IAM 憑證資料表中，選取要新增的憑證，然後選擇包含為下方待處理項目。
7. 如果憑證不是由 ACM 或 IAM 管理，請選擇匯入憑證、完成表單，然後選擇匯入。
8. 選擇新增待定憑證。

使用將憑證新增至憑證清單 AWS CLI

使用 [add-listener-certificates](#) 命令。

從憑證清單中移除憑證

您可以使用以下程序，從 HTTPS 接聽程式的憑證清單中移除憑證。若要移除 HTTPS 接聽程式的預設憑證，請參閱 [更換預設憑證](#)。

使用主控台從憑證清單中移除憑證

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。

4. 在接聽程式和規則索引標籤上，選取通訊協定：連接埠資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
5. 在憑證索引標籤上，選取憑證的核取方塊，然後選擇移除。
6. 出現確認提示時，請輸入 **confirm**，然後選擇移除。

若要使用 AWS CLI

使用 [remove-listener-certificates](#) 命令。

更新安全政策

建立 HTTPS 接聽程式時，您可以選取符合您的需求的安全政策。新增安全政策後，您可以更新 HTTPS 接聽程式，以使用新的安全政策。Application Load Balancers 不支援自訂安全政策。如需詳細資訊，請參閱 [安全政策](#)。

在應用 Application Load Balancer 上使用 FIPS 原則：

連結至「Application Load Balancer」的所有安全接聽程式都必須使用 FIPS 安全性原則或非 FIPS 安全性原則；它們不能混合使用。如果現有的 Application Load Balancer 有兩個或多個使用非 FIP 原則的接聽程式，而且您希望接聽程式改用 FIPS 安全性原則，請移除所有接聽程式，直到只有一個偵聽器為止。將監聽器的安全性原則變更為 FIPS，然後使用 FIPS 安全性原則建立其他接聽程式。或者，您也可以僅使用 FIPS 安全性原則，建立含有新接聽程式的新應用程式負載平衡器。

使用主控台更新安全政策

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選取通訊協定：連接埠資料欄中的文字，以開啟接聽程式的詳細資訊頁面。
5. 在詳細資訊頁面上，選擇動作、然後編輯接聽程式。
6. 在「安全監聽器設定值」段落的「安全性原則」下，選擇新的安全原則。
7. 選擇儲存變更。

若要使用更新安全性原則 AWS CLI

使用 [modify-listener](#) 命令。

Application Load Balancer 中的 TLS 相互驗證

相互 TLS 驗證是傳輸層安全性 (TLS) 的一種變體。傳統 TLS 會在伺服器與用戶端之間建立安全通訊，伺服器需要將其身分提供給用戶端。使用相互 TLS 時，負載平衡器會在交涉 TLS 時交涉用戶端與伺服器之間的相互驗證。當您將相互 TLS 與應用程式負載平衡器搭配使用時，可簡化驗證管理並減少應用程式的負載。

透過將相互 TLS 與應用程式負載平衡器搭配使用，您的負載平衡器可以管理用戶端驗證，以協助確保只有受信任的用戶端與您的後端應用。當您使用此功能時，Application Load Balancer 會使用來自協力廠商憑證授權單位 (CA) 的憑證，或選擇性地使用 AWS Private Certificate Authority (PCA) 進行撤銷檢查來驗證用戶端。應用 Application Load Balancer 會將用戶端憑證資訊傳遞至後端，您的應用程式可用於授權。透過在應用程式負載平衡器中使用相互 TLS，您可以為使用已建立程式庫的憑證型實體取得內建、可調整的受管理驗證。

應用程式負載平衡器的相互 TLS 提供下列兩個選項來驗證 X.509v3 用戶端憑證：

備註：不支援 X.509v1 用戶端憑證。

- **相互 TLS 傳遞：**當您使用相互 TLS 傳遞模式時，Application Load Balancer 會使用 HTTP 標頭將整個用戶端憑證鏈傳送至目標。然後，透過使用用戶端憑證鏈結，您可以在應用程式中實作對應的驗證和授權邏輯。
- **相互 TLS 驗證：**當您使用相互 TLS 驗證模式時，Application Load Balancer 會在負載平衡器交涉 TLS 連線時，為用戶端執行 X.509 用戶端憑證驗證。

若要在應用 Application Load Balancer 中使用傳遞開始使用相互 TLS，您只需要設定接聽程式以接受來自用戶端的任何憑證即可。若要搭配驗證使用相互 TLS，您必須執行下列動作：

- 建立新的信任儲存區資源。
- 上傳您的憑證授權單位 (CA) 服務包，以及 (選擇性) 撤銷清單。
- 將信任儲存區附加到設定為驗證用戶端憑證的監聽器。

如需使用應用 step-by-step 程式負載平衡器設定相互 TLS 驗證模式的程序，請參閱 [在 Application Load Balancer 器上設定相互 TLS](#)。

在 Application Load Balancer 上開始設定相互 TLS 之前

在 Application Load Balancer 上開始設定相互 TLS 之前，請注意下列事項：

配額

應用程式負載平衡器包含與您 AWS 帳戶中使用的信任存放區、CA 憑證和憑證撤銷清單數量相關的某些限制。

如需詳細資訊，請參閱[應用程式負載平衡器的配額](#)。

憑證需求

應用程式負載平衡器針對與相互 TLS 驗證搭配使用的憑證支援下列項目：

- 支援的憑證：
- 支持的公共密鑰：RSA 2K-8K 或 ECDSA 秒 256r1，秒 521r1
- 支援的簽章演算法：SHA256、384、512 搭配使用 RSA/SHA256、384、512 搭配使用 RSA/SHA256,384,512 雜湊，搭配 MGF1

CA 憑證組合包

以下內容適用於憑證授權單位 (CA) 套裝軟體：

- 應用程式負載平衡器會以批次方式上傳每個憑證授權單位 (CA) 憑證服務包。應用程式負載平衡器不支援上傳個別憑證。如果您需要新增憑證，您必須上傳憑證組合檔案。
- 若要取代 CA 憑證服務包，請使用[ModifyTrust存放區](#) API。

傳遞的憑證訂單

當您使用相互 TLS 傳遞時，Application Load Balancer 會插入標頭，將用戶端憑證鏈結呈現給後端目標。簡報的順序從葉證書開始，並以根證書結束。

工作階段恢

透過應用程式負載平衡器使用相互 TLS 傳遞或驗證模式時，不支援工作階段重新開始。

HTTP 標頭

應用程式負載平衡器使用相互 TLS 交涉用戶端連線時，會使用 X-Amzn-Mtls 標頭傳送憑證資訊。如需詳細資訊和範例標頭，請參閱[HTTP 標頭和相互 TLS](#)。

CA 憑證檔案

CA 憑證檔案必須符合下列需求：

- 憑證檔案必須使用 PEM (隱私權加強郵件) 格式。
- 憑證內容必須包含在-----BEGIN CERTIFICATE-----和-----END CERTIFICATE-----邊界內。
- 註解前必須有一個#字元。

- 不能有任何空行。

不接受的憑證範例 (無效) :

```
# comments

Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number: 01
  Signature Algorithm: ecdsa-with-SHA384
  Issuer: C=US, O=EXAMPLE, OU=EXAMPLE, CN=EXAMPLE
  Validity
    Not Before: Jan 11 23:57:57 2024 GMT
    Not After : Jan 10 00:57:57 2029 GMT
  Subject: C=US, O=EXAMPLE, OU=EXAMPLE, CN=EXAMPLE
  Subject Public Key Info:
    Public Key Algorithm: id-ecPublicKey
    Public-Key: (384 bit)
    pub:
      00:01:02:03:04:05:06:07:08
    ASN1 OID: secp384r1
    NIST CURVE: P-384
  X509v3 extensions:
    X509v3 Key Usage: critical
      Digital Signature, Key Encipherment, Certificate Sign, CRL Sign
    X509v3 Basic Constraints: critical
      CA:TRUE
    X509v3 Subject Key Identifier:
      00:01:02:03:04:05:06:07:08
    X509v3 Subject Alternative Name:
      URI:EXAMPLE.COM
  Signature Algorithm: ecdsa-with-SHA384
    00:01:02:03:04:05:06:07:08
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
```

接受的憑證範例 (有效) :

1. 單一憑證 (PEM 編碼) :

```
# comments
-----BEGIN CERTIFICATE-----
```

```
Base64-encoded certificate
-----END CERTIFICATE-----
```

2. 多個憑證 (PEM 編碼) :

```
# comments
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
# comments
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
```

HTTP 標頭和相互 TLS

本節說明在與使用相互 TLS 的用戶端交涉連線時，應用程式負載平衡器用來傳送憑證資訊的 HTTP 標頭。應用程式負載平衡器使用的特定 X-Amzn-Mtls 標頭取決於您指定的相互 TLS 模式：傳遞模式或驗證模式。

如需應用程式負載平衡器支援的其他 HTTP 標頭的相關資訊，請參閱 [HTTP 標頭和 Application Load Balancer](#)。

用於直通模式的 HTTP 標頭

對於傳遞模式下的相互 TLS，應用程式負載平衡器會使用下列標頭。

X-阿姆贊-山特倫-客戶端證書

此標頭包含連線中顯示之整個用戶端憑證鏈結的 URL 編碼 PEM 格式，並以安全字元形式顯示。+ = /

示例頭內容：

```
X-Amzn-Mtls-Clientcert: -----BEGIN%20CERTIFICATE-----%0AMIID<...reduced...>do0g
%3D%3D%0A-----END%20CERTIFICATE-----%0A-----BEGIN%20CERTIFICATE-----
%0AMIID1<...reduced...>3eZlyKA%3D%3D%0A-----END%20CERTIFICATE-----%0A
```

用於驗證模式的 HTTP 標頭

對於驗證模式下的相互 TLS，應用程式負載平衡器會使用下列標頭。

X-阿姆森-山地客戶證序列號

此標頭包含分葉憑證序列號的十六進位表示法。

示例頭內容：

```
X-Amzn-Mtls-Clientcert-Serial-Number: 03A5B1
```

X-阿姆津-山地區-客戶發行人

此標頭包含簽發者辨別名稱 (DN) 的 RFC2253 字串表示。

示例頭內容：

```
X-Amzn-Mtls-Clientcert-Issuer:  
CN=rootcamtls.com,OU=rootCA,O=mTLS,L=Seattle,ST=Washington,C=US
```

X-安森-山地區客戶端主題

此標頭包含主旨辨別名稱 (DN) 的 RFC2253 字串表示。

示例頭內容：

```
X-Amzn-Mtls-Clientcert-Subject: CN=client_.com,OU=client-3,O=mTLS,ST=Washington,C=US
```

X-安森-MTLS 客戶端有效性

此標頭包含notBefore和notAfter日期的 ISO8601 格式。

示例頭內容：

```
X-Amzn-Mtls-Clientcert-Validity:  
NotBefore=2023-09-21T01:50:17Z;NotAfter=2024-09-20T01:50:17Z
```

X-阿姆津山脈-客戶端-葉

此標頭包含葉憑證的 URL 編碼 PEM 格式，+=/以安全字元形式。

示例頭內容：

```
X-Amzn-Mtls-Clientcert-Leaf: -----BEGIN%20CERTIFICATE-----%0AMIIG<...reduced...>NmUlw%0A-----END%20CERTIFICATE-----%0A
```

在 Application Load Balancer 器上設定相互 TLS

本節包括為應用程式負載平衡器上的驗證設定相互 TLS 驗證模式的程序。

若要使用相互 TLS 傳遞模式，您只需要將接聽程式設定為接受來自用戶端的任何憑證。當您使用相互 TLS 傳遞時，應用 Application Load Balancer 會使用 HTTP 標頭將整個用戶端憑證鏈傳送至目標，讓您在應用程式中實作對應的驗證和授權邏輯。如需詳細資訊，請參閱[為應用程式負載平衡器建立 HTTPS 接聽程式](#)。

當您在驗證模式下使用相互 TLS 時，當負載平衡器交涉 TLS 連線時，應用程式負載平衡器會為用戶端執行 X.509 用戶端憑證驗證。

若要使用相互 TLS 驗證模式，請執行下列動作：

- 建立新的信任儲存區資源。
- 上傳您的憑證授權單位 (CA) 服務包，以及 (選擇性) 撤銷清單。
- 將信任儲存區附加到設定為驗證用戶端憑證的監聽器。

請遵循本節中的程序，在中的應用程式負載平衡器上設定相互 TLS 驗證模式 AWS Management Console。若要使用 API 作業而非主控台來設定相互 TLS，請參閱[應用程式負載平衡器 API 參考指南](#)。

任務

- [建立信任存放區](#)
- [建立信任存放區關聯](#)
- [檢視信任存放區詳細](#)
- [修改信任存放區](#)
- [刪除信任存放區](#)

建立信任存放區

建立信任存放區的方法有三種：建立應用程式負載平衡器時、建立安全接聽程式時，以及使用 Trust Store 主控台。當您在建立負載平衡器或接聽程式時新增信任存放區時，信任存放區會自動與新的接聽程式產生關聯。使用「信任存放區」主控台建立信任存放區時，您必須自行將其與接聽程式建立關聯。

本節涵蓋使用 Trust Store 主控台建立信任存放區，但建立應用程式負載平衡器或接聽程式時所使用的步驟相同。如需詳細資訊，請參閱[設定負載平衡器和接聽程式](#)和[新增 HTTPS 接聽程式](#)。

事前準備：

- 若要建立信任存放區，您必須擁有來自憑證授權單位 (CA) 的憑證服務包。

使用主控台建立信任存放區

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取建立信任存放區。
4. 信任存放區組態
 - a. 針對「信任」商店名稱，輸入信任存放區的名稱。
 - b. 對於憑證授權單位服務包，請輸入您希望信任存放區使用的 ca 憑證服務包的 Amazon S3 路徑。

可選：使用物件版本選取先前版本的 ca 憑證套裝軟體。否則會使用目前的版本。

5. 對於撤銷，您可以選擇性地將憑證撤銷清單新增至您的信任存放區。
 - 在憑證撤銷清單下，輸入您希望信任存放區使用的憑證撤銷清單的 Amazon S3 路徑。

可選：使用物件版本選取舊版憑證撤銷清單。否則會使用目前的版本。

6. 對於 Trust 商店標籤，您可以選擇性地輸入最多 50 個標籤，以套用至您的信任存放區。
7. 選取建立信任存放區。

建立信任存放區關聯

建立信任存放區之後，您必須先將其與接聽程式建立關聯，應用程式負載平衡器才能開始使用信任存放區。您只能將一個信任存放區與每個安全偵聽程式相關聯，但是一個信任存放區可以與多個偵聽程式相關聯。

本節涵蓋將信任存放區與現有接聽程式相關聯。或者，您可以在建立應用程式負載平衡器或接聽程式時關聯信任儲存區。如需詳細資訊，請參閱[設定負載平衡器和接聽程式](#)和[新增 HTTPS 接聽程式](#)。

使用主控台建立信任儲存區的關聯

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器以檢視其詳細資訊頁面。
4. 在監聽器和規則頁籤上，選擇協定：連接埠資料欄中的連結，以開啟安全監聽器的詳細資訊頁面。
5. 在安全性索引標籤上，選擇編輯安全接聽程式設定。
6. (選擇性) 如果未啟用相互 TLS，請選取 [用戶端憑證處理] 下的 [相互驗證 (MTL)]，然後選擇 [以信任存放區驗證]。
7. 在 [信任存放區] 底下，選擇您建立的信任存放區。
8. 選擇儲存變更。

檢視信任存放區詳細

CA 憑證組合包

CA 憑證服務包是信任存放區的必要元件。它是受信任的根憑證和中繼憑證的集合，這些憑證已經由憑證授權單位驗證。這些驗證的憑證可確保用戶端可以信任所提供的憑證是由負載平衡器所擁有。

您可以隨時在信任存放區中檢視目前 CA 憑證服務包的內容。

檢視 CA 憑證套裝軟體

使用控制台檢視 CA 憑證套裝軟體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視詳細資料頁面。
4. 選擇動作，然後選擇取得 CA 套裝軟體。
5. 選擇 [分享連結] 或 [下載]。

憑證撤銷清單

或者，您可以為信任存放區建立憑證撤銷清單。撤銷清單會由憑證授權單位發行，並包含已撤銷之憑證的資料。應用程式負載平衡器僅支援 PEM 格式的憑證撤銷清單。

將憑證撤銷清單新增至信任存放區時，會提供撤銷 ID。每次新增至信任存放區的撤銷清單，撤銷 ID 都會增加，而且無法變更。如果從信任存放區刪除憑證撤銷清單，它的撤銷 ID 也會一併刪除，而且不會在信任存放區的有效期間重複使用。

Note

應用程式負載平衡器無法撤銷憑證撤銷清單中具有負數序號的憑證。

檢視憑證撤銷清單

使用主控台檢視撤銷清單

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視詳細資料頁面。
4. 在憑證撤銷清單索引標籤上，選取動作，然後選取取得撤銷清單。
5. 選擇 [分享連結] 或 [下載]。

修改信任存放區

信任存放區一次只能包含一個 CA 憑證服務包，但您可以在建立信任存放區之後隨時取代 CA 憑證服務包。

取代 CA 憑證套裝軟體

使用控制台取代 CA 憑證套裝軟體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視詳細資料頁面。
4. 選擇動作，然後選擇取代 CA 套裝軟體。
5. 在「取代 CA 服務包」頁面的「憑證授權單位服務包」下，輸入所需 CA 服務包的 Amazon S3 位置。
6. (選擇性) 使用物件版本選取舊版憑證撤銷清單。否則會使用目前的版本。
7. 選取取代 CA 套裝軟體。

新增憑證撤銷清單

若要使用主控台新增撤銷清單

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視其詳細資料頁面。
4. 在憑證撤銷清單索引標籤上，選取動作，然後選取新增撤銷清單。
5. 在 [新增撤銷清單] 頁面的 [憑證撤銷清單] 下，輸入所需憑證撤銷清單的 Amazon S3 位置
6. (選擇性) 使用物件版本選取舊版憑證撤銷清單。否則會使用目前的版本。
7. 選擇添加撤銷列表

刪除憑證撤銷清單

若要使用主控台刪除撤銷清單

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視詳細資料頁面。
4. 在憑證撤銷清單索引標籤上，選取動作，然後選取刪除撤銷清單。
5. 輸入以確認刪除confirm。
6. 選取刪除。

刪除信任存放區

當您不再使用信任存放區時，您可以將其刪除。

注意：您無法刪除目前與監聽器關聯的信任儲存區。

使用主控台刪除信任存放區

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格中，選擇 [信任存放區]。
3. 選取信任存放區以檢視其詳細資料頁面。
4. 選擇動作，然後選擇刪除信任存放區。

5. 輸入以確認刪除confirm。
6. 選擇刪除

應用程式負載平衡器的連線記錄

Elastic Load Balancing 提供連線記錄，可擷取傳送至應用程式負載平衡器之要求的相關屬性。連線記錄包含資訊，例如用戶端 IP 位址和連接埠、用戶端憑證資訊、連線結果，以及正在使用的 TLS 密碼。然後，這些連線記錄可用於檢閱請求模式和其他趨勢。

若要深入瞭解連線記錄，請參閱 [Application Load Balancer 的連線記錄](#)

使用 Application Load Balancer 來驗證使用者身分

您可以設定 Application Load Balancer，以在使用者存取應用程式時安全地驗證使用者身分。這可讓您將驗證使用者的工作卸載到您的負載平衡器，使得您的應用程式可以專注在商業邏輯上。

支援下列使用案例：

- 透過與 OpenID Connect (OIDC) 相容的身分提供者 (IdP) 驗證使用者。
- 透過 Amazon Cognito 支援的使用者集區 Facebook，透過社交媒體 IdPs (例如亞馬遜或 Google) 對使用者進行身份驗證。
- 透過 Amazon Cognito 支援的使用者集區，使用 SAML、OpenID Connect (OIDC) 或 OAuth，透過公司身分來驗證使用者身分。

準備使用 OIDC 合規 IdP

如果您使用 OIDC 合規 IdP 搭配 Application Load Balancer，請執行下列動作：

- 在 IdP 中建立新 OIDC 應用程式。IdP 的 DNS 必須可公開解析。
- 您必須設定用戶端 ID 和用戶端機密。
- 取得 IdP 發佈的以下端點：授權、字符和使用者資訊。您可以在此組態中找到此資訊。
- IdP 端點憑證的發行者應是受信任的公用憑證授權單位。
- 端點的 DNS 項目必須可公開解析，即使會解析為私有 IP 地址也可以。
- 允許以下其中一個重新導向 URL 加入您的使用者將使用的 IdP 應用程式，其中的 DNS 是您的負載平衡器的網域名稱，而 CNAME 為您的應用程式的 DNS 別名：

- <https://DNS/oauth2/idpresponse>
- <https://CNAME/oauth2/idpresponse>

準備使用 Amazon Cognito

可用地區

下列區域提供適用於應用程式負載平衡器的 Amazon Cognito 整合：

- 美國東部 (維吉尼亞北部)
- 美國東部 (俄亥俄)
- 美國西部 (加利佛尼亞北部)
- 美國西部 (奧勒岡)
- 加拿大 (中部)
- 歐洲 (斯德哥爾摩)
- 歐洲 (米蘭)
- 歐洲 (法蘭克福)
- 歐洲 (蘇黎世)
- 歐洲 (愛爾蘭)
- 歐洲 (倫敦)
- Europe (Paris)
- 南美洲 (聖保羅)
- 亞太區域 (東京)
- 亞太區域 (首爾)
- 亞太區域 (大阪)
- 亞太區域 (孟買)
- 亞太區域 (新加坡)
- 亞太區域 (悉尼)
- 亞太區域 (雅加達)
- 中東 (阿拉伯聯合大公國)
- Middle East (Bahrain)

- 非洲 (開普敦)
- 以色列 (特拉維夫)

如果您使用與 Amazon Cognito 使用者集區搭配 Application Load Balancer，請執行下列動作：

- 建立使用者集區。如需詳細資訊，請參閱《Amazon Cognito 開發人員指南》中的[Amazon Cognito 使用者集區](#)。
- 建立使用者集區用戶端。您必須設定用戶端來產生用戶端機密，使用代碼授予流程，並支援負載平衡器使用的相同 OAuth 範圍。如需詳細資訊，請參閱《Amazon Cognito 開發人員指南》中的[設定使用者集區應用程式用戶端](#)。
- 建立使用者集區物件網域。如需詳細資訊，請參閱《Amazon Cognito 開發人員指南》中的[為使用者集區新增網域名稱](#)。
- 驗證請求的範圍傳回 ID 字符。例如，預設範圍 `openid` 會傳回 ID 字符，但 `aws.cognito.signin.user.admin` 範圍則不會。

備註：應用程式負載平衡器不支援 Amazon Cognito 發行的自訂存取權杖。如需詳細資訊，請參閱 Amazon Cognito 開發人員指南中的[預先產生權杖](#)。

- 若要聯合社交或公司 IdP，請在聯合區段啟用 IdP。如需詳細資訊，請參閱《Amazon Cognito 開發人員指南》中的[新增社交登入至使用者集區或以 SAML 身分提供者新增登入至使用者集區](#)。
- 允許 Amazon Cognito 的回呼 URL 欄位中使用以下重新導向 URL，其中 DNS 是負載平衡器的網域名稱，而 CNAME 為應用程式的 DNS 別名 (如果您有使用)：
 - `https://DNS/oauth2/idpresponse`
 - `https://CNAME/oauth2/idpresponse`
- 允許 IdP 應用程式的回呼 URL 中使用使用者集區網域。針對 IdP 使用此格式。例如：
 - `https://domain-prefix.auth.region.amazoncognito.com/saml2/idpresponse`
 - `https://user-pool-domain/oauth2/idpresponse`

應用程式用戶端設定中的回呼 URL 必須全都使用小寫字母。

若要讓使用者設定負載平衡器以使用 Amazon Cognito 來驗證使用者身分，您必須授予使用者呼叫 `cognito-idp:DescribeUserPoolClient` 動作的許可。

準備使用 Amazon CloudFront

如果您在應用 Application Load Balancer 前面使用 CloudFront 散發，請啟用下列設定：

- 轉寄要求標頭 (all) — 確保 CloudFront 不快速取已驗證要求的回應。這可避免在驗證工作階段過期後從快速取中提供回應。或者，若要在啟用快速取時降低此風險，CloudFront 發行版的擁有者可以將 time-to-live (TTL) 值設定為在驗證 Cookie 到期之前過期。
- 查詢字串轉送和快速取 (全部) – 確保負載平衡器可存取向 IdP 驗證使用者身分時所需的查詢字串參數。
- Cookie 轉送 (全部) — 確保將所有驗證 Cookie CloudFront 轉送至負載平衡器。

設定使用者身分驗證

您透過為一個或多個接聽程式規則建立驗證動作來設定使用者身分驗證。authenticate-cognito 和 authenticate-oidc 動作類型僅支援使用 HTTPS 接聽程式。如需對應欄位的說明，請參閱 Elastic Load Balancing API 參考版本 2015-12-01 [AuthenticateOidcActionConfig](#) 中的 [AuthenticateCognitoActionConfig](#) 和。

負載平衡器會傳送工作階段 Cookie 給用戶端，以維護驗證狀態。此 Cookie 永遠包含 secure 屬性，因為使用者驗證需要 HTTPS 接聽程式。此 Cookie 包含具有 CORS (跨來源資源共享) 請求的 SameSite=None 屬性。

對於支援需要獨立用戶端身分驗證之多個應用程式的負載平衡器，每個具有身分驗證動作的接聽程式規則都應具有唯一的 Cookie 名稱。這可確保用戶端在路由傳送至規則中指定的目標群組之前，一律會使用 IdP 進行驗證。

Application Load Balancer 不支援 URL 編碼的 Cookie 值。

依預設，SessionTimeout 欄位會設為 7 天。如果您需要較短的工作階段，您可以將工作階段逾時設定為最短至 1 秒。如需詳細資訊，請參閱 [工作階段逾時](#)。

根據您的應用程式適當地設定 OnUnauthenticatedRequest 欄位。例如：

- 需要使用者使用社交或公司身分登入的應用程式 – 這是透過預設選項 (authenticate) 支援。如果使用者未登入，負載平衡器會將請求重新導向至 IdP 授權端點，並且 IdP 會提示使用者使用其使用者界面登入。
- 為登入的使用者提供個人化檢視或對未登入的使用者提供一般檢視的應用程式 – 若要支援這類應用程式，請使用 allow 選項。如果使用者已登入，負載平衡器會提供使用者宣告，而應用程式可提供個人化的檢視。如果使用者未登入，負載平衡器會轉送不帶使用者宣告的請求，而應用程式可提供一般檢視。
- 載入每隔幾秒鐘的單一頁面應用程式 — 如果您使用 JavaScript 此 deny 選項，負載平衡器會將 HTTP 401 未經授權的錯誤傳回給沒有驗證資訊的 AJAX 呼叫。但是，如果使用者具有已過期的身分驗證資訊，則負載平衡器會將用戶端重新導向至 IdP 授權端點。

負載平衡器必須能夠與 IdP 字符端點 (TokenEndpoint) 和 IdP 使用者資訊端點 (UserInfoEndpoint) 通訊。應用程式負載平衡器僅在與這些端點通訊時支援 IPv4。如果您的 IdP 使用公用位址，請確保負載平衡器的安全群組和 VPC 的網路 ACL 允許存取端點。使用內部負載平衡器或 IP 位址類型時 `dualstack-without-public-ipv4`，NAT 閘道可讓負載平衡器與端點通訊。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [NAT 閘道基本概念](#)。

使用以下 `create-rule` 命令來設定使用者驗證。

```
aws elbv2 create-rule --listener-arn listener-arn --priority 10 \
--conditions Field=path-pattern,Values="/login" --actions file://actions.json
```

以下是指定 `authenticate-oidc` 動作和 `forward` 動作的 `actions.json` 檔案範例。AuthenticationRequestExtraParams 可讓您在身分驗證期間將額外的參數傳遞至 IdP。請依照身分提供者提供的文件來判斷受支援的欄位

```
[{
  "Type": "authenticate-oidc",
  "AuthenticateOidcConfig": {
    "Issuer": "https://idp-issuer.com",
    "AuthorizationEndpoint": "https://authorization-endpoint.com",
    "TokenEndpoint": "https://token-endpoint.com",
    "UserInfoEndpoint": "https://user-info-endpoint.com",
    "ClientId": "abcdefghijklmnopqrstuvwxy123456789",
    "ClientSecret": "123456789012345678901234567890",
    "SessionCookieName": "my-cookie",
    "SessionTimeout": 3600,
    "Scope": "email",
    "AuthenticationRequestExtraParams": {
      "display": "page",
      "prompt": "login"
    },
    "OnUnauthenticatedRequest": "deny"
  },
  "Order": 1
},
{
  "Type": "forward",
  "TargetGroupArn": "arn:aws:elasticloadbalancing:region-code:account-id:targetgroup/target-group-name/target-group-id",
  "Order": 2
}]
```

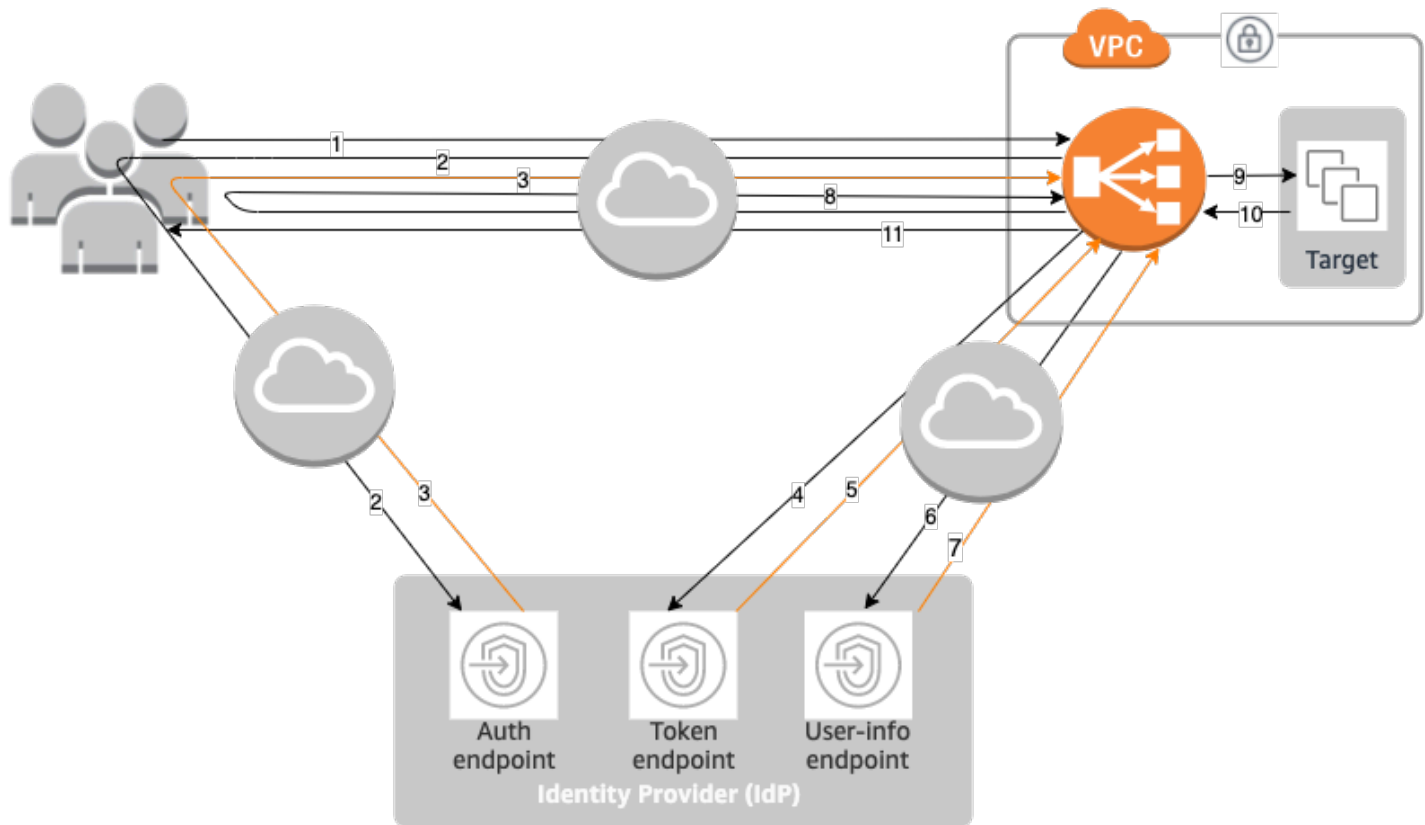
下列是會指定 `authenticate-cognito` 動作和 `forward` 動作的 `actions.json` 檔案的範例。

```
[[
  {
    "Type": "authenticate-cognito",
    "AuthenticateCognitoConfig": {
      "UserPoolArn": "arn:aws:cognito-idp:region-code:account-id:userpool/user-pool-id",
      "UserPoolClientId": "abcdefghijklmnopqrstuvxyz123456789",
      "UserPoolDomain": "userPoolDomain1",
      "SessionCookieName": "my-cookie",
      "SessionTimeout": 3600,
      "Scope": "email",
      "AuthenticationRequestExtraParams": {
        "display": "page",
        "prompt": "login"
      },
      "OnUnauthenticatedRequest": "deny"
    },
    "Order": 1
  },
  {
    "Type": "forward",
    "TargetGroupArn": "arn:aws:elasticloadbalancing:region-code:account-id:targetgroup/target-group-name/target-group-id",
    "Order": 2
  }
]]
```

如需詳細資訊，請參閱 [接聽程式規則](#)。

身分驗證流程

下列網路圖是 Application Load Balancer 如何使用 OIDC 驗證使用者身分的視覺化表示。



下面的帶編號項目著重解釋在前述網路圖中顯示的元素。

1. 使用者將 HTTPS 請求傳送至在 Application Load Balancer 後方託管的網站。當規則的條件滿足某個驗證動作時，負載平衡器會請求標頭中檢查身分驗證工作階段 Cookie。
2. 如果 Cookie 不存在，負載平衡器會將使用者重新導向到 IdP 授權端點，使得 IdP 可驗證使用者。
3. 在使用者通過身分驗證之後，IdP 會將使用者重新傳送回負載平衡器並帶有一個授權授予代碼。
4. 負載平衡器會向 IdP 權杖端點提供授權授予代碼。
5. 在收到有效的授權授予代碼後，IdP 會向 Application Load Balancer 提供 ID 權杖和存取權杖。
6. 然後，Application Load Balancer 會將存取權杖傳送到使用者資訊端點。
7. 使用者資訊端點會將存取權杖交換為使用者宣告。
8. Application Load Balancer 會將具有 AWSELB 身分驗證工作階段 Cookie 的使用者重新導向至原始 URI。由於大部分瀏覽器會將 Cookie 大小限制在 4K，負載平衡器會將大小大於 4K 的 Cookie 分成多個 Cookie 碎片。如果從 IdP 收到的使用者宣告和存取字符的總大小大於 11K 位元組，負載平衡器會傳回 HTTP 500 錯誤給用戶端，並且遞增 ELBAuthUserClaimsSizeExceeded 指標。
9. Application Load Balancer 會驗證 Cookie，並將使用者資訊轉送至 X-AMZN-OIDC-* HTTP 標頭集中的目標。如需詳細資訊，請參閱 [使用者宣告編碼和簽章驗證](#)。

10. 此目標會將回應傳送至 Application Load Balancer。
11. Application Load Balancer 會將最終回應傳送給使用者。

每個新的請求都會經歷步驟 1 到 11，而後續的請求則會經歷步驟 9 到 11。也就是說，只要 Cookie 尚未過期，每個後續請求都會從步驟 9 開始。

使用者在 IdP 進行身分驗證之後，系統會將 AWSALBAuthNonce Cookie 新增至請求標頭。這並不會變更 Application Load Balancer 處理來自 IdP 重新導向請求的方式。

如果 IdP 在 ID 字符中提供有效的重新整理字符，負載平衡器會儲存該重新整理字符，並在每次存取字符過期時使用它來重新整理使用者宣告，直到工作階段逾時或 IdP 重新整理失敗。如果使用者登出，重新整理會失敗，並且負載平衡器會將使用者重新導向至 IdP 授權端點。這可讓負載平衡器在使用者登出後捨棄工作階段。如需詳細資訊，請參閱 [工作階段逾時](#)。

Note

Cookie 到期時間與身分驗證工作階段到期時間不同。Cookie 到期時間是 Cookie 的一種屬性，設為 7 天。身分驗證工作階段的實際長度取決於在 Application Load Balancer 上為身分驗證功能設定的工作階段逾時。此工作階段逾時包含在 Auth Cookie 值中，該值也經過加密。

使用者宣告編碼和簽章驗證

負載平衡器成功驗證使用者之後，它會將從 IdP 收到的使用者宣告傳送到目標。負載平衡器會簽署使用者宣告，讓應用程式可以驗證簽章並驗證宣告是由負載平衡器傳送。

負載平衡器會新增下列 HTTP 標頭：

`x-amzn-oidc-accesstoken`

來自字符端點的存取字符，純文字格式。

`x-amzn-oidc-identity`

來自使用者資訊端點的主旨欄位 (sub)，純文字格式。

注意：子宣告是識別特定使用者的最佳方法。

`x-amzn-oidc-data`

使用者宣告，JSON Web 字符 (JWT) 格式。

存取權杖和使用者宣告與 ID 權杖不同。存取權杖和使用者宣告僅允許存取伺服器資源，而 ID 權杖會附帶其他資訊來對使用者進行身分驗證。應用程式負載平衡器在對用戶進行身份驗證時創建一個新的訪問令牌，並僅將訪問令牌和聲明傳遞給後端，但是它不會傳遞 ID 令牌信息。

這些字符採用 JWT 格式，但不是 ID 字符。JWT 格式包含使用 base64 URL 編碼的標頭、承載和簽章，而且尾端包含填補字元。Application Load Balancer 會使用 ES256 (使用 P-256 和 SHA256 的 ECDSA) 來產生 JWT 簽章。

JWT 標頭是 JSON 物件，具有下列欄位：

```
{
  "alg": "algorithm",
  "kid": "12345678-1234-1234-1234-123456789012",
  "signer": "arn:aws:elasticloadbalancing:region-code:account-id:loadbalancer/
app/load-balancer-name/load-balancer-id",
  "iss": "url",
  "client": "client-id",
  "exp": "expiration"
}
```

JWT 承載為 JSON 物件，其中包含從 IdP 使用者資訊端點收到的使用者宣告。

```
{
  "sub": "1234567890",
  "name": "name",
  "email": "alias@example.com",
  ...
}
```

由於負載平衡器不會加密使用者宣告，我們建議您將目標群組設定為使用 HTTPS。如果將您的目標群組設定為使用 HTTP，請務必使用安全群組限制對您的負載平衡器的流量。

為了確保安全性，您必須在根據宣告執行任何授權之前驗證簽章，並驗證 JWT 標頭中的 `signer` 欄位是否包含預期的 Application Load Balancer ARN。

若要取得公有金鑰，請從 JWT 標頭取得金鑰 ID，並用其在端點查閱公有金鑰。每個 AWS 區域的端點如下：

```
https://public-keys.auth.elb.region.amazonaws.com/key-id
```

對於 AWS GovCloud (US)，端點如下所示：

```
https://s3-us-gov-west-1.amazonaws.com/aws-elb-public-keys-prod-us-gov-west-1/key-id
https://s3-us-gov-east-1.amazonaws.com/aws-elb-public-keys-prod-us-gov-east-1/key-id
```

下列範例顯示如何在 Python 3.x 中取得金鑰 ID、公有金鑰和承載：

```
import jwt
import requests
import base64
import json

# Step 1: Validate the signer
expected_alb_arn = 'arn:aws:elasticloadbalancing:region-code:account-id:loadbalancer/
app/load-balancer-name/load-balancer-id'

encoded_jwt = headers.dict['x-amzn-oidc-data']
jwt_headers = encoded_jwt.split('.')[0]
decoded_jwt_headers = base64.b64decode(jwt_headers)
decoded_jwt_headers = decoded_jwt_headers.decode("utf-8")
decoded_json = json.loads(decoded_jwt_headers)
received_alb_arn = decoded_json['signer']

assert expected_alb_arn == received_alb_arn, "Invalid Signer"

# Step 2: Get the key id from JWT headers (the kid field)
kid = decoded_json['kid']

# Step 3: Get the public key from regional endpoint
url = 'https://public-keys.auth.elb.' + region + '.amazonaws.com/' + kid
req = requests.get(url)
pub_key = req.text

# Step 4: Get the payload
payload = jwt.decode(encoded_jwt, pub_key, algorithms=['ES256'])
```

下列範例顯示如何在 Python 2.7 中取得金鑰 ID、公有金鑰和承載：

```
import jwt
import requests
import base64
import json

# Step 1: Validate the signer
```

```
expected_alb_arn = 'arn:aws:elasticloadbalancing:region-code:account-id:loadbalancer/
app/load-balancer-name/load-balancer-id'

encoded_jwt = headers.dict['x-amzn-oidc-data']
jwt_headers = encoded_jwt.split('.')[0]
decoded_jwt_headers = base64.b64decode(jwt_headers)
decoded_json = json.loads(decoded_jwt_headers)
received_alb_arn = decoded_json['signer']

assert expected_alb_arn == received_alb_arn, "Invalid Signer"

# Step 2: Get the key id from JWT headers (the kid field)
kid = decoded_json['kid']

# Step 3: Get the public key from regional endpoint
url = 'https://public-keys.auth.elb.' + region + '.amazonaws.com/' + kid
req = requests.get(url)
pub_key = req.text

# Step 4: Get the payload
payload = jwt.decode(encoded_jwt, pub_key, algorithms=['ES256'])
```

考量事項

- 這些範例並不包括如何使用權杖中的簽章來驗證發行者的簽章。
- 標準程式庫與 JWT 格式的 Application Load Balancer 驗證權杖中包含的填補不相容。

逾時

工作階段逾時

重新整理字符和工作階段逾時的共同運作方式如下所示：

- 如果工作階段逾時短於存取字符過期，負載平衡器會採用該工作階段逾時。如果使用者有 IdP 的主動工作階段，應該就不會提示使用者再次登入。否則，系統會將使用者重新導向至登入。
- 如果 IdP 工作階段逾時較 Application Load Balancer 工作階段逾時更久，則使用者不需要提供憑證即可重新登入。而且，IdP 會使用新的授權授予代碼重新導向至 Application Load Balancer。即使沒有重新登入，授權碼也只能使用一次。

- 如果 IdP 工作階段逾時較 Application Load Balancer 工作階段逾時一樣久或更久，則會要求使用者提供憑證以重新登入。使用者登入後，IdP 會使用新的授權授予代碼重新導向至 Application Load Balancer，其餘的身分驗證流程會繼續進行，直到請求到達後端為止。
- 如果工作階段逾時較存取權杖過期時間更久，並且 IdP 不支援重新整理權杖，負載平衡器會保留該身分驗證工作階段，直到其逾時為止。然後，負載平衡器會要求使用者再次登入。
- 如果工作階段逾時為長於存取字符過期，並且 IdP 支援重新整理字符，負載平衡器會在每次存取字符過期時重新整理該使用者工作階段。負載平衡器只會在驗證工作階段逾或重新整理流程失敗時，才會將使用者再次登入。

用戶端登入逾時

用戶端必須在 15 分鐘內啟動並完成身分驗證程序。如果用戶端無法在 15 分鐘的限制內完成身分驗證，則會從負載平衡器收到 HTTP 401 錯誤。此逾時限制無法變更或移除。

例如，如果使用者透過 Application Load Balancer 載入登入頁面，則必須在 15 分鐘內完成登入程序。如果使用者在 15 分鐘逾時過期後嘗試登入，負載平衡器會傳回 HTTP 401 錯誤。使用者必須重新整理頁面並嘗試再次登入。

身分驗證登出

當應用程式需要將已驗證的使用者登出時，您應該將驗證工作階段 Cookie 的過期時間設定為 -1，並將用戶端重新導向至 IdP 登出端點 (如果 IdP 有支援)。為了防止使用者重複使用已刪除的 Cookie，建議您將存取字符設定為合理的簡短過期時間。如果用戶端向負載平衡器提供的工作階段 Cookie 具有過期的存取權杖 (具有非 NULL 重新整理權杖)，負載平衡器會聯絡 IdP 以判斷使用者是否仍處於登入狀態。

用戶端登出登陸頁面是未經驗證的頁面。這表示其不能位於需要身分驗證的 Application Load Balancer 規則之後。

- 請求傳送至目標後，應用程式必須將所有身分驗證 Cookie 的到期時間設定為 -1。Application Load Balancer 支援的 Cookie 大小上限為 16K，因此最多會建立 4 個碎片以傳送給用戶端。
 - 如果 IdP 具有登出端點，則應發出一個至 IdP 登出端點 (例如《Amazon Cognito 開發人員指南》中記錄的[登出端點](#)) 的重新導向。
 - 如果 IdP 沒有登出端點，請求會回到用戶端登出登陸頁面，並登入程序會重新啟動。
- 假設 IdP 具有登出端點，則 IdP 必須使存取權杖和重新整理權杖過期，並將使用者重新導向回用戶端登出登陸頁面。
- 後續請求會遵循原始身分驗證流程。

HTTP 標頭和 Application Load Balancer

HTTP 請求和 HTTP 回應使用標頭欄位來傳送有關 HTTP 訊息的資訊。HTTP 標頭會自動新增。標頭欄位是以冒號分隔的名稱值組，以歸位字元 (CR) 和換行 (LF) 分隔。一組以 RFC 2616 定義的標準 HTTP 標頭欄位，[訊息標頭](#)。也有應用程式廣泛採用的非標準 HTTP 標頭可用 (而且會自動新增)。有些非標準 HTTP 標頭擁有 X-Forwarded 字首。Application Load Balancer 支援以下 X-Forwarded 標頭。

如需 HTTP 連線的詳細資訊，請參閱 Elastic Load Balancing 使用者指南中的[請求路由](#)。

X-Forwarded 標頭

- [X-Forwarded-For](#)
- [X-Forwarded-Proto](#)
- [X-Forwarded-Port](#)

X-Forwarded-For

當您使用 HTTP 或 HTTPS 負載平衡器時，X-Forwarded-For 請求標頭會協助您識別用戶端的 IP 地址。由於負載平衡器攔截用戶端和伺服器之間的流量，伺服器存取日誌僅包含負載平衡器的 IP 地址。若要查看用戶端的 IP 地址，請使用 `routing.http.xff_header_processing.mode` 屬性。此屬性可讓您在 Application Load Balancer 將 HTTP 請求傳送至目標之前，修改、保留或移除該請求中的 X-Forwarded-For 標頭。此屬性的可能值為 `append`、`preserve` 和 `remove`。此屬性的預設值為 `append`。

Important

標 X-Forwarded-For 標頭應謹慎使用，因為可能存在安全風險。只有在網路中受到適當保護的系統新增時，才能將這些項目視為值得信賴。

附加

Application Load Balancer 依預設會將用戶端的 IP 地址儲存在 X-Forwarded-For 請求標頭，並將標頭傳遞給伺服器。如果 X-Forwarded-For 請求標頭未包含在原始請求中，負載平衡器會以用戶端 IP 地址作為請求值建立請求標頭。否則，負載平衡器會將用戶端 IP 地址附加至現有的標頭，然後將標頭傳遞至您的伺服器。X-Forwarded-For 請求標頭可能包含以逗號分隔的多個 IP 地址。

X-Forwarded-For 請求標頭採用以下格式：

```
X-Forwarded-For: client-ip-address
```

下列是具有 IP 地址 203.0.113.7 之用戶端的範例 X-Forwarded-For 請求標頭。

```
X-Forwarded-For: 203.0.113.7
```

下列是具有 IPv6 地址 2001:DB8::21f:5bff:febf:ce22:8a2e 之用戶端的範例 X-Forwarded-For 請求標頭。

```
X-Forwarded-For: 2001:DB8::21f:5bff:febf:ce22:8a2e
```

負載平衡器上啟用用戶端連接埠保留屬性 (routing.http.xff_client_port.enabled) 後，X-Forwarded-For 請求標頭會包含在 client-ip-address 附加的 client-port-number (以冒號分隔)。標頭採用以下格式：

```
IPv4 -- X-Forwarded-For: client-ip-address:client-port-number
```

```
IPv6 -- X-Forwarded-For: [client-ip-address]:client-port-number
```

請注意，對於 IPv6，當負載平衡器將 client-ip-address 附加到現有的標頭時，其會以方括號括住該地址。

下列是用戶端 (IPv4 地址為 12.34.56.78，連接埠號碼為 8080) 的 X-Forwarded-For 請求標頭範例。

```
X-Forwarded-For: 12.34.56.78:8080
```

下列是用戶端 (IPv6 地址為 2001:db8:85a3:8d3:1319:8a2e:370:7348，連接埠號碼為 8080) 的 X-Forwarded-For 請求標頭範例。

```
X-Forwarded-For: [2001:db8:85a3:8d3:1319:8a2e:370:7348]:8080
```

保留

屬性中的 preserve 模式可確保 HTTP 請求的 X-Forwarded-For 標頭在傳送到目標之前，請求中不會以任何方式遭到修改。

Remove (移除)

屬性中的 `remove` 模式會在將 HTTP 請求的 `X-Forwarded-For` 標頭在傳送到目標之前將其移除。

Note

如果您啟用用戶端連接埠保留屬性 (`routing.http.xff_client_port.enabled`)，並為 `routing.http.xff_header_processing.mode` 屬性選取 `preserve` 或 `remove`，則 Application Load Balancer 會覆寫用戶端連接埠保留屬性。此屬性可將 `X-Forwarded-For` 標頭保持不變，或者根據您選取的模式將其移除，然後再將標頭傳送到目標。

下表顯示當您選取 `append`、`preserve` 或 `remove` 模式時，目標接收到的 `X-Forwarded-For` 標頭範例。在此範例中，最後一個跳轉的 IP 地址為 `127.0.0.1`。

請求說明	範例請求	<code>append</code> 模式中的 XFF	<code>preserve</code> 模式中的 XFF	<code>remove</code> 模式中的 XFF
傳送的請求沒有 XFF 標頭	GET / index.html HTTP/1.1 Host: example.com	X-Forwarded-For: 127.0.0.1	不存在	不存在
傳送的請求包含 XFF 標頭和用戶端 IP 地址。	GET / index.html HTTP/1.1 Host: example.com X-Forwarded-For: 127.0.0.4	X-Forwarded-For: 127.0.0.4, 127.0.0.1	X-Forwarded-For: 127.0.0.4	不存在
傳送的請求包含 XFF 標頭和多個用戶端 IP 地址。	GET / index.html HTTP/1.1 Host: example.com	X-Forwarded-For: 127.0.0.4, 127.0.0.8, 127.0.0.1	X-Forwarded-For: 127.0.0.4, 127.0.0.8	不存在

請求說明	範例請求	append 模式中的 XFF	preserve 模式中的 XFF	remove 模式中的 XFF
	X-Forwarded-For: 127.0.0.4, 127.0.0.8			

使用主控台修改、保留或移除 X-Forwarded-For 標頭

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在屬性索引標籤中，選擇編輯。
5. 在流量組態區段的封包處理下，在 X-Forwarded-For 標頭中，選擇附加 (預設)、保留或移除。
6. 選擇儲存變更。

若要使用修改、保留或移除 X-Forwarded-For 標頭 AWS CLI

以 [屬性來使用](#) `modify-load-balancer-attributes routing.http.xff_header_processing.mode` 命令。

X-Forwarded-Proto

X-Forwarded-Proto 請求標頭協助您識別用戶端用於連接到您的負載平衡器的通訊協定 (HTTP 或 HTTPS)。您的伺服器存取日誌僅包含在伺服器和負載平衡器之間使用的通訊協定，但不包含用戶端和負載平衡器之間使用的通訊協定相關資訊。若要判斷用戶端和負載平衡器之間使用的通訊協定，請使用 X-Forwarded-Proto 請求標頭。Elastic Load Balancing 會將用戶端和負載平衡器之間使用的通訊協定儲存在 X-Forwarded-Proto 請求標頭，並將標頭傳遞給您的伺服器。

您的應用程式或網站可以使用存放在 X-Forwarded-Proto 請求標頭中的通訊協定，藉以產生重新導向到適當的 URL 的回應。

X-Forwarded-Proto 請求標頭採用以下格式：

```
X-Forwarded-Proto: originatingProtocol
```

以下範例包含適用於從用戶端產生的 X-Forwarded-Proto 請求標頭，以做為 HTTPS 請求：

```
X-Forwarded-Proto: https
```

X-Forwarded-Port

X-Forwarded-Port 請求標頭協助您識別用戶端用於連接到負載平衡器的目的地連接埠。

接聽程式和規則的標籤

標籤可協助您以不同方式來分類接聽程式和規則。例如，您可以依用途、擁有者或環境來標記資源。

您可以將多個標籤新增至每個接聽程式和規則。每個接聽程式和規則的標籤索引鍵必須是唯一的。如果新增的標籤已有與接聽程式和規則建立關聯的索引鍵，則此動作會更新該標籤的值。

當您使用完標籤之後，可以將其移除。

限制

- 每一資源標籤數上限：50
- 索引鍵長度上限：127 個 Unicode 字元
- 數值長度上限：255 個 Unicode 字元
- 標籤金鑰與值皆區分大小寫。允許的字元包括可用 UTF-8 表示的英文字母、空格和數字，還有以下特殊字元：+ - = . _ : / @。不可使用結尾或前方空格。
- 請勿在標籤名稱或值中使用aws:前置詞，因為它已保留供 AWS 使用。您不可編輯或刪除具此字首的標籤名稱或值。具此字首的標籤，不算在受資源限制的標籤計數內。

更新接聽程式標籤

使用主控台來更新接聽程式的標籤

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡器)，選擇 Load Balancer (負載平衡器)。
3. 選擇包含您要更新之接聽程式的負載平衡器名稱，以開啟其詳細資訊頁面。
4. 在接聽程式和規則索引標籤上，執行下列其中一個動作：
 - a. 選取通訊協定：連接埠資料欄中的文字，即可開啟接聽程式的詳細資訊頁面。

- 在 Tags (標籤) 索引標籤上，選擇 Manage tags (管理標籤)。
 - b. 選取您要更新其標籤的接聽程式。

選擇管理接聽程式，然後選擇管理標籤。
 - c. 選取標籤資料欄中的文字，以在標籤索引標籤上開啟接聽程式的詳細資訊頁面。

選擇管理標籤。
5. 在管理標籤頁面上，可以執行下列一個或多個動作：
 - a. 若要更新標籤，請為索引鍵和值輸入新值。
 - b. 如要新增標籤，請選擇新增標籤，然後輸入索引鍵和值的值。
 - c. 若要移除標籤，請選擇標籤旁的移除。
6. 完成標籤的更新作業後，請選擇儲存變更。

使用更新監聽器的標籤 AWS CLI

使用 [add-tags](#) 和 [remove-tags](#) 指令。

更新規則標籤

使用主控台更新規則的標籤

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡器)，選擇 Load Balancer (負載平衡器)。
3. 選擇包含您要更新之規則的負載平衡器名稱，以開啟其詳細資訊頁面。
4. 在接聽程式和規則索引標籤上，在通訊協定：連接埠資料欄中選取內含您要更新之規則的接聽程式的文字，以開啟接聽程式的詳細資訊頁面
5. 在接聽程式詳細資訊頁面上，執行下列其中一個動作：
 - a. 選取名稱標籤資料欄中的文字，以開啟規則的詳細資訊頁面。

在規則詳細資訊頁面上，選擇管理標籤。
 - b. 在標籤資料欄中，選擇您要更新之規則的文字。

在標籤摘要快顯視窗中選擇管理標籤。
6. 在管理標籤頁面上，可以執行下列一個或多個動作：

- a. 若要更新標籤，請為索引鍵和值輸入新值。
 - b. 如要新增標籤，請選擇新增標籤，然後輸入索引鍵和值的值。
 - c. 若要移除標籤，請選擇標籤旁的移除。
7. 完成標籤的更新作業後，請選擇儲存變更。

若要使用更新規則的標籤 AWS CLI

使用 [add-tags](#) 和 [remove-tags](#) 指令。

刪除 Application Load Balancer 的接聽程式

您可隨時刪除接聽程式。當您刪除負載平衡器後，其所有接聽程式將被刪除。

使用主控台刪除接聽程式

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上選擇 Load Balancers (負載平衡器)。
3. 選取負載平衡器。
4. 在接聽程式和規則索引標籤上，選取接聽程式的核取方塊，然後選擇管理接聽程式、刪除接聽程式。
5. 出現確認提示時，請輸入 **confirm**，然後選擇刪除。

使用刪除監聽器 AWS CLI

使用 [delete-listener](#) 命令。

Application Load Balancer 的目標群組

目標群組使用您指定的通訊協定和連接埠號碼將請求路由到單獨已註冊的目標，例如 EC2 執行個體。您可以向多個目標群組註冊任一目標。您可以針對每個目標群組設定運作狀態檢查。凡已註冊至負載平衡器的接聽程式規則中指定之目標群組的所有目標，系統將對其執行運作狀態檢查。

每個目標群組會用來將請求轉送到一個或多個註冊的目標。在建立每個接聽程式規則時，您會指定目標群組和條件。規則的條件符合時，會將流量轉送到對應的目標群組。您可以針對不同類型的請求，建立不同的目標群組。例如，針對一般請求建立一個目標群組，然後再針對應用程式微型服務的請求，建立其他的目標群組。每個目標群組只能搭配一個負載平衡器使用。如需詳細資訊，請參閱 [Application Load Balancer 元件](#)。

您可以針對每個目標群組，指定負載平衡器的運作狀態檢查設定。除非您在建立目標群組時覆寫這些設定，或是在之後修改設定，否則每個目標群組都會使用預設的運作狀態檢查設定。當您在接聽程式的規則中指定目標群組後，負載平衡器會針對自己已啟用可用區域中的目標群組，持續地監控透過該目標群組註冊的所有目標，以了解目標的運作狀態。負載平衡器會將請求路由至運作狀態良好的已註冊目標。

目錄

- [路由組態](#)
- [Target type \(目標類型\)](#)
- [IP 地址類型](#)
- [通訊協定版本](#)
- [已登記的目標](#)
- [目標群組屬性](#)
- [路由算法](#)
- [自動目標權重 \(ATW\)](#)
- [取消登記的延遲](#)
- [慢速啟動模式](#)
- [建立目標群組](#)
- [目標群組運作狀態檢查](#)
- [目標群組的跨區域負載平衡](#)
- [目標群組運作狀態](#)
- [透過目標群組來登記目標](#)
- [Application Load Balancer 的粘性會話](#)

- [Lambda 函數作為目標](#)
- [目標群組的標籤](#)
- [刪除目標群組](#)

路由組態

根據預設，負載平衡器會使用您在建立目標群組時所指定的通訊協定和埠號，來將請求路由至其目標。或者，您可以在使用目標群組來登錄目標時，覆寫用來將流量轉傳到目標的連接埠。

目標群組支援下列的通訊協定和連接埠：

- Protocols (通訊協定)：HTTP、HTTPS
- Ports (連接埠)：1-65535

如果已將目標群組設定為使用 HTTPS 通訊協定或使用 HTTPS 運作狀態檢查，則與目標的 TLS 連線會使用來自 ELBSecurityPolicy-2016-08 政策的安全設定。負載平衡器會使用您在目標上安裝的憑證，與目標建立 TLS 連線。負載平衡器不會驗證這些憑證。因此，您可以使用自我簽署的憑證或已過期的憑證。由於負載平衡器及其目標位於虛擬私有雲 (VPC) 中，因此負載平衡器和目標之間的流量會在封包層級進行驗證，因此即使目標上的憑證無效，也不會受到 man-in-the-middle 攻擊或詐騙的風險。離開的流量 AWS 將不會有這些相同的保護措施，並且可能需要額外的步驟來進一步保護流量。

Target type (目標類型)

建立目標群組時，您會指定其目標類型，這會決定您對此目標群組註冊目標時指定的目標類型。目標群組建立之後，您就無法更改其目標類型。

下列是可能的目標類型：

instance

以執行個體 ID 來指定目標。

ip


目標為 IP 地址。

lambda

目標是 Lambda 函數。

如果目標類型是 `ip`，您可以從下列其中一個 CIDR 區塊指定 IP 地址：


- 目標群組 VPC 的子網路
- 10.0.0.0/8 ([RFC 1918](#))
- 100.64.0.0/10 ([RFC 6598](#))
- 172.16.0.0/12 (RFC 1918)
- 192.168.0.0/16 (RFC 1918)

 Important

您無法指定可公開路由傳送的 IP 地址。

所有支援的 CIDR 區塊都可讓您將下列目標註冊至目標群組：

- 與負載平衡器 VPC (相同區域或不同區域) 對等之 VPC 中的執行個體。
- AWS 可透過 IP 位址和連接埠 (例如資料庫) 定址的資源。
- AWS 透過 AWS Direct Connect Site-to-Site VPN 連線連結至的內部部署資源。

 Note

對於在 Local Zone 內部署的 Application Load Balancer，`ip` 目標必須位於相同的 Local Zone，才能接收流量。

如需詳細資訊，請參閱 [什麼是 Local Zones？](#)

如果使用執行個體 ID 來指定目標，會利用在執行個體的主要網路界面中，所指定的主要私有 IP 地址，來將流量轉送到執行個體。如果使用 IP 地址來指定目標，您可以利用來自一個或多個網路界面的任何私有 IP 地址，將流量轉送到執行個體。這可讓執行個體上的多個應用程式，使用相同的連接埠。每個網路界面都可以有自己的安全群組。

如果目標群組的目標類型是 `lambda`，則可以註冊單一 Lambda 函數。當負載平衡器收到 Lambda 函數的請求時，它會呼叫 Lambda 函數。如需詳細資訊，請參閱 [Lambda 函數作為目標](#)。

您可以將 Amazon Elastic Container Service (Amazon ECS) 設定為 Application Load Balancer 的目標。如需詳細資訊，請參閱 Amazon 彈性容器服務使用者指南中的 [建立應用 Application Load Balancer](#) 器 AWS Fargate。

IP 地址類型

建立新目標群組時，您可以選取目標群組的 IP 地址類型。這會控制用來與目標通訊並檢查目標運作狀態的 IP 版本。

Application Load Balancer 支援 IPv4 和 IPv6 目標群組。預設的選取為 IPv4。

考量事項

- 目標群組中的所有 IP 地址都必須具有相同的 IP 地址類型。例如，您無法在 IPv6 目標群組註冊 IPv4 目標。
- IPv6 目標群組只能與 dualstack 負載平衡器搭配使用。
- IPv6 目標群組支援 IP 和執行個體類型目標。

通訊協定版本

根據預設，Application Load Balancer 會使用 HTTP/1.1 將請求傳送至目標。您可以使用通訊協定版本，使用 HTTP/2 或 gRPC 將請求傳送至目標。

下表摘要說明請求通訊協定與目標群組通訊協定版本組合的結果。

請求通訊協定	通訊協定版本	結果
HTTP/1.1	HTTP/1.1	Success (成功)
HTTP/2	HTTP/1.1	Success (成功)
gRPC	HTTP/1.1	錯誤
HTTP/1.1	HTTP/2	錯誤
HTTP/2	HTTP/2	Success (成功)
gRPC	HTTP/2	如果目標支援 gRPC，則成功

請求通訊協定	通訊協定版本	結果
HTTP/1.1	gRPC	錯誤
HTTP/2	gRPC	如果是 POST 請求，則成功
gRPC	gRPC	Success (成功)

gRPC 通訊協定版本的考量事項

- 唯一支援的接聽程式通訊協定是 HTTPS。
- 接聽程式規則唯一支援的動作類型為 forward。
- 支援的目標類型僅為 instance 和 ip。
- 負載平衡器會剖析 gRPC 請求，並根據套件、服務和方法將 gRPC 呼叫路由至適當的目標群組。
- 負載平衡器支援一元、用戶端串流、伺服器端串流和雙向串流。
- 必須以 `/package.service/method` 格式提供自訂運作狀態檢查方法。
- 必須指定在檢查是否有來自目標的成功回應時要使用的 gRPC 狀態程式碼。
- 無法將 Lambda 函數用作目標。

HTTP/2 通訊協定版本的考量事項

- 唯一支援的接聽程式通訊協定是 HTTPS。
- 接聽程式規則唯一支援的動作類型為 forward。
- 支援的目標類型僅為 instance 和 ip。
- 負載平衡器支援從用戶端進行串流。負載平衡器不支援串流至目標。

已登記的目標

您的負載平衡器可做為用戶端的單一聯絡窗口，並將傳入的流量分配到各個運作狀態良好的已登錄目標。您可以利用一個或多個群組來登錄每個目標。

如果對應用程式的需求增加，您可以利用一個或多個目標群組來登錄額外的目標，來應付需求。當註冊程序完成且目標通過第一個初始健全狀況檢查 (不論設定的臨界值為何)，負載平衡器會立即開始將流量路由到新註冊的目標。

如果對您應用程式的需求減少，或者您需要為目標提供服務，可以從目標群組取消目標的登錄。取消目標的登錄，會將該目標從目標群組中移除，但不會影響到目標。取消目標的註冊之後，負載平衡器就會立即停止將請求路由到目標。目標會進入 draining 狀態，直到處理中的請求已完成。當您準備讓目標再繼續接收請求時，可以將目標註冊到目標群組。

如果是根據執行個體 ID 來註冊目標，您可以使用負載平衡器搭配 Auto Scaling 群組。在將目標群組連接到 Auto Scaling 群組之後，自動擴展會在該群組啟動這些目標時，將目標註冊到目標群組。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling User Guide 中的 [Attaching a load balancer to your Auto Scaling group](#)。

限制

- 您無法在相同的 VPC 中註冊另一個 Application Load Balancer 的 IP 地址。如果另一個 Application Load Balancer 位於與負載平衡器 VPC 對等的 VPC 中，您可以註冊其 IP 地址。
- 如果執行個體位於與負載平衡器 VPC (相同區域或不同區域) 對等的 VPC 中，則您無法依執行個體 ID 註冊執行個體。您可以依照 IP 地址來註冊這些執行個體。

目標群組屬性

如果目標群組類型為 instance 或 ip，則支援以下目標群組屬性：

deregistration_delay.timeout_seconds

取消註冊目標之前，Elastic Load Balancing 要等待的時間量。範圍介於 0–3600 秒之間。預設值為 300 秒。

load_balancing.algorithm.type

負載平衡演算法判斷路由請求時，負載平衡器如何選取目標。值為 round_robinleast_outstanding_requests、或 weighted_random。預設值為 round_robin。

load_balancing.algorithm.anomaly_mitigation

僅當 load_balancing.algorithm.type 是時可用 weighted_random。指出是否已啟用異常緩和措施。此值為 on 或 off。預設值為 off。

load_balancing.cross_zone.enabled

表示是否已啟用跨區域負載平衡。此值為 true、false 或 use_load_balancer_configuration。預設值為 use_load_balancer_configuration。

`slow_start.duration_seconds`

時間期間 (秒)，在此期間負載平衡器會將新註冊的目標流量的線性增加共用傳送至目標群組。此範圍介於 30–900 秒之間 (15 分鐘)。預設值為 0 秒 (已停用)。

`stickiness.enabled`

指出是否已啟用黏性工作階段。此值為 `true` 或 `false`。預設值為 `false`。

`stickiness.app_cookie.cookie_name`

應用程式 Cookie 名稱。應用程式 Cookie 名稱不能有下列字首：AWSALB、AWSALBAPP 或 AWSALBTG；它們預留供負載平衡器使用。

`stickiness.app_cookie.duration_seconds`

應用程式型 Cookie 過期期間 (秒)。在此期間之後，便會將 Cookie 視為過時。最小值為 1 秒，最大值為 7 天 (604800 秒)。預設值為 1 天 (86400 秒)。

`stickiness.lb_cookie.duration_seconds`

持續時間型 Cookie 過期期間 (秒)。在此期間之後，便會將 Cookie 視為過時。最小值為 1 秒，最大值為 7 天 (604800 秒)。預設值為 1 天 (86400 秒)。

`stickiness.type`

黏性的類型。可能的值為 `lb_cookie` 和 `app_cookie`。

`target_group_health.dns_failover.minimum_healthy_targets.count`

運作狀態必須良好的目標數量下限。如果運作狀態良好的目標數量低於此值，請在 DNS 中將區域標記為運作狀態不佳，以便只將流量路由至運作狀態良好的區域。目標可能的值為 `off`，或介於 1 到數目上限的整數。當 `off` 時，會停用 DNS 故障，代表每個目標群組會獨立貢獻 DNS 備援。預設為 1。

`target_group_health.dns_failover.minimum_healthy_targets.percentage`

運作狀態必須良好的目標百分比下限。如果運作狀態良好的目標百分比低於此值，請在 DNS 中將區域標記為運作狀態不佳，以便只將流量路由至運作狀態良好的區域。可能的值為 `off`，或是介於 1 到目標數量上限的整數。當 `off` 時，會停用 DNS 備援，這表示每個目標群組會獨立貢獻 DNS 備援。預設為 1。

`target_group_health.unhealthy_state_routing.minimum_healthy_targets.count`

運作狀態必須良好的目標最低數量。如果運作狀態良好的目標數量低於此值，請將流量傳送至所有目標，包括運作狀態不佳的目標。範圍介於 1 到目標最高數量。預設為 1。

`target_group_health.unhealthy_state_routing.minimum_healthy_targets.percentage`

運作狀態必須良好的目標最低百分比。如果運作狀態良好的目標百分比低於此值，請將流量傳送至所有目標，包括運作狀態不佳的目標。可能的值為 `off`，或介於 1 到 100 之間的整數。預設值為 `off`。

如果目標群組類型為 `lambda`，則支援以下目標群組屬性：

`lambda.multi_value_headers.enabled`

指出負載平衡器與 Lambda 函數之間的請求和回應標頭交換是否包含值或字串的陣列。可能的值為 `true` 或 `false`。預設值為 `false`。如需詳細資訊，請參閱 [多值標頭](#)。

路由算法

路由演算法是負載平衡器在判斷哪些目標將接收要求時所使用的方法。依預設，會使用循環配置資源路由演算法，在目標群組層級路由要求。根據您的應用程式的需求，也可以使用最不突出的請求和加權隨機路由算法。一個目標群組一次只能有一個使用中的路由演算法，但可以視需要更新路由演算法。

如果您啟用黏滯工作階段，則選取的路由演算法會用於初始目標選取。來自相同用戶端的未來要求將會轉送至相同的目標，而略過選取的路由演算法。

循環賽

- 循環配置資源路由演算法會依序將要求平均路由至目標群組中正常狀態的目標。
- 當收到的要求複雜度相似、註冊的目標在處理能力上相似，或者您需要在目標之間平均分配要求時，通常會使用此演算法。

最少未完成的請求

- 未完成的要求路由演算法會將要求路由傳送至進行中要求數目最少的目標。
- 當收到的請求的複雜性不同，註冊的目標處理能力不同時，通常使用此算法。
- 當支援 HTTP/2 的負載平衡器使用僅支援 HTTP/1.1 的目標時，它會將要求轉換成多個 HTTP/1.1 要求。在此配置中，最不處理的請求算法將每個 HTTP/2 請求視為多個請求。
- 使用時 WebSockets，會使用最不未處理的要求演算法選取目標。選取之後，負載平衡器會建立與目標的連線，並透過此連線傳送所有訊息。
- 最不未完成的要求路由演算法不能與慢速啟動模式一起使用。

加權隨機

- 加權隨機路由演算法會以隨機順序在目標群組中的正常狀態目標之間平均路由傳送要求。
- 此演算法支援自動目標權重 (ATW) 異常緩和措施。
- 加權隨機路由演算法不能與慢啟動模式一起使用。

修改目標群組的路由演算法

您可以隨時修改目標群組的路由演算法。

若要使用新主控台修改路由演算法

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在目標群組詳細資訊頁面的屬性標籤上，選擇編輯。
5. 在 [編輯目標群組屬性] 頁面的 [流量設定] 區段的 [負載平衡演算法] 下，選擇 [循環配置資源]、[最少未處理的要求] 或 [加權隨機]。
6. 選擇儲存變更。

若要使用修改路由演算法 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配 `load_balancing.algorithm.type` 屬性。

自動目標權重 (ATW)

自動目標權重 (ATW) 會持續監控執行應用程式的目標，偵測顯著的效能偏差 (稱為異常)。ATW 可透過即時資料異常偵測，動態調整路由至目標的流量量。

自動目標權重 (ATW) 會自動對帳戶中的每個 Application Load Balancer 執行異常偵測。當識別出異常目標時，ATW 可以通過減少路由的流量 (稱為異常緩解) 來自動嘗試穩定它們。ATW 持續最佳化流量分佈，以最大化每個目標的成功率，同時將目標群組失敗率降至最低。

考量：

- 異常偵測目前會監控來自目標的 HTTP 5xx 回應碼，以及目標的連線失敗。異常偵測始終處於開啟狀態且無法關閉。

- 使用 Lambda 做為目標時，不支援 ATW。

異常偵測

ATW 異常偵測會監控行為與其目標群組中其他目標顯示明顯偏差的任何目標。這些偏差 (稱為異常) 是透過比較一個目標的誤差百分比與目標群組中其他目標的誤差百分比來決定的。這些錯誤可以是連接錯誤和 HTTP 錯誤代碼。報告明顯高於同儕的目標會被視為異常狀況。

異常偵測需要目標群組中至少三個健康狀況良好的目標。將目標註冊到目標群組時，必須先通過健全狀況檢查才能開始接收流量。一旦目標接收目標，ATW 就會開始監視目標並持續發佈異常結果。對於沒有異常的目標，異常結果為。normal對於具有異常的目標，異常結果為。anomalous

ATW 異常偵測獨立於目標群組健康狀態檢查。目標可以通過所有目標群組健康狀態檢查，但由於錯誤率提高，仍會標示為異常狀況。異常的目標不會影響其目標群組健康狀態檢查狀態。

異常偵測狀態

ATW 會持續發佈它在目標上執行的異常偵測的狀態。您可以隨時使用 AWS Management Console 或檢視目前狀態 AWS CLI。

使用主控台檢視異常偵測狀態

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在目標群組詳細資訊頁面上，選擇目標頁籤。
5. 在「已註冊的目標」表格中，您可以在「異常偵測結果」欄中檢視每個目標異常狀態。

如果未偵測到異常，則結果為normal。

如果偵測到異常，結果為anomalous。

若要檢視異常偵測結果，請使用 AWS CLI

使用[描述-目標健康](#)指令，並將屬性值設定為Include.member.N. AnomalyDetection

异常缓解

Important

只有在使用加權隨機路由演算法時，才能使用 ATW 的異常緩解功能。

ATW 異常緩解功能會自動將流量路由遠離異常目標，讓他們有機會進行復原。

緩解期間：

- ATW 會定期調整路由到異常目標的流量。目前，該週期是每五秒鐘一次。
- ATW 將路由到異常目標的流量減少到執行異常緩解所需的最小量。
- 不再被偵測為異常的目標，將逐漸有更多流量路由到達目標群組中其他正常目標的同位檢查為止。

開啟 ATW 異常緩解

您可以隨時開啟異常緩解功能。

使用主控台開啟異常緩解

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在目標群組詳細資訊頁面的屬性標籤上，選擇編輯。
5. 在 [編輯目標群組屬性] 頁面的 [流量設定] 區段的 [負載平衡演算法] 下，確定已選取 [加權隨機]。

附註：一開始選取加權隨機演算法時，預設會開啟異常偵測。

6. 在「異常緩解」下，確保已選取「開啟異常緩和措施」。
7. 選擇儲存變更。

若要使用開啟異常緩解 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配

`load_balancing.algorithm.anomaly_mitigation` 屬性。

異常緩解狀態

每當 ATW 對目標執行緩和措施時，您都可以隨時使用 AWS Management Console 或 AWS CLI 來檢視目前的狀態。

使用主控台檢視異常緩解狀態

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在目標群組詳細資訊頁面上，選擇目標頁籤。
5. 在 [已註冊的目標] 表格中，您可以在 [有效的緩和措施] 欄中檢視每個目標異常緩和措施狀態。

如果緩解未在進行中，則狀態為yes。

如果緩解正在進行中，則狀態為no。

若要檢視異常緩解狀態，請使用 AWS CLI

使用[描述-目標健康](#)指令，並將屬性值設定為Include.member.N.AnomalyDetection

取消登記的延遲

Elastic Load Balancing 會停止將請求傳送給正在取消註冊的目標。在預設情況下，Elastic Load Balancing 需要等待 300 秒，才能完成取消註冊程序，這有助於至目標的傳輸中請求完成。若要變更 Elastic Load Balancing 等待的時間，請更新取消註冊延遲時間值。

取消註冊中的目標，其初始狀態為 draining。經過取消註冊延遲之後，取消註冊程序會完成，並且目標的狀態為 unused。如果目標是 Auto Scaling 群組的一部分，可加以終止和取代。

如果取消註冊的目標沒有傳輸中的請求，也沒有作用中連線，Elastic Load Balancing 會立即完成取消註冊程序，而不會等候取消註冊延遲時間結束。不過，即使目標取消註冊完成，目標的狀態仍會顯示為 draining，直到取消註冊延遲逾時結束。逾時結束後，目標會轉變為 unused 狀態。

如果取消註冊目標在取消註冊延遲經過之前終止連線，用戶端會收到 500 層級的錯誤回應。

使用主控台來更新取消登錄的延遲值

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資訊索引標籤的屬性區段中，選擇編輯。
5. 在編輯屬性頁面上，視需要變更取消註冊延遲時間的值。
6. 選擇儲存變更。

若要使用更新取消註冊延遲值 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配 `deregistration_delay.timeout_seconds` 屬性。

慢速啟動模式

在預設情況下，在向目標群組註冊目標並通過初始運作狀態檢查之後，目標隨即會開始接收其請求的完整共用。使用慢速啟動模式可讓目標在負載平衡器向它們傳送請求的完整共用之前，有時間進行暖機。

啟用目標群組的慢啟動後，當目標群組被視為運作狀態良好時，其目標會進入慢速啟動模式。慢速啟動模式的目標，會在設定的慢啟動超過持續的時間或目標變得狀態不良時，結束慢速啟動模式。負載平衡器會線性增加可以在慢速啟動模式中傳送到目標的請求數量。在運作狀態良好的目標退出慢速啟動模式之後，負載平衡器可以將完整份額的請求傳送給它。

考量事項

- 啟用目標群組的慢啟動時，運作狀態良好的已註冊目標不會進入慢速啟動模式。
- 為空白目標群組啟用慢速啟動，然後使用單一註冊操作註冊目標時，這些目標不會進入慢速啟動模式。只有在至少一個運作狀態良好的目標不處於慢速啟動模式時，新註冊的目標才會進入慢速啟動模式。
- 如果您將處於慢速啟動模式的目標取消註冊，該目標會退出慢速啟動模式。如果您再次註冊相同的目標，當目標群組視為運作狀態良好時，它會進入慢速啟動模式。
- 如果處於慢速啟動模式的目標變得運作狀態不佳，目標就會結束慢速啟動模式。當目標狀況良好時，它會再次進入慢速啟動模式。
- 使用最不未處理的要求或加權隨機路由演算法時，您無法啟用慢速啟動模式。

使用主控台更新慢速啟動持續期間值

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資訊索引標籤的屬性區段中，選擇編輯。
5. 在編輯屬性頁面上，視需要變更慢速啟動持續時間的值。若要停用慢速啟動模式，請將持續期間設定為 0。
6. 選擇儲存變更。

若要使用更新慢速啟動持續時間值 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配 `slow_start.duration_seconds` 屬性。

建立目標群組

您會向目標群組註冊您的目標。根據預設，負載平衡器會使用您針對目標群組所指定的埠號和通訊協定，來將請求傳送到登錄的目標。在透過目標群組來註冊每個目標時，您可以覆寫此埠號。

在建立目標群組之後，您可以新增標籤。

若要將流量路由到目標群組中的目標，請在建立接聽程式或為接聽程式建立規則時，於動作中指定目標群組。如需詳細資訊，請參閱 [接聽程式規則](#)。您可以在多個接聽程式中指定相同的目標群組，但這些接聽程式必須屬於相同的 Application Load Balancer。若要將目標群組與負載平衡器搭配使用，您必須確認任何其他負載平衡器的接聽程式未使用此目標群組。

您可以隨時從目標群組新增或移除目標。如需詳細資訊，請參閱 [透過目標群組來登記目標](#)。您也可以修改目標群組的運作狀態檢查設定。如需詳細資訊，請參閱 [修改目標群組的運作狀態檢查設定](#)。

使用主控台來建立目標群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇 Create target group (建立目標群組)。
4. 在選擇目標類型中，選取執行個體可依執行個體 ID 註冊目標，選取 IP 地址可依 IP 地址註冊目標，或選取 Lambda 函數 依 Lambda 函數註冊目標。
5. 針對 Target group name (目標群組名稱)，輸入目標群組的名稱。此名稱在每個帳戶的每個區域中都必須是唯一的，其長度上限為 32 個字元，並且必須僅包含英數字元或連字號，且開頭或結尾不可以是連字號。

6. (選用) 針對 Protocol (通訊協定) 和 Port (連接埠)，視需要修改預設值。
7. 如果目標類型為執行個體或 IP 地址，請選擇 IPv4 或 IPv6 作為 IP 地址類型，否則請跳至下一個步驟。

請注意，只有具有所選 IP 地址類型的目標才能包含在此目標群組中。建立目標群組後，便無法變更 IP 地址類型。

8. 針對 VPC (VPC) 選擇虛擬私有雲端 (VPC)。請注意，對於 IP addresses (IP 地址) 目標類型，可供選擇的 VPC 是支援您在上一個步驟中選擇之 IP address type (IP 地址類型) 的 VPC。
9. (選用) 針對 Protocol version (通訊協定版本)，視需要修改預設值。
10. (選用) 在 Health checks (運作狀態檢查) 區段中，視需要修改預設設定。
11. 如果目標類型為 Lambda 函數，您可以透過選取 Health checks (運作狀態檢查) 區段中的 啟用 (Enable) 來啟用運作狀態檢查。
12. (選用) 新增一個或多個標籤，如下所示：
 - a. 展開 Tags (標籤) 區段。
 - b. 選擇 Add tag (新增標籤)。
 - c. 輸入標籤金鑰和標籤值。
13. 選擇下一步。
14. (選用) 新增一個或多個目標，如下所示：
 - 如果目標類型為執行個體，請選取一個或多個執行個體，輸入一個或多個連接埠，然後選擇包含為下方待處理項目。

注意：執行個體必須具有指派的主要 IPv6 地址，才能在 IPv6 目標群組中註冊。
 - 如果目標類型是 IP 地址，請執行下列動作：
 - a. 從清單中選取網路 VPC，或選擇其他私人 IP 地址。
 - b. 手動輸入 IP 地址，或使用執行個體詳細資料尋找 IP 地址。一次最多可輸入五個 IP 地址。
 - c. 輸入用於將流量路由到指定 IP 地址的連接埠。
 - d. 選擇包含為下方待處理項目。
 - 如果目標類型是 Lambda 函數，請指定單一 Lambda 函數，或省略此步驟，稍後再指定 Lambda 函數。

15. 選擇 Create target group (建立目標群組)。

16. (選擇性) 您可以在接聽程式規則中指定目標群組。如需詳細資訊，請參閱[接聽程式規則](#)。

若要使用建立目標群組 AWS CLI

使用 [create-target-group](#) 指令來建立目標群組、使用 [add-tags](#) 指令來標記目標群組、使用 [register-targets](#) 指令來新增目標。

目標群組運作狀態檢查

Application Load Balancer 會定期將請求傳送到已註冊的目標來測試其狀態。這些測試稱為運作狀況檢查。

每個負載平衡器節點只會將請求路由至負載平衡器已啟用可用區域內運作狀態良好的目標。每個負載平衡器節點會使用各目標註冊所屬目標群組的運作狀態檢查設定，檢查該目標的運作狀態。目標註冊後，必須通過一次運作狀態檢查，才算運作狀態良好。每次運作狀態檢查完成後，負載平衡器節點即會關閉其為執行運作狀態檢查而建立的連線。

如果目標群組只包含運作狀態不佳的已註冊目標，負載平衡器會將請求路由至所有這些目標，而不論這些目標的運作狀態為何。這表示如果所有已啟用可用區域中的所有目標同時都未通過運作狀態檢查，則負載平衡器會故障開啟。故障開啟的結果是系統會根據負載平衡演算法，允許流量傳輸到所有已啟用可用區域中的所有目標，無論這些目標的運作狀態為何。

Health 狀態檢查不支援 WebSockets。

運作狀態檢查設定

您需要按下表中的描述為目標群組中的目標設定運作狀態檢查。表中使用的設定名稱是 API 中使用的名稱。負載平衡器會使用指定的連接埠、通訊協定和健全狀況檢查路徑，每 `HealthCheckIntervalSeconds` 秒傳送健康狀態檢查要求至每個已註冊的目標。每個運作狀態檢查請求各自獨立，且在整個間隔內持續保持此結果。目標回應所花的時間不影響下次運作狀態檢查請求的間隔。如果健全狀況檢查超過 `UnhealthyThresholdCount` 連續的失敗，負載平衡器會將目標停止服務。當健全狀況檢查超過 `HealthyThresholdCount` 連續成功時，負載平衡器會將目標重新啟用。

設定	描述
<code>HealthCheckProtocol</code>	負載平衡器對目標執行運作狀態檢查時使用的通訊協定。可能的通訊協定是 HTTP 和 HTTPS。預設為 HTTP 通訊協定。 這些通訊協定會使用 HTTP GET 方法，來傳送運作狀態檢查請求。

設定	描述
HealthCheckPort	負載平衡器對目標執行運作狀態檢查時使用的連接埠。預設為使用每個目標從負載平衡器接收流量的連接埠。
HealthCheckPath	<p>目標上運作狀態檢查的目的地。</p> <p>如果通訊協定版本是 HTTP/1.1 或 HTTP/2，請指定有效的 URI (/path?query)。預設為 /。</p> <p>如果通訊協定版本是 gRPC，則使用 /package.service/method 格式指定自訂運作狀態檢查方法的路徑。預設值為 /AWS.ALB/healthcheck。</p>
HealthCheckTimeoutSeconds	以秒為單位的時間量，若目標在此期間內毫無回應即表示運作狀態檢查失敗。範圍介於 2 到 120 秒之間。如果目標類型是 instance 或 ip，則預設為 5 秒，如果是 lambda，則預設為 30 秒。
HealthCheckIntervalSeconds	個別目標每次執行運作狀態檢查的大約間隔時間量，以秒為單位。範圍介於 5–300 秒之間。如果目標類型是 instance 或 ip，則預設為 30 秒，如果是 lambda，則預設為 35 秒。
HealthyThresholdCount	將運作狀態不佳的目標視為運作狀態良好之前，運作狀態檢查需連續成功的次數。範圍介於 2–10 之間。預設值為 5。
UnhealthyThresholdCount	將目標視為運作狀態不佳之前，運作狀態檢查需連續失敗的次數。範圍介於 2–10 之間。預設為 2。

設定	描述
Matcher	<p>檢查是否收到來自目標的成功回應時所使用的代碼。這些在主控台中稱為成功代碼。</p> <p>如果通訊協定版本是 HTTP/1.1 或 HTTP/2，則值範圍是 200 到 499。您可以指定多個值 (例如，"200,202") 或值範圍 (例如，"200-299")。預設值為 200。</p> <p>如果通訊協定版本是 gRPC，則值範圍是 0 到 99。您可以指定多個值 (例如，"0,1") 或值範圍 (例如，"0-5")。預設值為 12。</p>

目標運作狀態

在負載平衡器向目標傳送運作狀態檢查請求之前，您必須向目標群組註冊該目標，由接聽程式規則中指定其目標群組，並確保負載平衡器已啟用該目標的可用區域。目標必須通過初次運作狀態檢查，才能從負載平衡器收到請求。在目標通過初次運作狀態檢查後，它的狀態是 Healthy。

下表說明已註冊目標的運作狀態可能的值。

Value	描述
initial	<p>負載平衡器正在註冊目標或對目標執行初始運作狀態檢查。</p> <p>相關原因代碼：Elb.RegistrationInProgress Elb.InitialHealthChecking</p>
healthy	<p>目標的運作狀態良好。</p> <p>相關原因代碼：無</p>
unhealthy	<p>目標未回應運作狀態檢查或未通過運作狀態檢查。</p>

Value	描述
	相關原因碼：Target.ResponseCodeMismatch Target.Timeout Target.FailedHealthChecks Elb.InternalError
unused	目標未向目標群組註冊、未在接聽程式規則中使用目標群組、目標位於未啟用的可用區域，或目標處於已停止或已終止狀態。 相關原因碼：Target.NotRegistered Target.NotInUse Target.InvalidState Target.IpUnusable
draining	目標正在取消註冊，連接耗盡作業進行中。 相關原因碼：Target.DeregistrationInProgress
unavailable	目標群組的運作狀態檢查已停用。 相關原因碼：Target.HealthCheckDisabled

運作狀態檢查原因代碼

如果目標的狀態是 Healthy 以外的任何值，API 將傳回問題的原因代碼和描述，而且主控台會顯示同樣的描述。開頭為 Elb 的原因代碼源自負載平衡器端，而開頭為 Target 的原因代碼源自目標端。如需運作狀態檢查失敗可能原因的詳細資訊，請參閱 [Troubleshooting](#)。

原因代碼	描述
Elb.InitialHealthChecking	初始運作狀態檢查正進行中
Elb.InternalError	運作狀態檢查由於內部錯誤而失敗
Elb.RegistrationInProgress	目標註冊正進行中

原因代碼	描述
Target.DeregistrationInProgress	目標取消註冊正進行中
Target.FailedHealthChecks	運作狀態檢查失敗
Target.HealthCheckDisabled	運作狀態檢查已停用
Target.InvalidState	目標處於停止狀態 目標處於終止狀態 目標處於終止或停止狀態 目標處於無效狀態
Target.IpUnusable	IP 地址不能做為目標，因為負載平衡器正在使用它
Target.NotInUse	目標群組未設定為接收來自負載平衡器的流量 目標位於負載平衡器未啟用的可用區域
Target.NotRegistered	目標未向目標群組註冊
Target.ResponseCodeMismatch	運作狀態檢查失敗，顯示以下代碼：[code]
Target.Timeout	請求逾時

檢查目標的運作狀態

您可以檢查已向目標群組註冊的各個目標的運作狀態。

使用主控台檢查目標的運作狀態

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。

4. 在 Targets (目標) 標籤上，Status (狀態) 欄指出各目標的狀態。
5. 如果狀態是 Healthy 以外的任何值，則狀態詳細資料欄會包含更多資訊。如需有關運作狀態檢查失敗的說明，請參閱 [Troubleshooting](#)。

使用檢查目標的健康狀態 AWS CLI

使用 [describe-target-health](#) 命令。此命令的輸出包含目標的運作狀態。如果狀態為 Healthy 以外的任何值，則輸出也會包含原因代碼。

接收有關狀態不良目標的電子郵件通知

使用 CloudWatch 警示觸發 Lambda 函數，以傳送有關健康狀態不良目標的詳細資訊。如需 step-by-step 指示，請參閱下列部落格文章：[識別負載平衡器運作狀況不良的目標](#)。

修改目標群組的運作狀態檢查設定

您可以隨時修改目標群組的運作狀態檢查設定。

使用主控台修改目標群組的運作狀態檢查設定

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資料索引標籤的運作狀態檢查設定區段中，選擇編輯。
5. 在編輯運作狀態檢查設定頁面上，視需要修改設定，然後選擇儲存變更。

使用修改目標群組的健全狀況檢查設定 AWS CLI

使用 [modify-target-group](#) 命令。

目標群組的跨區域負載平衡

負載平衡器的節點會將請求從用戶端分發到已註冊的目標。跨區域負載平衡開啟後，每個負載平衡器節點會將流量分散到所有已註冊可用區域內的已註冊目標。跨區域負載平衡關閉後，每個負載平衡器節點只會將流量分散到其可用區域內已註冊的目標。如果區域故障網域優先於地區故障網域，就可能發生此情形，以確保運作狀態良好的區域不受運作狀態不佳區域的影響，也可能是為改善整體延遲。

使用 Application Load Balancer 時，跨區域負載平衡一律會在負載平衡器層級開啟，而且無法關閉。對於目標群組，預設設定為使用負載平衡器設定，但您可以在目標群組層級明確關閉跨區域負載平衡來覆寫預設設定。

考量事項

- 跨區域負載平衡關閉後，不支援目標粘性。
- 跨區域負載平衡關閉後，不支援將 Lambda 函數作為目標。
- 如果有任何目標的參數 AvailabilityZone 設定為 all，則嘗試透過 ModifyTargetGroupAttributes API 關閉跨區域負載平衡會導致錯誤。
- 註冊目標時需要 AvailabilityZone 參數。跨區域負載平衡關閉後，僅允許特定可用區域值。否則，系統會忽略此參數並將其當作 all。

最佳實務

- 針對您預期使用的所有可用區域，為每個目標群組規劃足夠的目標容量。如果無法為所有參與的可用區域規劃足夠容量，建議將跨區域負載平衡保持開啟的狀態。
- 如果 Application Load Balancer 設定有多個目標群組，請確定所有目標群組都在設定的區域內參與相同的可用區域。這是為了避免跨區域負載平衡關閉後可用區域變成空白的狀態，因為這會對進入空白可用區域的所有 HTTP 請求觸發 503 錯誤。
- 避免建立空白子網路。Application Load Balancer 會透過 DNS 公開空白子網路的區域 IP 地址，這會針對 HTTP 請求觸發 503 錯誤。
- 還可能會發生這種情況：關閉跨區域負載平衡的目標群組為每個可用區域都計劃有足夠的目標容量，但可用區域中的所有目標都變成運作狀態不佳。當至少一個目標群組包含的所有目標都運作狀態不佳時，就會將負載平衡器節點的 IP 地址從 DNS 移除。當目標群組具有至少一個運作狀態良好的目標之後，IP 地址就會還原到 DNS。

關閉跨區域負載平衡

您可以隨時為 Application Load Balancer 目標群組關閉跨區域負載平衡。

使用主控台關閉跨區域負載平衡

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡下，選取目標群組。
3. 選取目標群組的名稱，以開啟其詳細資訊頁面。

4. 在屬性索引標籤上，選取編輯。
5. 在編輯目標群組屬性頁面上，針對跨區域負載平衡選取關閉。
6. 選擇儲存變更。

使用 AWS CLI 關閉跨區域負載平衡

使用 [modify-target-group-attributes](#) 命令並將 `load_balancing.cross_zone.enabled` 屬性設為 `false`。

```
aws elbv2 modify-target-group-attributes --target-group-arn my-targetgroup-arn --attributes Key=load_balancing.cross_zone.enabled,Value=false
```

以下是回應範例：

```
{
  "Attributes": [
    {
      "Key": "load_balancing.cross_zone.enabled",
      "Value": "false"
    },
  ]
}
```

開啟跨區域負載平衡

您可以隨時為 Application Load Balancer 目標群組開啟跨區域負載平衡。目標群組層級的跨區域負載平衡設定會覆寫負載平衡器層級的設定。

使用主控台開啟跨區域負載平衡

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡下，選取目標群組。
3. 選取目標群組的名稱，以開啟其詳細資訊頁面。
4. 在屬性索引標籤上，選取編輯。
5. 在編輯目標群組屬性頁面上，針對跨區域負載平衡選取開啟。
6. 選擇儲存變更。

使用 AWS CLI 開啟跨區域負載平衡

使用 [modify-target-group-attributes](#) 命令並將 `load_balancing.cross_zone.enabled` 屬性設為 `true`。

```
aws elbv2 modify-target-group-attributes --target-group-arn my-targetgroup-arn --attributes Key=load_balancing.cross_zone.enabled,Value=true
```

以下是回應範例：

```
{
  "Attributes": [
    {
      "Key": "load_balancing.cross_zone.enabled",
      "Value": "true"
    },
  ]
}
```

目標群組運作狀態

依預設，只要目標群組至少有一個運作狀態良好的目標，就會被視為運作狀態良好。如果您擁有龐大的機群，則只有一個運作狀態良好的目標服務流量是不夠的。相反地，您可以指定必須為運作狀態良好的目標最小計數或百分比，以及當運作狀態良好目標低於指定臨界值時，負載平衡器會採取哪些動作。這提高了可用性。

運作運作狀態不佳

您可以針對下列動作設定運作狀態良好的臨界值：

- **DNS 備援** — 當區域中運作狀態良好的目標低於閾值時，我們會在 DNS 中將區域的負載平衡器節點 IP 地址標記為運作狀態不佳。因此，當用戶端解析負載平衡器 DNS 名稱時，流量只會路由至運作狀態良好的區域。
- **路由容錯移轉** — 當區域中運作狀態良好的目標低於臨界值時，負載平衡器會將流量傳送至負載平衡器節點可用的所有目標，包括運作狀態不良的目標。這會增加用戶端連線成功的機會，尤其是當目標暫時無法通過運作狀態檢查時，並降低運作狀態良好目標超載的風險。

需求和考量事項

- 您無法在目標為 Lambda 函數的目標群組中使用此功能。如果 Application Load Balancer 是 Network Load Balancer 或 Global Accelerator 的目標，請勿設定 DNS 備援的閾值。
- 如果您為動作指定兩種類型的閾值 (計數和百分比)，則當違反任一閾值時，負載平衡器都會採取動作。
- 如果您指定這兩個動作的臨界值，DNS 備援的臨界值必須大於或等於路由容錯移轉的臨界值，以便 DNS 備援發生在路由容錯移轉或之前。
- 如果您將臨界值指定為百分比，我們會根據向目標群組註冊的目標總數來動態計算值。
- 目標總數取決於是關閉還是開啟跨區域負載平衡。如果關閉跨區域負載平衡，則每個節點只會將流量傳送到其自身區域中的目標，這代表臨界值會分別套用至每個已啟用區域中的目標數目。如果跨區域負載平衡是開啟狀態，則每個節點會將流量傳送到所有已啟用區域中的所有目標，這代表會對所有已啟用區域中的目標總數套用此閾值。
- 透過 DNS 備援，我們會從負載平衡器的 DNS 主機名稱中移除運作狀態不佳區域的 IP 地址。不過，本機用戶端 DNS 快取可能會包含這些 IP 位址，直到 DNS 記錄中的 time-to-live (TTL) 到期 (60 秒) 為止。
- 發生 DNS 備援時，這會影響與負載平衡器關聯的所有目標群組。確保剩餘區域中有足夠的容量來處理這些額外的流量，尤其是在跨區域負載平衡關閉的情況下。
- 使用 DNS 備援時，如果將所有負載平衡器區域視為運作狀態不佳，負載平衡器會將流量傳送到所有區域，包括運作狀態不佳的區域。
- 除了是否有足夠運作狀態良好的目標可能導致 DNS 備援之外，還有其他因素，例如區域的運作狀況。

監控

若要監視目標群組的健全狀況，請參閱目標群組[健全狀況的CloudWatch 指標](#)。

範例

以下範例示範如何套用目標群組運作狀態設定。

案例

- 支援 A 和 B 兩個可用區域的負載平衡器
- 每個可用區域包含 10 個已註冊目標

- 目標群組具有下列目標群組運作狀態設定：
 - DNS 備援 - 50%
 - 路由容錯移轉 - 50%
- 可用區域 B 中有六個目標失敗

如果停用跨區域負載平衡

- 每個可用區域中的負載平衡器節點只能將流量傳送到其可用區域中的 10 個目標。
- 可用區域 A 中有 10 個運作狀態良好的目標，符合運作狀態目標的必要百分比。負載平衡器會繼續在 10 個運作狀態良好的目標之間分配流量。
- 可用區域 B 中只有 4 個運作狀態良好的目標，這是可用區域 B 中負載平衡器節點目標的 40%，因為小於運作狀態良好目標的必要百分比，所以負載平衡器會採取下列動作：
 - DNS 備援 - 可用性區域 B 在 DNS 中標示為運作狀態不良。由於用戶端無法將負載平衡器名稱解析為可用區域 B 中的負載平衡器節點，且可用區域 A 運作狀態良好，因此用戶端會將新的連線傳送至可用區域 A。
 - 路由容錯移轉 - 當新連線明確傳送至可用區域 B 時，負載平衡器會將流量分配給可用性區域 B 中的所有目標，包括運作狀態不佳的目標。這樣可以防止剩餘運作狀態良好的目標中斷。

如果啟用跨區域負載平衡

- 每個負載平衡器節點都可以將流量傳送到兩個可用區域的所有 20 個已註冊目標。
- 可用區域 A 中有 10 個運作狀態良好的目標，而可用區域 B 中有 4 個運作狀態良好的目標，總共有 14 個運作狀態良好目標。這是兩個可用區域中負載平衡器節點目標的 70%，符合運作狀態良好目標的必要百分比。
- 負載平衡器會在兩個可用區域中 14 個運作狀況良好的目標之間分配流量。

修改目標群組運作狀態良好設定

您可以依照下列方式修改目標群組的目標群組運作狀況設定。

若要使用主控台修改目標群組的運作狀態設定

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的負載平衡中，選擇目標群組。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。

4. 在屬性索引標籤中，選擇編輯。
5. 檢查是否開啟或關閉跨區域負載平衡。視需要更新此設定，以確保您有足夠的容量可在區域發生故障時處理額外的流量。
6. 展開目標群組運作狀況需求。
7. 針對組態類型，建議您選擇統一組態，這兩個動作都會設定相同的臨界值。
8. 對於狀態良好的狀態要求，請執行下列其中一項：
 - 選擇最小運作狀況目標計數，然後輸入從 1 到目標群組目標數目上限的數字。
 - 選擇最小狀態良好目標百分比，然後輸入 1 到 100 之間的數字。
9. 選擇儲存變更。

使用修改目標群組健全設定 AWS CLI

使用 [modify-target-group-attributes](#) 指令。下列範例會將兩個運作狀態不佳的動作的運作狀態良好閾值設定為 50%。

```
aws elbv2 modify-target-group-attributes \  
--target-group-arn arn:aws:elasticloadbalancing:region:123456789012:targetgroup/my-  
targets/73e2d6bc24d8a067 \  
--attributes  
Key=target_group_health.dns_failover.minimum_healthy_targets.percentage,Value=50 \  
  
Key=target_group_health.unhealthy_state_routing.minimum_healthy_targets.percentage,Value=50
```

針對您的負載平衡器使用 Route 53 DNS 備援

如果您使用 Route 53 將 DNS 查詢路由傳送到負載平衡器，您也可以使用 Route 53 設定負載平衡器的 DNS 備援。在容錯移轉組態中，Route 53 會檢查負載平衡器的目標群組目標的運作狀態，以判斷是否可用。如果沒有負載平衡器註冊的狀態良好目標，或者負載平衡器本身運作狀態不佳，Route 53 會將流量路由到另一可用資源，例如運作狀態良好的負載平衡器或 Amazon S3 中的靜態網站。

例如，假設您有一個 `www.example.com` Web 應用程式，而且您需要在後方執行兩個負載平衡器備援執行個體，位於不同的區域。您希望流量在一個區域主要路由到負載平衡器，而且您想要在其他區域使用負載平衡器，以供失敗時備份。如果您設定 DNS 容錯移轉，您可以指定您的主要和次要 (備份) 負載平衡器。Route 53 會引導流量到可用的主要負載平衡器，或是次要負載平衡器。

使用「評估目標運作狀態」

- 如果 Application Load Balancer 別名記錄上的「評估目標運作狀態」設定為 Yes，Route 53 會評估 alias target 值所指定資源的運作狀態。如果是 Application Load Balancer，Route 53 會使用與負載平衡器關聯的目標群組運作狀態檢查。
- 如果 Application Load Balancer 中所有的目標群組運作狀態都是良好，Route 53 會將別名記錄標記為運作狀態良好。如果目標群組包含至少一個運作狀態良好的目標，則目標群組的運作狀態檢查會通過。之後，Route 53 會根據您的路由政策傳回記錄。如果使用容錯移轉路由政策，則 Route 53 會傳回主要記錄。
- 如果 Application Load Balancer 中有任何目標群組運作狀態不佳，則別名記錄不會通過 Route 53 運作狀態檢查 (故障開啟)。如果使用「評估目標運作狀態」，這將使容錯移轉路由政策失敗。
- 如果 Application Load Balancer 中的所有目標群組都是空的 (沒有目標)，則 Route 53 會將記錄視為運作狀態不佳 (故障開啟)。如果使用「評估目標運作狀態」，這將使容錯移轉路由政策失敗。

如需詳細資訊，請參閱 Amazon Route 53 開發人員指南中的[設定 DNS 容錯移轉](#)。

透過目標群組來登記目標

您會向目標群組註冊您的目標。建立目標群組時，您會指定其目標類型，這會決定您目標的註冊方式。例如，您可以註冊執行個體 ID、IP 地址或 Lambda 函數。如需詳細資訊，請參閱 [Application Load Balancer 的目標群組](#)。

如果對目前已註冊目標的需求增加，您可以註冊額外的目標來應付需求。當目標準備好處理請求時，請透過目標群組來註冊目標。在註冊程序完成、目標通過初始的運作狀態檢查之後，負載平衡器就會立即開始將請求轉送到目標。

如果對已註冊目標的需求減少，或是需要為目標提供服務，您可以從目標群組取消目標的註冊。取消目標的註冊之後，負載平衡器就會立即停止將請求轉送到目標。當目標準備好接收請求時，您可以再次將目標註冊到目標群組。

當您取消註冊目標時，負載平衡器會等到傳輸中的請求完成。這稱為連接耗盡。當連接耗盡作業正在進行時，目標的狀態是 draining。

取消註冊透過 IP 地址註冊的目標時，您必須等待取消註冊延遲完成，之後才能再次註冊相同的 IP 地址。

如果是根據執行個體 ID 來註冊目標，您可以使用負載平衡器搭配 Auto Scaling 群組。在將目標群組連接到 Auto Scaling 群組，而且群組擴展之後，由 Auto Scaling 群組啟動的執行個體會自動註冊到目

標群組。如果分離目標群組與 Auto Scaling 群組的連結，會自動從該目標群組中取消註冊執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling User Guide 中的 [Attaching a load balancer to your Auto Scaling group](#)。

目標安全群組

當您將 EC2 執行個體註冊為目標時，必須確定執行個體的安全群組，會允許負載平衡器同時透過接聽程式連接埠和運作狀態檢查連接埠與您的執行個體通訊。

建議的規則

Inbound

Source	Port Range	Comment
#####	#####	允許來自負載平衡器在執行個體接聽程式連接埠上的流量
#####	#####	允許負載平衡器透過運作狀態檢查連接埠傳送的流量

我們也建議您允許傳入 ICMP 流量，以支援路徑 MTU 探索。如需詳細資訊，請參閱 Amazon EC2 使用者指南中的 [路徑 MTU 探索](#)。

共用子網路

參與者可以在共用 VPC 中建立 Application Load Balancer。參與者無法註冊在未與他們共用的子網路中執行的目標。

登記和取消登記目標

目標群組的目標類型會決定您向該目標群組註冊目標的方式。如需詳細資訊，請參閱 [Target type \(目標類型\)](#)。

目錄

- [根據執行個體 ID 來登記或取消登記目標](#)
- [根據 IP 地址來登記或取消登記目標](#)
- [註冊或取消註冊 Lambda 函數](#)
- [使用 AWS CLI 來登記或取消登記目標](#)

根據執行個體 ID 來登記或取消登記目標

Note

依執行個體 ID 註冊 IPv6 目標群組的目標時，目標必須具有指派的主要 IPv6 地址。若要進一步了解，請參閱 Amazon EC2 使用者指南中的 [IPv6 地址](#)

執行個體必須位在您為目標群組指定的虛擬私有雲端 (VPC)。在註冊時，執行個體也必須處於 running 狀態。

使用主控台根據執行個體 ID 來註冊或取消註冊目標

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 選擇 Targets (目標) 標籤。
5. 若要註冊執行個體，請選擇註冊目標。選取一或多個執行個體，視需要輸入預設執行個體連接埠，然後選擇包含為以下待定的項目。完成執行個體新增時，請選擇註冊待處理的目標。

請注意：

- 執行個體必須具有指派的主要 IPv6 地址，才能向 IPv6 目標群組註冊。
 - AWS GovCloud (US) Region 不支援使用主控台指派主要 IPv6 地址。您必須使用 API 來指派中的主要 IPv6 AWS GovCloud (US) Region 位址。
6. 若要取消註冊執行個體，請選取執行個體，然後選擇取消註冊。

根據 IP 地址來登記或取消登記目標

IPv4 目標

您註冊的 IP 地址必須來自下列 CIDR 區塊：

- 目標群組 VPC 的子網路
- 10.0.0.0/8 (RFC 1918)
- 100.64.0.0/10 (RFC 6598)
- 172.16.0.0/12 (RFC 1918)

- 192.168.0.0/16 (RFC 1918)

您無法在相同的 VPC 中註冊另一個 Application Load Balancer 的 IP 地址。如果另一個 Application Load Balancer 位於與負載平衡器 VPC 對等的 VPC 中，您可以註冊其 IP 地址。

IPv6 目標

- 您註冊的 IP 地址必須位於 VPC CIDR 區塊內或位於對等的 VPC CIDR 區塊內。

使用主控台根據 IP 地址來註冊或取消註冊目標

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 選擇 Targets (目標) 標籤。
5. 若要註冊 IP 地址，請選擇註冊目標。為每個 IP 地址選取網路，輸入 IP 地址和連接埠，然後選擇包含為下方待處理項目。完成指定地址的動作後，請選擇註冊待處理的目標。
6. 若要取消註冊 IP 地址，請選取 IP 地址，然後選擇取消註冊。如果您擁有多個已登錄的 IP 地址，新增篩選條件或變更排序順序，可能會很有幫助。

註冊或取消註冊 Lambda 函數

您可以在每個目標群組中註冊單一 Lambda 函數。Elastic Load Balancing 必須具有調用 Lambda 函數的許可。如果您不再需要將流量傳送到您的 Lambda 函數，則可以將它取消註冊。取消註冊 Lambda 函數之後，傳輸中的請求會失敗，出現 HTTP 5XX 錯誤。若要取代 Lambda 函數，最好是改為建立新的目標群組。如需詳細資訊，請參閱 [Lambda 函數作為目標](#)。

使用主控台註冊或取消註冊 Lambda 函數

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 選擇 Targets (目標) 標籤。
5. 如果尚未註冊 Lambda 函數，請選擇 Register (註冊)。選取 Lambda 函數，然後選擇 Register (註冊)。

- 若要取消註冊 Lambda 函數，請選擇 Deregister (取消註冊)。出現確認的提示時，請選擇取消註冊。

使用 AWS CLI來登記或取消登記目標

使用 [register-targets](#) 指令來新增目標；使用 [deregister-targets](#) 指令來移除目標。

Application Load Balancer 的粘性會話

根據預設，Application Load Balancer 會根據所選的負載平衡演算法，將每個請求獨立路由至註冊的目標。不過，您可以使用粘性會話功能 (也稱為工作階段親和性)，讓負載平衡器將使用者的工作階段繫結到特定目標。這樣能確保該工作階段期間所有的使用者請求都能傳送到同一個目標。這功能對於維護狀態資訊以便為用戶端提供持續體驗的伺服器來說很實用。若要使用粘性會話，用戶端必須支援 Cookie。

Application Load Balancer 支援持續時間型 Cookie 和應用程式型 Cookie。粘性會話會在目標群組層級啟用。您可以組合使用持續時間型粘性、應用程式型粘性，以及目標群體之間無粘性。

管理粘性會話的金鑰是決定您的負載平衡器應該持續將使用者請求路由到同一個目標的時間。如果您的應用程式有自己的工作階段 Cookie，則您可以使用應用程式型粘性，負載平衡器工作階段 Cookie 會遵循應用程式的工作階段 Cookie 指定的持續時間。如果您的應用程式沒有自己的工作階段 Cookie，則您可以使用持續時間型粘性，來產生具有指定持續時間的負載平衡器工作階段 Cookie。

系統會使用輪換金鑰來對負載平衡器產生的 Cookie 內容進行加密。您不能解密或修改負載平衡器產生的 Cookie。

對於這兩種粘性類型，Application Load Balancer 會在每次請求後重設其產生的 Cookie 到期時間。如果 Cookie 過期，則工作階段不再具有粘性，用戶端應該從其 Cookie 存放區中刪除該 Cookie。

要求

- HTTP/HTTPS 負載平衡器。
- 在各個可用區域內啟動至少一個正常運作的執行個體。

考量事項

- 如果 [跨區域負載平衡已停用](#)，便不支援粘性會話。在跨區域負載平衡已停用時，啟用粘性會話的嘗試會失敗。

- 對於應用程式型 Cookie，必須針對每個目標群組個別指定 Cookie 名稱。但是，對於持續時間型 Cookie，AWSALB 是所有目標群組中唯一使用的名稱。
- 如果您使用多層 Application Load Balancer，則可以使用應用程式型 Cookie 在所有層級間啟用粘性會話。但是，如果使用持續時間型 Cookie，便只能在一個層上啟用粘性會話，因為 AWSALB 是唯一可用的名稱。
- 應用程式型粘性不適用於加權目標群組。
- 如果您有一個轉送規則涉及多個目標群組，且一個或多個目標群組已啟用粘性會話，則您必須啟用目標群組層級的粘性。
- WebSocket 連接本質上是粘性的。如果用戶端要求連線升級至 WebSockets，則傳回 HTTP 101 狀態碼以接受連線升級的目標就是連線中使用的 WebSockets 目標。WebSockets 升級完成之後，就不會使用以 Cookie 為基礎的黏性。
- Application Load Balancer 會使用 Cookie 標頭中的 Expires 屬性，而非 Max-Age 屬性。
- Application Load Balancer 不支援 URL 編碼的 Cookie 值。

持續時間型粘性

持續時間型粘性會使用負載平衡器產生的 Cookie (AWSALB)，將請求路由至目標群組中的同一個目標。Cookie 用於將工作階段映射到目標。如果您的應用程式沒有自己的工作階段 Cookie，您可以指定自己的粘性持續時間，並管理負載平衡器應持續地將使用者的請求路由至同一個目標的時間。

負載平衡器首次從用戶端收到請求時，它會將請求路由到目標 (根據所選演算法)，並產生一個名為 AWSALB 的 Cookie。它會對所選目標的資訊進行編碼，對 Cookie 進行加密，並在對用戶端的回應中包含 Cookie。負載平衡器產生的 Cookie 自己有 7 天的有效期，且無法設定。

在後續的請求中，用戶端應該包括 AWSALB Cookie。負載平衡器收到來自用戶端包含 Cookie 的請求時，會偵測到此 Cookie，並會將該請求路由至同一個目標。如果 Cookie 存在但無法解碼，或者它參照了已取消註冊或狀況不良的目標，負載平衡器會選取新的目標，並以新目標的相關資訊更新 Cookie。

對於跨源資源共享 (CORS) 請求，某些瀏覽器需 SameSite=None; Secure 要啟用粘性。為了支持這些瀏覽器，負載平衡器始終生成第二個粘性 cookie AWSALBCORS，其中包含與原始粘性 cookie 相同的信息以及屬性。SameSite 客戶端會收到兩個 cookie，包括非 CORS 請求。

使用主控台啟用持續時間型粘性

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資訊索引標籤的屬性區段中，選擇編輯。
5. 在 Edit attributes (編輯屬性) 頁面上，執行下列動作：
 - a. 選取粘性。
 - b. 在粘性類型中，選取負載平衡器產生的 Cookie。
 - c. 針對 Stickiness duration (黏性持續期間)，指定介於 1 秒到 7 天之間的值。
 - d. 選擇儲存變更。

若要啟用以持續時間為基礎的黏性，請使用 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配 `stickiness.enabled` 和 `stickiness.lb_cookie.duration_seconds` 屬性。

請使用下列命令啟用持續時間型粘性。

```
aws elbv2 modify-target-group-attributes --target-group-arn ARN --attributes
  Key=stickiness.enabled,Value=true
  Key=stickiness.lb_cookie.duration_seconds,Value=time-in-seconds
```

輸出內容應如下範例所示。

```
{
  "Attributes": [
    ...
    {
      "Key": "stickiness.enabled",
      "Value": "true"
    },
    {
      "Key": "stickiness.lb_cookie.duration_seconds",
      "Value": "86500"
    },
    ...
  ]
}
```

應用程式型粘性

應用程式型粘性可讓您靈活地設定自己的用戶端目標粘性標準。啟用應用程式型粘性後，負載平衡器會根據選擇的演算法，將第一個請求路由到目標群組內的目標。目標預期會設定與負載平衡器上設定的 Cookie 相符的自訂應用程式 Cookie，以啟用粘性。這個自定義 Cookie 可以包括應用程式所需的任何 Cookie 屬性。

Application Load Balancer 從目標收到自訂應用程式 Cookie 後，會自動產生新的加密應用程式 Cookie，以擷取粘性資訊。此負載平衡器產生的應用程式 Cookie 會擷取每個已啟用應用程式型粘性之目標群組的粘性資訊。

負載平衡器產生的應用程式 Cookie 不會複製目標所設定之自訂 Cookie 的屬性。它自己有 7 天的有效期，且無法設定。在對用戶端的回應中，Application Load Balancer 只會驗證在目標群組層級設定之自訂 Cookie 的名稱，而不會驗證自訂 Cookie 的值或到期屬性。只要名稱相符，負載平衡器會在對用戶端待回應中傳送兩個 Cookie，即目標設定的自訂 Cookie 以及負載平衡器產生的應用程式 Cookie。

在後續請求中，用戶端必須傳回這兩個 Cookie 以保持粘性。負載平衡器會解密應用程式 Cookie，並檢查設定的粘性持續時間是否仍然有效。然後，它會使用 Cookie 中的資訊來將請求傳送給目標群組中的同一個目標，以維持粘性。負載平衡器也會將自訂應用程式 Cookie 代理至目標，且不會對其進行檢查或修改。在後續回應中，負載平衡器產生的應用程式 Cookie 到期時間，以及在負載平衡器上設定的粘性持續時間都會重設。為了保持用戶端和目標之間的粘性，Cookie 的到期時間和粘性的持續時間不應結束。

如果目標失敗或運作狀態不佳，負載平衡器會停止路由請求到該目標，並根據選擇的負載平衡演算法選擇運作狀態良好的新目標。負載平衡器會將工作階段視為「粘到」運作狀態良好的新目標，並持續路由請求到運作狀態良好的新目標，即使失敗的目標恢復也是如此。

對於跨來源資源共用 (CORS) 請求，若要啟用粘性，負載平衡器只會在使用者代理程式版本為 Chromium80 或更高版本時，將 SameSite=None; Secure 屬性新增至負載平衡器產生的應用程式 Cookie。

由於大部分瀏覽器會將 Cookie 大小限制在 4K，負載平衡器會將大於 4K 的應用程式 Cookie 分成多個 Cookie 碎片。Application Load Balancer 支援的 Cookie 大小上限為 16K，因此最多會建立 4 個傳送到用戶端的碎片。用戶端看到的應用程式 Cookie 名稱以「AWSALBAPP-」開頭，並包含片段編號。例如，如果 Cookie 大小為 0-4K，則用戶端會看到 AWSALBAPP -0。如果 Cookie 的大小是 4-8k，則客戶端會看到 AWSALBAPP -0 和 AWSALBAPP -1，依此類推。

使用主控台啟用應用程式型粘性

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資訊索引標籤的屬性區段中，選擇編輯。
5. 在 Edit attributes (編輯屬性) 頁面上，執行下列動作：
 - a. 選取粘性。
 - b. 在粘性類型中，選取應用程式型 Cookie。
 - c. 針對 Stickiness duration (黏性持續期間)，指定介於 1 秒到 7 天之間的值。
 - d. 在應用程式 Cookie 名稱中，輸入應用程式型 Cookie 的名稱。

請勿使用 AWSALB、AWSALBAPP、或 AWSALBTG 作為 Cookie 名稱；它們預留供負載平衡器使用。

- e. 選擇儲存變更。

若要啟用以應用程式為基礎的黏性，請使用 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配以下屬性：

- `stickiness.enabled`
- `stickiness.type`
- `stickiness.app_cookie.cookie_name`
- `stickiness.app_cookie.duration_seconds`

請使用下列命令啟用應用程式型粘性。

```
aws elbv2 modify-target-group-attributes --target-group-arn ARN --attributes
Key=stickiness.enabled,Value=true Key=stickiness.type,Value=app_cookie
Key=stickiness.app_cookie.cookie_name,Value=my-cookie-name
Key=stickiness.app_cookie.duration_seconds,Value=time-in-seconds
```

輸出內容應如下範例所示。

```
{
  "Attributes": [
    ...
  ]
}
```

```
    {
      "Key": "stickiness.enabled",
      "Value": "true"
    },
    {
      "Key": "stickiness.app_cookie.cookie_name",
      "Value": "MyCookie"
    },
    {
      "Key": "stickiness.type",
      "Value": "app_cookie"
    },
    {
      "Key": "stickiness.app_cookie.duration_seconds",
      "Value": "86500"
    },
    ...
  ]
}
```

手動重新平衡

縱向擴展時，如果目標數量大幅增加，則由於粘性，可能會導致負載分配不均衡。在這種情況下，可以使用下列兩個選項重新平衡目標上的負載：

- 在由應用程式產生的 Cookie 上設定早於目前日期和時間的到期時間。這樣可防止用戶端將 Cookie 傳送至 Application Load Balancer，重新啟動建立粘性的程序。
- 在負載平衡器的應用程式型粘性組態上設定非常短的持續時間，例如 1 秒。這會強制 Application Load Balancer 重新建立粘性，即使目標所設定的 Cookie 尚未過期。

Lambda 函數作為目標

您可以將 Lambda 函數註冊為目標，並設定接聽程式規則，將請求轉送到 Lambda 函數的目標群組。當負載平衡器將請求轉送到使用 Lambda 函數做為目標的目標群組時，它會呼叫您的 Lambda 函數，並將請求的內容以 JSON 格式傳遞至 Lambda 函數。

限制

- Lambda 函數和目標群組必須在相同的帳戶中，且在相同的區域內。

- 您可以傳送到 Lambda 函數之請求內文的大小上限是 1 MB。如需相關的大小限制，請參閱 [HTTP header limits](#)。
- Lambda 函數可以傳送的回應 JSON 的大小上限是 1 MB。
- WebSockets 不受支援。升級請求會被拒絕，出現 HTTP 400 代碼。
- 不支援 Local Zone。
- 不支援自動目標加權 (ATW)。

目錄

- [準備 Lambda 函數](#)
- [為 Lambda 函數建立目標群組](#)
- [從負載平衡器接收事件](#)
- [對負載平衡器進行回應](#)
- [多值標頭](#)
- [啟用運作狀態檢查](#)
- [取消註冊 Lambda 函數](#)

如需示範，請參閱 [Lambda Target on Application Load Balancer](#)。

準備 Lambda 函數

如果將 Lambda 函數與 Application Load Balancer 搭配使用，請採用下列建議。

調用 Lambda 函數的許可

如果您使用 AWS Management Console 來建立目標群組和註冊 Lambda 函數，主控台會代表您將所需許可新增至 Lambda 函數政策。否則，在建立目標群組並使用註冊函數之後 AWS CLI，您必須使用 [新增權限](#) 命令授與 Elastic Load Balancing 權限，以呼叫 Lambda 函數。我們建議您使用 `aws:SourceAccount` 和 `aws:SourceArn` 條件索引鍵來將函數調用限制在指定的目標群組。如需詳細資訊，請參閱《IAM 使用者指南》中的 [混淆代理人問題](#)。

```
aws lambda add-permission \  
--function-name lambda-function-arn-with-alias-name \  
--statement-id elb1 \  
--principal elasticloadbalancing.amazonaws.com \  
--action lambda:InvokeFunction \  

```

```
--source-arn target-group-arn \  
--source-account target-group-account-id
```

Lambda 函數版本控制

您可以為每個目標群組註冊一個 Lambda 函數。為了確保您可以變更 Lambda 函數，且負載平衡器一律會呼叫目前版本的 Lambda 函數，請建立一個函數別名，並將該別名包含在向負載平衡器註冊 Lambda 函數時的函數 ARN 中。如需詳細資訊，請參閱《AWS Lambda 開發人員指南》中的 [AWS Lambda 函數版本控制與別名功能](#)和[使用別名轉移流量](#)。

函數逾時

負載平衡器會等待直到您的 Lambda 函數回應或逾時。建議您根據您預期的執行時間來設定 Lambda 函數逾時。如需有關預設逾時值和如何變更它的詳細資訊，請參閱[基本 AWS Lambda 函數組態](#)。如需有關可設定之最大逾時值的詳細資訊，請參閱[AWS Lambda 限制](#)。

為 Lambda 函數建立目標群組

建立目標群組以用於請求路由。如果請求內容符合某接聽程式規則，內含會將它轉送到此目標群組的動作，則負載平衡器會呼叫註冊的 Lambda 函數。

使用主控台建立目標群組並註冊 Lambda 函數

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇 Create target group (建立目標群組)。
4. 在選取目標類型中，選取 Lambda 函數。
5. 針對 Target group name (目標群組名稱)，輸入目標群組的名稱。
6. (選用) 若要啟用運作狀態檢查，請選擇運作狀態檢查區段中的啟用。
7. (選用) 新增一個或多個標籤，如下所示：
 - a. 展開 Tags (標籤) 區段。
 - b. 選擇 Add tag (新增標籤)。
 - c. 輸入標籤金鑰和標籤值。
8. 選擇下一步。
9. 指定單一 Lambda 函數，或省略此步驟，稍後再指定 Lambda 函數。
10. 選擇 Create target group (建立目標群組)。

使用 AWS CLI 建立目標群組和註冊 Lambda 函數

使用 [create-target-group](#) 和 [register-targets](#) 命令。

從負載平衡器接收事件

負載平衡器同時支援透過 HTTP 和 HTTPS 的請求進行 Lambda 呼叫。負載平衡器會以 JSON 格式傳送事件。負載平衡器會將以下標頭新增至每個請求：X-Amzn-Trace-Id、X-Forwarded-For、X-Forwarded-Port 和 X-Forwarded-Proto。

如果 content-encoding 標頭存在，負載平衡器 Base64 會對內文使用 Base64 編碼並將 isBase64Encoded 設定為 true。

如果 content-encoding 標頭不存在，則 Base64 編碼取決於內容類型。如果是以下類型，負載平衡器會依原樣傳送內文並將 isBase64Encoded 設定為 false：text/*、application/json、application/javascript 和 application/xml。否則，負載平衡器會對內文使用 Base64 編碼，並將 isBase64Encoded 設定為 true。

以下為範例事件。

```
{
  "requestContext": {
    "elb": {
      "targetGroupArn":
        "arn:aws:elasticloadbalancing:region:123456789012:targetgroup/my-target-
        group/6d0ecf831eec9f09"
    }
  },
  "httpMethod": "GET",
  "path": "/",
  "queryStringParameters": {parameters},
  "headers": {
    "accept": "text/html,application/xhtml+xml",
    "accept-language": "en-US,en;q=0.8",
    "content-type": "text/plain",
    "cookie": "cookies",
    "host": "lambda-846800462-us-east-2.elb.amazonaws.com",
    "user-agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6)",
    "x-amzn-trace-id": "Root=1-5bdb40ca-556d8b0c50dc66f0511bf520",
    "x-forwarded-for": "72.21.198.66",
    "x-forwarded-port": "443",
    "x-forwarded-proto": "https"
  }
}
```

```
  },
  "isBase64Encoded": false,
  "body": "request_body"
}
```

對負載平衡器進行回應

來自 Lambda 函數的回應必須包含 Base64 編碼狀態、狀態碼、狀態描述和標頭。您可以省略內文。

若要在回應的內文中包含二進位內容，您必須將內容以 Base64 編碼，並將 `isBase64Encoded` 設定為 `true`。負載平衡器會解碼內容，以擷取二進位內容，並將其傳送至 HTTP 回應內文中的用戶端。

負載平衡器不接受 hop-by-hop 標頭，例如 `Connection` 或 `Transfer-Encoding`。您可以省略 `Content-Length` 標頭，因為負載平衡器會在將回應傳送至用戶端之前計算。

以下是來自基於 Lambda 函數的 `nodejs` 的範例回應。

```
{
  "isBase64Encoded": false,
  "statusCode": 200,
  "statusDescription": "200 OK",
  "headers": {
    "Set-cookie": "cookies",
    "Content-Type": "application/json"
  },
  "body": "Hello from Lambda (optional)"
}
```

若是與搭配 Application Load Balancer 使用的 Lambda 函數範本，相關資訊請至 GitHub 參閱 [application-load-balancer-serverless-app](#)。或者，開啟 [Lambda 主控台](#)，選擇應用程式、建立應用程式，然後從 AWS Serverless Application Repository 中選取下列其中一項：

- 阿爾 B-蘭姆達靶-S3 UploadFileto
- ALB-蘭姆達靶 BinaryResponse
- ALB-蘭姆達目標 IP WhatIsMy

多值標頭

如果來自用戶端的請求或來自 Lambda 函數的回應，包含具有多個值的標頭或包含相同標頭多次，或查詢參數具有多個值的相同索引鍵，您可以啟用對多值標頭語法的支援。啟用多值標頭之後，負載平衡

器和 Lambda 函數之間交換的標頭和查詢參數會使用陣列而不是字串。如果您未啟用多值標頭語法，且標頭或查詢參數具有多個值，負載平衡器會使用它接收的最後一個值。

目錄

- [具有多值標頭的請求](#)
- [具有多值標頭的回應](#)
- [啟用多值標頭](#)

具有多值標頭的請求

根據您是否為目標群組啟用多值標頭而定，用於標頭和查詢字串參數的欄位名稱有所不同。

以下範例請求具有使用相同金鑰的兩個查詢參數：

```
http://www.example.com?&myKey=val1&myKey=val2
```

採用預設格式時，負載平衡器會使用用戶端傳送的最後一個值，並使用 `queryStringParameters` 向您傳送包含查詢字串參數的事件。例如：

```
"queryStringParameters": { "myKey": "val2"},
```

如果您啟用多值標頭，負載平衡器會使用用戶端傳送的兩個金鑰值，並使用 `multiValueQueryStringParameters` 向您傳送包含查詢字串參數的事件。例如：

```
"multiValueQueryStringParameters": { "myKey": ["val1", "val2"] },
```

同樣地，假設用戶端會傳送的請求標頭中具有兩個 Cookie：

```
"cookie": "name1=value1",  
"cookie": "name2=value2",
```

採用預設格式時，負載平衡器會使用用戶端傳送的最後一個 Cookie，並使用 `headers` 向您傳送包含標頭的事件。例如：

```
"headers": {  
  "cookie": "name2=value2",
```

```
    ...
  },
```

如果您啟用多值標頭，負載平衡器會使用用戶端傳送的兩個 Cookie，並使用 `multiValueHeaders` 向您傳送包含標頭的事件。例如：

```
"multiValueHeaders": {
  "cookie": ["name1=value1", "name2=value2"],
  ...
},
```

如果查詢參數是 URL 編碼，負載平衡器不會進行解碼。您必須在 Lambda 函數中解碼。

具有多值標頭的回應

根據您是否為目標群組啟用多值標頭而定，用於標頭的欄位名稱有所不同。如果您已啟用多重值標頭和 `headers`，您必須使用 `multiValueHeaders`。

使用預設格式，您可以指定單一 Cookie：

```
{
  "headers": {
    "Set-cookie": "cookie-name=cookie-value;Domain=myweb.com;Secure;HttpOnly",
    "Content-Type": "application/json"
  },
}
```

如果您啟用多值標頭，您必須如下所示指定多個 Cookie：

```
{
  "multiValueHeaders": {
    "Set-cookie": ["cookie-name=cookie-
value;Domain=myweb.com;Secure;HttpOnly", "cookie-name=cookie-value;Expires=May 8,
2019"],
    "Content-Type": ["application/json"]
  },
}
```

負載平衡器可能會依照與 Lambda 回應承載中指定順序不同的順序將標頭傳送到用戶端。因此，不要指望標頭會以特定順序返回。

啟用多值標頭

您可以為具有目標類型 `lambda` 的目標群組啟用或停用多值標頭。

使用主控台啟用多值標頭

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資訊索引標籤的屬性區段中，選擇編輯。
5. 選取或清除多值標頭。
6. 選擇儲存變更。

若要使用啟用多值標頭 AWS CLI

使用 [modify-target-group-attributes](#) 命令搭配 `lambda.multi_value_headers.enabled` 屬性。

啟用運作狀態檢查

在預設情況下，會為類型 `lambda` 的目標群組停用運作狀態檢查。您可以啟用運作狀態檢查，以便使用 Amazon Route 53 來實作 DNS 備援。Lambda 函數可以檢查下游服務的運作狀態，之後再回應運作狀態檢查的請求。如果來自 Lambda 函數的回應指出運作狀態檢查失敗，則會將運作狀態檢查失敗傳遞至 Route 53。您可以設定 Route 53 以容錯移轉到備用的應用程式堆疊。

將向您就任何 Lambda 函數呼叫而進行的運作狀態檢查收費。

以下是傳送到 Lambda 函數的運作狀態檢查事件格式。若要檢查事件是否為運作狀態檢查事件，請檢查 `user-agent` 欄位的值。運作狀態檢查的使用者代理程式為 `ELB-HealthChecker/2.0`。

```
{
  "requestContext": {
    "elb": {
      "targetGroupArn":
        "arn:aws:elasticloadbalancing:region:123456789012:targetgroup/my-target-
        group/6d0ecf831eec9f09"
    }
  },
  "httpMethod": "GET",
  "path": "/",
  "queryStringParameters": {},
```

```
"headers": {
  "user-agent": "ELB-HealthChecker/2.0"
},
"body": "",
"isBase64Encoded": false
}
```

使用主控台啟用目標群組的健全狀況檢查

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在群組詳細資料索引標籤的運作狀態檢查設定區段中，選擇編輯。
5. 在運作狀態檢查中，選取啟用。
6. 選擇儲存變更。

使用啟用目標群組的健全狀況檢查 AWS CLI

使用 [modify-target-group](#) 命令搭配 `--health-check-enabled` 選項。

取消註冊 Lambda 函數

如果您不再需要將流量傳送到您的 Lambda 函數，則可以將它取消註冊。取消註冊 Lambda 函數之後，傳輸中的請求會失敗，出現 HTTP 5XX 錯誤。

若要取代 Lambda 函數，建議您建立新的目標群組、向新目標群組註冊新函數，並更新接聽程式規則以使用新的目標群組，而非現有的目標群組。

若要使用主控台取消註冊 Lambda 函數

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在 Targets (目標) 索引標籤上，選擇 Deregister (取消註冊)。
5. 出現確認的提示時，請選擇取消註冊。

若要使用取消註冊 Lambda 函數 AWS CLI

使用 [deregister-targets](#) 命令。

目標群組的標籤

標籤可幫助您以不同的方式來將目標群組分類，例如，根據目的、擁有者或環境。

您可以在每個目標群組中加入多個標籤。每個目標群組的標籤索引鍵必須是唯一的。如果所新增的標籤，其索引鍵已經和目標群組具有關聯，則此動作會更新該標籤的值。

當您使用完標籤之後，可以將其移除。

限制

- 每一資源標籤數上限：50
- 索引鍵長度上限：127 個 Unicode 字元
- 數值長度上限：255 個 Unicode 字元
- 標籤金鑰與值皆區分大小寫。允許的字元包括可用 UTF-8 表示的英文字母、空格和數字，還有以下特殊字元：`+ - = . _ : / @`。不可使用結尾或前方空格。
- 請勿在標籤名稱或值中使用 `aws:` 前置詞，因為它已保留供 AWS 使用。您不可編輯或刪除具此字首的標籤名稱或值。具此字首的標籤，不算在受資源限制的標籤計數內。

使用主控台來更新目標群組的標籤

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選擇目標群組的名稱，以開啟其詳細資訊頁面。
4. 在標籤索引標籤上，選擇管理標籤，並執行下列一個或多個動作：
 - a. 若要更新標籤，請為索引鍵和值輸入新值。
 - b. 如要新增標籤，請選擇新增標籤，然後輸入索引鍵和值的值。
 - c. 若要移除標籤，請選擇標籤旁的移除。
5. 完成標籤的更新作業後，請選擇儲存變更。

若要使用更新目標群組的標記 AWS CLI

使用 [add-tags](#) 和 [remove-tags](#) 指令。

刪除目標群組

如果沒有任何接聽程式規則的轉送動作參照某目標群組，即可刪除該目標群組。刪除目標群組不會影響透過該目標群組登錄的目標。如果不再需要註冊的 EC2 執行個體，則可以停止或終止它。

使用主控台來刪除目標群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Load Balancing (負載平衡) 中，選擇 Target Groups (目標群組)。
3. 選取目標群組，然後依序選擇 Actions (動作)、Delete (刪除)。
4. 出現確認提示時，選擇是，刪除。

若要使用刪除目標群組 AWS CLI

使用 [delete-target-group](#) 指令。

監控 Application Load Balancer

您可使用以下功能來監控負載平衡器、分析流量模式並對與負載平衡器和目標相關的問題進行疑難排解。

CloudWatch 度量

您可以使用 Amazon 擷取 CloudWatch 取負載平衡器和目標資料點的相關統計資料，做為一組排序的時間序列資料 (稱為指標)。您可以使用這些指標來確認您的系統是否依照預期執行。如需詳細資訊，請參閱 [CloudWatch Application Load Balancer 的指標](#)。

存取日誌

您可以使用存取日誌，來擷取對負載平衡器發出之請求的詳細資訊，並將這些資訊作為日誌檔案存放在 Amazon S3。您可以使用這些存取日誌來分析流量模式，並排除目標的問題。如需詳細資訊，請參閱 [Application Load Balancer 的存取日誌](#)。

連線日誌

您可以使用連線日誌擷取傳送至負載平衡器的請求相關屬性，並將其作為日誌檔存放在 Amazon S3 中。您可以使用這些連線記錄檔來判斷使用的用戶端 IP 位址和連接埠、用戶端憑證資訊、連線結果以及 TLS 加密。然後，這些連線記錄可用於檢閱請求模式和其他趨勢。如需詳細資訊，請參閱 [Application Load Balancer 的連線記錄](#)。

請求追蹤

您可以使用請求追蹤來追蹤 HTTP 請求。負載平衡器會在收到的每個請求中新增標頭和追蹤識別符。如需詳細資訊，請參閱 [Application Load Balancer 上的請求追蹤](#)。

CloudTrail 日誌

您可以使用擷取 AWS CloudTrail 取有關 Elastic Load Balancing API 呼叫的詳細資訊，並將它們作為日誌檔存放在 Amazon S3 中。您可以使用這些 CloudTrail 記錄來判斷撥打哪些呼叫、來源 IP 位址呼叫來源、撥打電話的人員、撥打電話的時間等等。如需詳細資訊，請參閱 [使用 AWS CloudTrail 記錄 Application Load Balancer 的 API 呼叫](#)。

CloudWatch Application Load Balancer 的指標

Elastic Load Balancing 會針對 CloudWatch 對負載平衡器和目標將資料點發佈至 Amazon。

CloudWatch 可讓您擷取有關這些資料點的統計資料，做為一組排序的時間序列資料 (稱為指標)。您可

以將指標視為要監控的變數，且資料點是該變數在不同時間點的值。例如，您可以監控負載平衡器在一段指定期間內的運作狀態良好的目標總數量。每個資料點都有關聯的時間戳記和可選的測量單位。

您可以使用指標來確認系統的運作符合預期。例如，如果指標超出您認為可接受的範圍，您可以建立 CloudWatch 警示來監控指定的指標並啟動動作 (例如傳送通知至電子郵件地址)。

CloudWatch 只有在要求流經負載平衡器時，「Elastic Load Balancing」才會將度量報告給。如果有請求進出負載平衡器，Elastic Load Balancing 會以 60 秒為間隔來測量並傳送其指標。如果沒有請求流經負載平衡器，或者指標沒有資料，則不會回報該指標。

如需詳細資訊，請參閱 [Amazon CloudWatch 使用者指南](#)。

目錄

- [Application Load Balancer 指標](#)
- [Application Load Balancer 的指標維度](#)
- [Application Load Balancer 指標的統計資料](#)
- [CloudWatch 檢視負載平衡器的指標](#)

Application Load Balancer 指標

- [負載平衡器](#)
- [目標](#)
- [目標群組運作狀態](#)
- [Lambda 函數](#)
- [使用者身分驗證](#)

AWS/ApplicationELB 命名空間包含下列負載平衡器指標。

指標	描述
ActiveConnectionCount	從用戶端到負載平衡器以及從負載平衡器到目標的並行作用中 TCP 連線總數。 報告條件：有非零值 統計資訊：最實用的統計資訊是 Sum。

指標	描述
	<p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
AnomalousHostCount	<p>偵測到異常的主機數目。</p> <p>報告條件：一律報告</p> <p>統計資訊：最實用的統計資訊是 Average、Minimum 與 Maximum。</p> <p>維度</p> <ul style="list-style-type: none"> • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer
ClientTLSNegotiationErrorCount	<p>由於 TLS 錯誤而未與負載平衡器建立工作階段之用戶端所啟動的 TLS 連線數目。可能的原因包括密碼或通訊協定不相符，或用戶端無法驗證伺服器憑證並關閉連線。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
ConsumedLCUs	<p>負載平衡器所使用的負載平衡器容量單位 (LCU) 數目。您需要按每小時使用的 LCU 數目付費。如需詳細資訊，請參閱 Elastic Load Balancing 定價。</p> <p>報告條件：一律報告</p> <p>統計資訊：全部</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer
DesyncMitigationMode_NonCompliant_Request_Count	<p>不符合 RFC 7230 的請求數量。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
DroppedInvalidHeaderRequestCount	<p>在傳送請求之前，負載平衡器移除具有無效標頭欄位的 HTTP 標頭的請求數目。只有在 <code>routing.http.drop_invalid_header_fields.enabled</code> 屬性設定為 <code>true</code> 時，負載平衡器才會移除這些標頭。</p> <p>報告條件：有非零值</p> <p>統計資訊：全部</p> <p>維度</p> <ul style="list-style-type: none"> • AvailabilityZone , LoadBalancer

指標	描述
MitigatedHostCount	<p>緩和措施的目標數目。</p> <p>報告條件：一律報告</p> <p>統計資訊：最實用的統計資訊是 Average、Minimum 與 Maximum。</p> <p>維度</p> <ul style="list-style-type: none"> • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer
ForwardedInvalidHeaderRequestCount	<p>由 HTTP 標頭具有無效標頭欄位的負載平衡器所傳送的請求數目。只有在 <code>routing.http.drop_invalid_header_fields.enabled</code> 屬性設定為 <code>false</code> 時，負載平衡器才會轉送具有這些標頭的請求。</p> <p>報告條件：一律報告</p> <p>統計資訊：全部</p> <p>維度</p> <ul style="list-style-type: none"> • AvailabilityZone , LoadBalancer
GrpcRequestCount	<p>透過 IPv4 與 IPv6 處理的 gRPC 請求數量。</p> <p>報告條件：有非零值</p> <p>統計資料：最實用的統計數量是 Sum。Minimum、Maximum 和 Average 都會傳回 1。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
HTTP_Fixed_Response_Count	<p>成功的固定回應動作次數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none">• LoadBalancer• AvailabilityZone , LoadBalancer
HTTP_Redirect_Count	<p>成功的重新導向動作次數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none">• LoadBalancer• AvailabilityZone , LoadBalancer
HTTP_Redirect_Url_Limit_Exceeded_Count	<p>因為回應位置標頭中的 URL 大於 8K 而無法完成的重新導向動作數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none">• LoadBalancer• AvailabilityZone , LoadBalancer

指標	描述
HTTPCode_ELB_3XX_Count	<p>源自於負載平衡器的 HTTP 3XX 重新導向代碼數目。此計數未包含目標所產生的回應碼。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none">• LoadBalancer• AvailabilityZone , LoadBalancer
HTTPCode_ELB_4XX_Count	<p>源自於負載平衡器的 HTTP 4XX 用戶端錯誤碼數目。此計數未包含目標所產生的回應碼。</p> <p>要求的格式不正確或不完整時，會產生用戶端錯誤。除了負載平衡器傳回 HTTP 460 錯誤碼 的情況之外，目標沒有收到這些要求。此計數未包含目標所產生的任何回應碼。</p> <p>報告條件：有非零值</p> <p>統計資料：最實用的統計數量是 Sum。Minimum、Maximum 和 Average 都會傳回 1。</p> <p>維度</p> <ul style="list-style-type: none">• LoadBalancer• AvailabilityZone , LoadBalancer

指標	描述
HTTPCode_ELB_5XX_Count	<p>源自於負載平衡器的 HTTP 5XX 伺服器錯誤碼數目。此計數未包含目標所產生的任何回應碼。</p> <p>報告條件：有非零值</p> <p>統計資料：最實用的統計數量是 Sum。Minimum、Maximum 和 Average 都會傳回 1。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
HTTPCode_ELB_500_Count	<p>源自於負載平衡器的 HTTP 500 錯誤碼數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
HTTPCode_ELB_502_Count	<p>源自於負載平衡器的 HTTP 502 錯誤碼數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
HTTPCode_ELB_503_Count	<p>源自於負載平衡器的 HTTP 503 錯誤碼數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
HTTPCode_ELB_504_Count	<p>源自於負載平衡器的 HTTP 504 錯誤碼數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
IPv6ProcessedBytes	<p>負載平衡器透過 IPv6 所處理的位元組總數。此計數包含在 ProcessedBytes 中。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
IPv6RequestCount	<p>負載平衡器收到的 IPv6 要求數目。</p> <p>報告條件：有非零值</p> <p>統計資料：最實用的統計數量是 Sum。Minimum、Maximum 和 Average 都會傳回 1。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
NewConnectionCount	<p>從用戶端到負載平衡器以及從負載平衡器到目標建立的新 TCP 連線總數。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
NonStickyRequestCount	<p>負載平衡器因為無法使用現有的黏性工作階段而選擇新目標時的請求數目。例如，請求是來自新用戶端的第一個請求且黏性 Cookie 不存在、黏性 Cookie 存在但未指定已向此目標群組註冊的目標、黏性 Cookie 的格式不正確或過期，或內部錯誤使負載平衡器無法讀取黏性 Cookie。</p> <p>報告條件：目標群組上啟用黏性。</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
ProcessedBytes	<p>負載平衡器透過 IPv4 與 IPv6 (HTTP 標頭和 HTTP 承載) 所處理的位元組總數。此計數包含進出用戶端和 Lambda 函數的流量，以及來自身分識別提供者 (IdP) 的流量 (如果使用者身分驗證已啟用)。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
RejectedConnectionCount	<p>因負載平衡器已達其連線數目上限而拒絕的連線數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
RequestCount	<p>透過 IPv4 與 IPv6 處理的要求數目。對於負載平衡器節點能夠選擇目標的請求，此指標的值才會遞增。在選擇目標之前遭拒絕的請求不會反映在此指標中。</p> <p>報告條件：一律報告</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • LoadBalancer , AvailabilityZone • LoadBalancer , TargetGroup • LoadBalancer , AvailabilityZone , TargetGroup

指標	描述
RuleEvaluations	<p>具有一小時平均要求率之負載平衡器所處理的規則數目。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer

AWS/ApplicationELB 命名空間包含下列目標指標。

指標	描述
HealthyHostCount	<p>視為健康的目標數目。</p> <p>報告條件：運作狀態檢查啟用時報告</p> <p>統計資訊：最實用的統計資訊是 Average、Minimum 與 Maximum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • LoadBalancer , AvailabilityZone , TargetGroup
HTTPCode_Target_2XX_Count , HTTPCode_Target_3XX_Count , HTTPCode_Target_4XX_Count , HTTPCode_Target_5XX_Count	<p>目標所產生的 HTTP 回應碼數目。這未包含負載平衡器所產生的任何回應碼。</p> <p>報告條件：有非零值</p> <p>統計資料：最實用的統計數量是 Sum。Minimum、Maximum 和 Average 都會傳回 1。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
	<ul style="list-style-type: none"> • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer
RequestCountPerTarget	<p>目標群組中每個目標的平均要求計數。您必須使用 TargetGroup 維度指定目標群組。如果目標是 Lambda 函數，則此指標不適用。</p> <p>此計數會使用目標群組接收的要求總數，除以目標群組中健全狀況良好的目標數目。如果目標群組中沒有健全狀況的目標，則會報告目標的總數。</p> <p>報告條件：一律報告</p> <p>統計資訊：唯一有效的統計資訊是 Sum。這代表平均值，而不是總和。</p> <p>維度</p> <ul style="list-style-type: none"> • TargetGroup • TargetGroup , AvailabilityZone • LoadBalancer , TargetGroup • LoadBalancer , AvailabilityZone , TargetGroup
TargetConnectionErrorCount	<p>負載平衡器與目標之間未成功建立的連線數目。如果目標是 Lambda 函數，則此指標不適用。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer

指標	描述
TargetResponseTime	<p>要求離開負載平衡器直到目標開始傳送回應標頭之前所經過的時間 (以秒為單位)。這等同於存取日誌中的 <code>target_processing_time</code> 欄位。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Average 與 pNN.NN (百分位數)。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer
TargetTLSNegotiationErrorCount	<p>未與目標建立工作階段之負載平衡器所啟動的 TLS 連線數目。可能的原因包含晶片或協定不相符。如果目標是 Lambda 函數，則此指標不適用。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer • TargetGroup , LoadBalancer • TargetGroup , AvailabilityZone , LoadBalancer

指標	描述
UnHealthyHostCount	<p>視為不健康的目標數目。</p> <p>報告條件：運作狀態檢查啟用時報告</p> <p>統計資訊：最實用的統計資訊是 Average、Minimum 與 Maximum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • LoadBalancer , AvailabilityZone , TargetGroup

AWS/ApplicationELB 命名空間包含下列目標群組運作狀態的指標。如需詳細資訊，請參閱 [the section called “目標群組運作狀態”](#)。

指標	描述
HealthyStateDNS	<p>符合 DNS 運作狀態良好需求的區域數目。</p> <p>統計資訊：最實用的統計資訊是 Min。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • AvailabilityZone , LoadBalancer , TargetGroup
HealthyStateRouting	<p>符合路由運作狀態良好需求的區域數目。</p> <p>統計資訊：最實用的統計資訊是 Min。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • AvailabilityZone , LoadBalancer , TargetGroup
UnhealthyRoutingRequestCount	<p>使用路由容錯移轉動作 (故障開啟) 路由的請求數量。</p> <p>統計資訊：最實用的統計資訊是 Sum。</p>

指標	描述
	<p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • AvailabilityZone , LoadBalancer , TargetGroup
UnhealthyStateDNS	<p>因不符合 DNS 運作狀態良好需求而在 DNS 中標記為運作狀態不佳的區域數量。</p> <p>統計資訊：最實用的統計資訊是 Min。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • AvailabilityZone , LoadBalancer , TargetGroup
UnhealthyStateRouting	<p>因不符合路由運作狀態良好需求而導致負載平衡器將流量分散給區域中的所有目標 (包括運作狀態不佳的目標) 的區域數量。</p> <p>統計資訊：最實用的統計資訊是 Min。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer , TargetGroup • AvailabilityZone , LoadBalancer , TargetGroup

對於註冊為目標的 Lambda 函數，AWS/ApplicationELB 命名空間包含下列指標。

指標	描述
LambdaInternalError	<p>因為負載平衡器或 AWS Lambda 的內部問題而失敗的 Lambda 函數請求數。若要取得錯誤原因代碼，請查看存取日誌的 error_reason 欄位。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p>

指標	描述
	<p>維度</p> <ul style="list-style-type: none"> • TargetGroup • TargetGroup , LoadBalancer
LambdaTargetProcessedBytes	<p>負載平衡器針對 Lambda 函數的請求和回應所處理的位元組總數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer
LambdaUserError	<p>Lambda 函數因為 Lambda 函數有問題而失敗的請求數。例如，負載平衡器沒有叫用函數的許可、負載平衡器從函數收到的 JSON 格式不正確或遺漏必要欄位，或請求內文或回應的大小超過大小上限 1 MB。若要取得錯誤原因代碼，請查看存取日誌的 error_reason 欄位。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • TargetGroup • TargetGroup , LoadBalancer

AWS/ApplicationELB 命名空間包含下列使用者身分驗證指標。

指標	描述
ELBAuthError	<p>由於身分驗證動作設定錯誤、負載平衡器無法與 IdP 建立連線，或負載平衡器由於內部錯誤而無法完成身分驗證流程，因而無法完成的使用者身分驗證數量。若要取得錯誤原因代碼，請查看存取日誌的 error_reason 欄位。</p>

指標	描述
	<p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
ELBAuthFailure	<p>由於 IdP 拒絕使用者存取或授權碼使用多次，因而無法完成的使用者身分驗證數量。若要取得錯誤原因代碼，請查看存取日誌的 error_reason 欄位。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
ELBAuthLatency	<p>向 IdP 查詢 ID 字符和使用者資訊所經歷的時間 (毫秒)。如果其中一或多項操作失敗，此為失敗的時間。</p> <p>報告條件：有非零值</p> <p>統計資訊：所有統計資訊都有意義。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

指標	描述
ELBAuthRefreshTokenSuccess	<p>負載平衡器使用 IdP 提供的重新整理字符而成功重新整理使用者宣告的次數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
ELBAuthSuccess	<p>成功的身分驗證動作次數。此指標在身分驗證工作流程結束時、負載平衡器從 IdP 擷取到使用者宣告之後遞增。</p> <p>報告條件：有非零值</p> <p>統計資訊：最實用的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer
ELBAuthUserClaimsSizeExceeded	<p>已設定的 IdP 傳回的使用者宣告大小超過 11K 位元組的次數。</p> <p>報告條件：有非零值</p> <p>統計資訊：唯一有意義的統計資訊是 Sum。</p> <p>維度</p> <ul style="list-style-type: none"> • LoadBalancer • AvailabilityZone , LoadBalancer

Application Load Balancer 的指標維度

若要篩選 Application Load Balancer 的指標，請使用下列維度。

維度	描述
AvailabilityZone	依可用區域篩選指標資料。
LoadBalancer	依負載平衡器篩選指標資料。如下指定負載平衡器：app/load-balancer-name/1234567890123456 (負載平衡器 ARN 的最終部分)。
TargetGroup	依目標群組篩選指標資料。如下指定目標群組：targetgroup/target-group-name/1234567890123456 (目標群組 ARN 的最終部分)。

Application Load Balancer 指標的統計資料

CloudWatch 根據「Elastic Load Balancing」公佈的測量結果資料點，提供統計資料。統計資料是隨著指定期間的指標資料彙總。當您請求統計資料時，傳回的資料流是藉由指標名稱和維度做識別。維度是可唯一識別指標的名稱/值組。例如，您可以為所有在特定可用區域內啟動的負載平衡器後方之運作狀態良好的 EC2 執行個體請求統計資料。

Minimum 和 Maximum 統計資料會反映每個抽樣時段中個別負載平衡器節點報告的資料點最小和最大值。例如，假設 Application Load Balancer 由 2 個負載平衡器節點組成。一個節點有內含 Minimum 2、Maximum 10、Average 6 的 HealthyHostCount，而其他節點有內含 Minimum 1、Maximum 5、以及 Average 3 的 HealthyHostCount。因此，負載平衡器有 Minimum 1、Maximum 10、以及因為約為 4 的 Average。

我們建議您監控 Minimum 統計資料中的非零值 UnHealthyHostCount，多個資料點出現非零值時提供警示。使用 Minimum 將偵測負載平衡器每個節點和可用區域何時將目標視為運作狀態不佳。如果您想要收到潛在問題的警示，則 Average 或 Maximum 警示非常有用，我們建議客戶檢閱此指標，並在發生次數不為零時進行調查。您可以遵循在 Amazon EC2 Auto Scaling 或 Amazon Elastic Container Service (Amazon ECS) 中使用負載平衡器運作狀態檢查的最佳實務，自動緩解故障。

Sum 統計資料為來自所有負載平衡器節點的彙總值。因為指標包和各期間的多個報告，Sum 僅可用於來自所有負載平衡器節點的彙總指標。

SampleCount 統計資料為測量而得的範本數量。因指標根據範本間隔與事件蒐集而得，此統計資料通常沒有幫助。例如，使用 HealthyHostCount，SampleCount 是根據每個負載平衡器節點回報的範本數量，而非運作狀態良好的主機數量。

百分位數指出資料集之某個值的相對位置。您可以指定任何百分位數，最多使用兩位小數 (例如，p95.45)。例如，第 95 個百分位數表示 95% 的資料低於這個值，而 5% 高於這個值。百分位數通常用於隔離異常。例如，假設應用程式以 1-2 毫秒處理快取中的大部分請求，但如果快取是空的，則是 100-200 毫秒。上限會反映最慢的情況，大約 200 毫秒。平均數不表示資料的分佈。百分位數以更有意義的觀點表達應用程式的效能。藉由使用第 99 個百分位數作為 Auto Scaling 觸發器或 CloudWatch 警示，您可以指定不超過 1% 的要求需要超過 2 毫秒的時間。

CloudWatch 檢視負載平衡器的指標

您可以使用 Amazon EC2 主控台檢視負載平衡器的 CloudWatch 指標。這些指標會以監控圖表的形式顯示。若啟用負載平衡器並接收請求，監控圖表會顯示資料點。

或者，您可以使用 CloudWatch 主控台檢視負載平衡器的指標。

使用 主控台檢視指標

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 若要檢視由目標群組篩選的指標，請執行下列動作：
 - a. 在導覽窗格中，選擇 Target Groups (目標群組)。
 - b. 選取您的目標群組，然後選擇 Monitoring (監控) 標籤。
 - c. (選用) 若要根據時間篩選結果，請選擇來自 Showing data for (顯示資料) 的時間範圍。
 - d. 若要放大檢視單一指標，請選取它的圖形。
3. 若要檢視由負載平衡器篩選的指標，請執行下列動作：
 - a. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
 - b. 選取您的負載平衡器，然後選擇 Monitoring (監控) 標籤。
 - c. (選用) 若要根據時間篩選結果，請選擇來自 Showing data for (顯示資料) 的時間範圍。
 - d. 若要放大檢視單一指標，請選取它的圖形。

使用 CloudWatch 主控台檢視指標

1. [請在以下位置開啟 CloudWatch 主控台。](https://console.aws.amazon.com/cloudwatch/) <https://console.aws.amazon.com/cloudwatch/>
2. 在導覽窗格中，選擇 指標。
3. 選取 ApplicationELB 命名空間。
4. (選用) 若要檢視所有維度的指標，請在搜尋欄位中輸入其名稱。

5. (選用) 若要根據維度來篩選，請選取下列其中一項：

- 若只要顯示針對負載平衡器而報告的指標，請選擇 Per AppELB Metrics (每個 AppELB 指標)。若要檢視單一負載平衡器的指標，請在搜尋欄位中輸入其名稱。
- 若只要顯示針對目標群組而報告的指標，請選擇 Per AppELB, per TG Metrics (每個 AppELB、每個 TG 指標)。若要檢視單一目標群組的指標，請在搜尋欄位中輸入其名稱。
- 若要依可用區域來只顯示針對負載平衡器而報告的指標，請選擇 Per AppELB, per AZ Metrics (每個 AppELB、每個 AZ 指標)。若要檢視單一負載平衡器的指標，請在搜尋欄位中輸入其名稱。若要檢視單一可用區域的指標，請在搜尋欄位中輸入其名稱。
- 若要依可用區域和目標群組來只顯示針對負載平衡器而報告的指標，請選擇 Per AppELB, per AZ, per TG Metrics (每個 AppELB、每個 AZ、每個 TG 指標)。若要檢視單一負載平衡器的指標，請在搜尋欄位中輸入其名稱。若要檢視單一目標群組的指標，請在搜尋欄位中輸入其名稱。若要檢視單一可用區域的指標，請在搜尋欄位中輸入其名稱。

若要使用 AWS CLI

使用下列 [list-metrics](#) 命令來列出可用指標：

```
aws cloudwatch list-metrics --namespace AWS/ApplicationELB
```

若要取得測量結果的統計資料，請使用 AWS CLI

使用下列取得 [量度統計資料命令取得指定之測量](#) 結果和維度的統計資料。CloudWatch 將每個唯一維度組合視為單獨的量度。您無法使用未具體發佈的維度組合來擷取統計資料。您必須指定建立指標時所使用的相同維度。

```
aws cloudwatch get-metric-statistics --namespace AWS/ApplicationELB \  
--metric-name UnHealthyHostCount --statistics Average --period 3600 \  
--dimensions Name=LoadBalancer,Value=app/my-load-balancer/50dc6c495c0c9188 \  
Name=TargetGroup,Value=targetgroup/my-targets/73e2d6bc24d8a067 \  
--start-time 2016-04-18T00:00:00Z --end-time 2016-04-21T00:00:00Z
```

下列為範例輸出：

```
{  
  "Datapoints": [  
    {  
      "Timestamp": "2016-04-18T22:00:00Z",
```

```
        "Average": 0.0,
        "Unit": "Count"
    },
    {
        "Timestamp": "2016-04-18T04:00:00Z",
        "Average": 0.0,
        "Unit": "Count"
    },
    ...
],
"Label": "UnHealthyHostCount"
}
```

Application Load Balancer 的存取日誌

Elastic Load Balancing 提供存取日誌，可針對傳送到負載平衡器的請求，擷取其詳細資訊。每個日誌包含收到請求的時間、用戶端的 IP 地址、延遲、請求路徑和伺服器回應等資訊。您可以使用這些存取日誌來分析流量模式和排除問題。

存取日誌是 Elastic Load Balancing 的選用功能，預設為停用。為負載平衡器啟用存取日誌之後，Elastic Load Balancing 會擷取日誌，並將其以壓縮檔案存放在您指定的 Amazon S3 儲存貯體中。您可以隨時停用存取日誌。

您將需支付 Amazon S3 的儲存費用，但 Elastic Load Balancing 將日誌檔傳送到 Amazon S3 所使用的頻寬不需要付費。如需有關儲存費用的詳細資訊，請參閱 [Amazon S3 定價](#)。

目錄

- [存取日誌檔](#)
- [存取日誌項目](#)
- [範例日誌項目](#)
- [處理存取日誌檔](#)
- [為 Application Load Balancer 啟用存取日誌](#)
- [停用 Application Load Balancer 的存取日誌](#)

存取日誌檔

Elastic Load Balancing 每 5 分鐘發佈每個負載平衡器節點的日誌檔。日誌傳遞最終會達到一致。負載平衡器可能在相同期間傳遞多個日誌。這通常是在網站的流量很高時才會發生。

存取日誌的檔案名稱使用以下格式：

```
bucket[/prefix]/AWSLogs/aws-account-id/elasticloadbalancing/region/yyyy/mm/dd/aws-account-id_elasticloadbalancing_region_app.load-balancer-id_end-time_ip-address_random-string.log.gz
```

bucket

S3 儲存貯體的名稱。

prefix

(選用) 儲存貯體的字首 (邏輯階層)。您指定的字首不得包含字串 AWSLogs。如需詳細資訊，請參閱[使用字首組織物件](#)。

AWSLogs

我們在您指定的儲存貯體名稱和可選字首之後，增加了以 AWSLogs 開頭的檔案名稱部分。

aws-account-id

擁有者的 AWS 帳號 ID。

region

負載平衡器和 S3 儲存貯體的區域。

yyyy/mm/dd

傳遞日誌的日期。

load-balancer-id

負載平衡器的資源 ID。如果資源 ID 包含任何斜線 (/)，斜線會換成句點 (.)。

end-time

記錄間隔結束的日期和時間。例如，結束時間 20140215T2340Z 的日子檔案包含在 23:35 和 23:40 (UTC 或 Zulu 時間) 之間發出的請求項目。

ip-address

處理請求之負載平衡器節點的 IP 地址。對於內部負載平衡器，這是私有 IP 地址。

random-string

系統產生的隨機字串。

以下是含字首的日誌檔案名稱範例：

```
s3://my-bucket/my-prefix/AWSLogs/123456789012/elasticloadbalancing/us-east-2/2022/05/01/123456789012_elasticloadbalancing_us-east-2_app.my-loadbalancer.1234567890abcdef_20220215T2340Z_172.160.001.192_20sg8hgm.log.gz
```

以下是不含字首的日誌檔案名稱範例：

```
s3://my-bucket/AWSLogs/123456789012/elasticloadbalancing/us-east-2/2022/05/01/123456789012_elasticloadbalancing_us-east-2_app.my-loadbalancer.1234567890abcdef_20220215T2340Z_172.160.001.192_20sg8hgm.log.gz
```

日誌檔案可存放於儲存貯體任意長時間，但您也可以定義 Amazon S3 生命週期規則，自動封存或刪除日誌檔案。如需詳細資訊，請參閱 Amazon Simple Storage Service 使用者指南中的 [物件生命週期管理](#)。

存取日誌項目

Elastic Load Balancing 會記錄向負載平衡器傳送的請求，包括從未送達目標的請求。例如，如果用戶端傳送格式不正確的請求，或沒有運作狀態良好的目標可回應請求，則仍然會記錄此請求。Elastic Load Balancing 不會記錄運作狀態檢查請求。

每個記錄項目都包含對負載平衡器建立的單一要求 (或連線 WebSockets) 的詳細資訊。對於 WebSockets，只有在關閉連線之後才會寫入項目。如果無法建立升級連線，則項目與 HTTP 或 HTTPS 請求的項目相同。

Important

Elastic Load Balancing 會盡可能記錄請求。建議您使用存取日誌來了解請求的性質，而不是為了全面解釋所有請求。

目錄

- [語法](#)
- [採取的動作](#)
- [分類原因](#)
- [錯誤原因代碼](#)

語法

下表依序說明存取日誌項目的欄位。所有欄位以空格分隔。引進的新欄位會新增到日誌項目尾端。您應該忽略日誌項目尾端任何非預期的欄位。

欄位	Description (描述)
type	<p>請求或連線的類型。可能的值如下所示 (忽略任何其他值)：</p> <ul style="list-style-type: none"> • http – HTTP • https – 透過 TLS 傳輸的 HTTP • h2 – 透過 TLS 傳輸的 HTTP/2 • grpc – 透過 TLS 傳輸的 gRPC • ws – WebSockets • wss – 透過 WebSockets 過 TLS
time	負載平衡器產生回應給用戶端的時間 (ISO 8601 格式)。對於 WebSockets，這是關閉連接的時間。
elb	負載平衡器的資源 ID。如果您剖析存取日誌項目，請注意，資源 ID 可能包含斜線 (/)。
client:port	提出請求之用戶端的 IP 地址和連接埠。如果負載平衡器前面有代理，則此欄位會包含代理的 IP 地址。
target:port	<p>處理此請求之目標的 IP 地址和連接埠。</p> <p>如果用戶端未傳送完整的請求，則負載平衡器無法將請求分派給目標，而這個值會設為 -。</p> <p>如果目標是 Lambda 函數，這個值會設定為 -。</p> <p>如果要求遭到封鎖 AWS WAF，則此值會設定為 -，而 elb_status_code 的值會設定為 403。</p>
request_processing_time	從負載平衡器收到請求到請求傳送到目標為止所經過的總時間 (以秒為單位，精確到毫秒)。

欄位	Description (描述)
	<p>如果負載平衡器無法將請求分派給目標，這個值會設為 -1。如果目標在閒置逾時之前關閉連線，或用戶端傳送格式不正確的請求，就可能發生此情況。</p> <p>如果已註冊的目標在閒置逾時之前沒有回應，這個值也可能設為 -1。</p> <p>如果 AWS WAF 為 Application Load Balancer 啟用，或者目標類型為 Lambda 函數，則用戶端傳送 POST 要求所需資料所需資料所需的時間會計入計入 <code>request_processing_time</code> 。</p>
<code>target_processing_time</code>	<p>從負載平衡器將請求傳送至目標開始，直到目標開始傳送回應標頭為止，所經過的總時間 (以秒為單位，精確到毫秒)。</p> <p>如果負載平衡器無法將請求分派給目標，這個值會設為 -1。如果目標在閒置逾時之前關閉連線，或用戶端傳送格式不正確的請求，就可能發生此情況。</p> <p>如果已註冊的目標在閒置逾時之前沒有回應，這個值也可能設為 -1。</p> <p>如果您 AWS WAF 的應用程式負載平衡器未啟用，用戶端傳送 POST 要求所需資料所需的時間會計入 <code>target_processing_time</code> 。</p>
<code>response_processing_time</code>	<p>從負載平衡器收到目標的回應標頭開始，直到開始將回應傳送到用戶端為止，所經過的總時間 (以秒為單位，精確到毫秒)。這包括負載平衡器上的佇列時間，以及從負載平衡器到用戶端的連線取得時間。</p> <p>如果負載平衡器未收到來自目標的回應，則此值會設為 -1。如果目標在閒置逾時之前關閉連線，或用戶端傳送格式不正確的請求，就可能發生此情況。</p>
<code>elb_status_code</code>	<p>來自負載平衡器的回應狀態碼。</p>
<code>target_status_code</code>	<p>來自目標的回應狀態碼。只有在對目標建立連線且目標傳送回應之後，才會記錄這個值。否則會設為 -。</p>
<code>received_bytes</code>	<p>從用戶端 (請求者) 收到的請求大小 (以位元組為單位)。對於 HTTP 請求，這包括標頭。對於 WebSockets，這是連線上從用戶端接收到的位元組總數。</p>

欄位	Description (描述)
sent_bytes	傳回到用戶端 (請求者) 的回應大小 (以位元組為單位)。對於 HTTP 請求，這包括標頭。對於 WebSockets，這是連線上傳送至用戶端的位元組總數。
"request"	來自用戶端的請求行，以雙引號括住，並採用以下格式來記錄：HTTP 方法 + protocol://host:port/uri+HTTP 版本。記錄請求 URI 時，負載平衡器會依原狀保留用戶端傳送的 URL。它不會為存取日誌檔案設定內容類型。處理此欄位時，請考量用戶端如何傳送 URL。
"user_agent"	User-Agent 字串，識別發出請求的用戶端 (以雙引號括住)。此字串包含一或多個產品識別符，product[/version]。如果字串超過 8 KB，則會截斷。
ssl_cipher	[HTTPS 接聽程式] SSL 加密。如果接聽程式不是 HTTPS 接聽程式，此值會設為 -。
ssl_protocol	[HTTPS 接聽程式] SSL 通訊協定。如果接聽程式不是 HTTPS 接聽程式，此值會設為 -。
target_group_arn	目標群組的 Amazon Resource Name (ARN)。
"trace_id"	X-Amzn-Trace-Id 標頭的內容，以雙引號括住。
"domain_name"	[HTTPS 接聽程式] 在 TLS 交握期間由用戶端提供的 SNI 網域，以雙引號括住。如果用戶端不支援 SNI，或網域不符合憑證而向用戶端出示預設憑證，這個值會設為 -。
"chosen_cert_arn"	[HTTPS 接聽程式] 向用戶端出示的憑證的 ARN，以雙引號括住。如果重複使用工作階段，這個值會設為 session-reused。如果接聽程式不是 HTTPS 接聽程式，此值會設為 -。
matched_rule_priority	符合請求之規則的優先順序值。如果規則符合，則此為 1 到 50,000 的值。如果沒有規則符合且採取預設動作，這個值會設為 0。如果在規則評估期間發生錯誤，則會設為 -1。對於任何其他錯誤，則會設為 -。
request_creation_time	負載平衡器從用戶端收到請求的時間 (ISO 8601 格式)。

欄位	Description (描述)
"actions_executed"	處理請求時所採取的動作，以雙引號括住。這個值是逗號分隔清單，可以包含 採取的動作 中所述的值。如果未採取任何動作，例如對於格式不正確的請求，這個值會設為 -。
"redirect_url"	在 HTTP 回應的位置標頭中，指重新導向目標的 URL，以雙引號括住。如果未採取重新導向動作，這個值會設為 -。
"error_reason"	錯誤原因代碼，以雙引號括住。如果請求失敗，則此為 錯誤原因代碼 所述的其中一個錯誤代碼。如果採取的動作不含驗證動作，或目標不是 Lambda 函數，此值會設為 -。
「target:port_list」	<p>處理此請求之目標之 IP 地址和連接埠以空格分隔的清單，用雙引號括注。目前，列表可以包含一個項目與其匹配的目標：port field。</p> <p>如果用戶端未傳送完整的請求，則負載平衡器無法將請求分派給目標，而這個值會設為 -。</p> <p>如果目標是 Lambda 函數，這個值會設定為 -。</p> <p>如果要求遭到封鎖 AWS WAF，則此值會設定為-，而 elb_status_code 的值會設定為 403。</p>
「target_status_code_list」	<p>目標回應以空格分隔的狀態代碼清單，用雙引號括注。目前，此清單可以包含一個項目與其匹配的 target_status_code 欄位。</p> <p>只有在對目標建立連線且目標傳送回應之後，才會記錄這個值。否則會設為 -。</p>
"classification"	<p>去同步緩解的分類 (以雙引號括住)。如果請求不符合 RFC 7230，可能的值是「可接受」、「不明確」和「嚴重」。</p> <p>如果請求符合 RFC 7230，則此值會設為 -。</p>
"classification_reason"	分類原因代碼 (以雙引號括住)。如果請求不符合 RFC 7230，這是 分類原因 中所述的其中一個分類代碼。如果請求符合 RFC 7230，則此值會設為 -。

欄位	Description (描述)
連續追蹤識別碼	連接追蹤性 ID 是用於識別每個連接的唯一不透明 ID。與用戶端建立連線之後，來自此用戶端的後續要求會在其各自的存取記錄項目中包含此 ID。此 ID 充當外鍵，可在連線和存取記錄之間建立連結。

採取的動作

負載平衡器會將它所採取的動作存放在存取日誌的 `actions_executed` 欄位中。

- `authenticate` – 負載平衡器驗證工作階段、驗證使用者身分，並將使用者資訊新增至請求標頭 (如規則組態所指定)。
- `fixed-response` – 負載平衡器發出固定回應 (如規則組態所指定)。
- `forward` – 負載平衡器將請求轉送至目標 (如規則組態所指定)。
- `redirect` – 負載平衡器將請求重新導向至另一個 URL (如規則組態所指定)。
- `waf` – 負載平衡器將請求轉送至 AWS WAF，以判斷是否要將請求轉送至目標。如果這是最終處理行動，請 AWS WAF 確定要求應拒絕。
- `waf-failed`— 負載平衡器嘗試將要求轉寄給 AWS WAF，但此程序失敗。

分類原因

如果請求不符合 RFC 7230，負載平衡器會在存取日誌的 `classification_reason` 欄位中存放下列其中一個代碼。如需詳細資訊，請參閱 [去同步緩解模式](#)。

代碼	描述	分類
<code>AmbiguousUri</code>	要求 URI 包含控制字元。	不明確
<code>BadContentLength</code>	<code>Content-Length</code> 標頭包含無法剖析或非有效數字的值。	嚴重
<code>BadHeader</code>	標頭包含空值字元或歸位字元。	嚴重
<code>BadTransferEncoding</code>	<code>Transfer-Encoding</code> 標頭包含錯誤的值。	嚴重

代碼	描述	分類
BadUri	要求 URI 包含空值字元或歸位字元。	嚴重
BadMethod	要求方法格式不正確。	嚴重
BadVersion	要求版本格式不正確。	嚴重
BothTeClPresent	要求同時包含 Transfer-Encoding 標頭和 Content-Length 標頭。	不明確
DuplicateContentLength	多個 Content-Length 標頭的值相同。	不明確
EmptyHeader	標頭空白或標頭列僅含空格。	不明確
GetHeadZeroContentLength	GET 或 HEAD 要求的 Content-Length 標頭值為 0。	可接受
MultipleContentLength	多個 Content-Length 標頭的值不同。	嚴重
MultipleTransferEncodingChunked	有多個 Transfer-Encoding : 區塊標頭。	嚴重
NonCompliantHeader	標頭包含非 ASCII 或控制字元。	可接受
NonCompliantVersion	要求版本包含錯誤的值。	可接受
SpaceInUri	要求 URI 包含非 URL 編碼的空格。	可接受
SuspiciousHeader	可使用通用文字正規化技術將標頭正規化為 Transfer-Encoding 或 Content-Length。	不明確

代碼	描述	分類
UndefinedContentLengthSemantics	GET 或 HEAD 請求有定義的 Content-Length 標頭。	不明確
UndefinedTransferEncodingSemantics	GET 或 HEAD 請求有定義的 Transfer-Encoding 標頭。	不明確

錯誤原因代碼

如果負載平衡器無法完成驗證動作，負載平衡器會將以下其中一個原因代碼存放在存取日誌的 `error_reason` 欄位。負載平衡器也會遞增對應的 CloudWatch 指標。如需詳細資訊，請參閱 [使用 Application Load Balancer 來驗證使用者身分](#)。

代碼	描述	指標
AuthInvalidCookie	身分驗證 Cookie 無效。	ELBAuthFailure
AuthInvalidGrantError	來自字符端點的授權碼無效。	ELBAuthFailure
AuthInvalidIdToken	ID 字符無效。	ELBAuthFailure
AuthInvalidStateParam	state 參數無效。	ELBAuthFailure
AuthInvalidTokenResponse	來自字符端點的回應無效。	ELBAuthFailure
AuthInvalidUserInfoResponse	來自使用者資訊端點的回應無效。	ELBAuthFailure

代碼	描述	指標
AuthMissingCodeParam	來自授權端點的身分驗證回應缺少名為 'code' 的查詢參數。	ELBAuthFailure
AuthMissingHostHeader	來自授權端點的身分驗證回應缺少主機標頭欄位。	ELBAuthError
AuthMissingStateParam	來自授權端點的身分驗證回應缺少名為 'state' 的查詢參數。	ELBAuthFailure
AuthTokenEpRequestFailed	字符端點傳回錯誤回應 (非 2XX)。	ELBAuthError
AuthTokenEpRequestTimeout	負載平衡器無法與字符端點通訊。	ELBAuthError
AuthUnhandledException	負載平衡器發生未處理的例外狀況。	ELBAuthError
AuthUserInfoEpRequestFailed	IdP 使用者資訊端點傳回錯誤回應 (非 2XX)。	ELBAuthError
AuthUserInfoEpRequestTimeout	負載平衡器無法與 IdP 使用者資訊端點通訊。	ELBAuthError
AuthUserInfoResponseSizeExceeded	IdP 傳回的宣告大小超過 11K 位元組。	ELBAuthUserClaimsSizeExceeded

如果對權重目標群組的請求失敗，負載平衡器會將以下其中一個錯誤代碼存放在存取日誌的 `error_reason` 欄位。

代碼	描述
AWSALBTGCookieInvalid	與加權目標群組搭配使用的 AWSALBTG Cookie 無效。例如，當 Cookie 值是 URL 編碼時，負載平衡器會傳回此錯誤。
WeightedTargetGroupsUnhandledException	負載平衡器發生未處理的例外狀況。

如果對 Lambda 函數的請求失敗，負載平衡器會將以下其中一個原因代碼存放在存取日誌的 `error_reason` 欄位。負載平衡器也會遞增對應的 CloudWatch 指標。如需詳細資訊，請參閱 [Lambda Invoke](#) 動作。

代碼	描述	指標
LambdaAccessDenied	負載平衡器沒有叫用 Lambda 函數的許可。	LambdaUserError
LambdaBadRequest	Lambda 呼叫失敗，因為用戶端要求標頭或內文不只包含 UTF-8 字元。	LambdaUserError
LambdaConnectionError	負載平衡器無法連線到 Lambda。	LambdaInternalError
LambdaConnectionTimeout	嘗試連接到 Lambda 逾時。	LambdaInternalError
LambdaEC2AccessDeniedException	Amazon EC2 在函數初始化期間拒絕存取 Lambda。	LambdaUserError
LambdaEC2ThrottledException	Amazon EC2 在函數初始化期間對 Lambda 進行節流。	LambdaUserError

代碼	描述	指標
LambdaEC2UnexpectedException	Amazon EC2 在函數初始化期間發生意外例外狀況。	LambdaUserError
LambdaENILimitReachedException	Lambda 無法在 Lambda 函數組態所指定的 VPC 中建立網路介面，因為已超出網路介面數量限制。	LambdaUserError
LambdaInvalidResponse	來自 Lambda 函數的回應格式不正確或缺少所需的欄位。	LambdaUserError
LambdaInvalidRuntimeException	不支援指定的 Lambda 執行期版本。	LambdaUserError
LambdaInvalidSecurityGroupIDException	Lambda 函數之組態中指定的安全群組 ID 無效。	LambdaUserError
LambdaInvalidSubnetIDException	Lambda 函數的組態中指定的子網路 ID 無效。	LambdaUserError
LambdaInvalidZipFileException	Lambda 無法解壓縮指定的函數 zip 檔案。	LambdaUserError
LambdaKMSAccessDeniedException	Lambda 無法解密環境變數，因為對 KMS 金鑰的存取遭拒。請檢查 Lambda 函數的 KMS 許可。	LambdaUserError
LambdaKMSDisabledException	Lambda 無法解密環境變數，因為指定的 KMS 金鑰已停用。請檢查 Lambda 函數的 KMS 金鑰設定。	LambdaUserError

代碼	描述	指標
LambdaKMSInvalidStateException	Lambda 無法解密環境變數，因為 KMS 金鑰的狀態無效。請檢查 Lambda 函數的 KMS 金鑰設定。	LambdaUserError
LambdaKMSNotFoundException	Lambda 無法解密環境變數，因為找不到 KMS 金鑰。請檢查 Lambda 函數的 KMS 金鑰設定。	LambdaUserError
LambdaRequestTooLarge	請求本文的大小超過 1 MB。	LambdaUserError
LambdaResourceNotFound	找不到 Lambda 函數。	LambdaUserError
LambdaResponseTooLarge	回應的大小超過 1 MB。	LambdaUserError
LambdaServiceException	Lambda 發生內部錯誤。	LambdaInternalError
LambdaSubnetIPAddressLimitReachedException	Lambda 無法設定 Lambda 函數的 VPC 存取，因為一個或多個子網路沒有可用的 IP 地址。	LambdaUserError
LambdaThrottling	因為有太多請求，Lambda 函數受到節制。	LambdaUserError
LambdaUnhandled	Lambda 函數發生未處理的例外狀況。	LambdaUserError
LambdaUnhandledException	負載平衡器發生未處理的例外狀況。	LambdaInternalError

代碼	描述	指標
LambdaWebsocketNotSupported	WebSockets 不支援使用 Lambda。	LambdaUserError

如果負載平衡器在將要求轉送至時遇到錯誤 AWS WAF，它會將下列其中一個錯誤碼儲存在存取記錄的 `error_reason` 欄位中。

代碼	描述
WAFConnectionError	負載平衡器無法連線到 AWS WAF。
WAFConnectionTimeout	要 AWS WAF 逾時的連線。
WAFResponseReadTimeout	AWS WAF 逾時的請求。
WAFServiceError	AWS WAF 傳回一個 5XX 錯誤訊息。
WAFUnhandledException	負載平衡器發生未處理的例外狀況。

範例日誌項目

以下為日誌項目範例。請注意，分成多行顯示文字只是為了更輕鬆閱讀。

範例 HTTP 項目

以下是 HTTP 接聽程式的範例日誌項目 (連接埠 80 到連接埠 80)：

```
http 2018-07-02T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
192.168.131.39:2817 10.0.0.1:80 0.000 0.001 0.000 200 200 34 366
"GET http://www.example.com:80/ HTTP/1.1" "curl/7.46.0" - -
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337262-36d228ad5d99923122bbe354" "-" "-"
0 2018-07-02T22:22:48.364000Z "forward" "-" "-" "10.0.0.1:80" "200" "-" "-"
```

範例 HTTPS 項目

以下是 HTTPS 接聽程式的範例日誌項目 (連接埠 443 到連接埠 80) :

```
https 2018-07-02T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
192.168.131.39:2817 10.0.0.1:80 0.086 0.048 0.037 200 200 0 57
"GET https://www.example.com:443/ HTTP/1.1" "curl/7.46.0" ECDHE-RSA-AES128-GCM-SHA256
  TLSv1.2
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337281-1d84f3d73c47ec4e58577259" "www.example.com" "arn:aws:acm:us-
east-2:123456789012:certificate/12345678-1234-1234-1234-123456789012"
1 2018-07-02T22:22:48.364000Z "authenticate,forward" "-" "-" "10.0.0.1:80" "200" "-"
  "-" TID_123456
```

HTTP/2 項目範例

以下是 HTTP/2 串流的範例日誌項目。

```
h2 2018-07-02T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
10.0.1.252:48160 10.0.0.66:9000 0.000 0.002 0.000 200 200 5 257
"GET https://10.0.2.105:773/ HTTP/2.0" "curl/7.46.0" ECDHE-RSA-AES128-GCM-SHA256
  TLSv1.2
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337327-72bd00b0343d75b906739c42" "-" "-"
1 2018-07-02T22:22:48.364000Z "redirect" "https://example.com:80/" "-" "10.0.0.66:9000"
  "200" "-" "-"
```

範例 WebSockets 項目

以下是 WebSockets 連線的範例記錄項目。

```
ws 2018-07-02T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
10.0.0.140:40914 10.0.1.192:8010 0.001 0.003 0.000 101 101 218 587
"GET http://10.0.0.30:80/ HTTP/1.1" "-" - -
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337364-23a8c76965a2ef7629b185e3" "-" "-"
1 2018-07-02T22:22:48.364000Z "forward" "-" "-" "10.0.1.192:8010" "101" "-" "-"
```

安全 WebSockets 入口示例

以下是安全 WebSockets 連線的範例記錄項目。

```
wss 2018-07-02T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
10.0.0.140:44244 10.0.0.171:8010 0.000 0.001 0.000 101 101 218 786
"GET https://10.0.0.30:443/ HTTP/1.1" "-" ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2
arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337364-23a8c76965a2ef7629b185e3" "-" "-"
1 2018-07-02T22:22:48.364000Z "forward" "-" "-" "10.0.0.171:8010" "101" "-" "-"
```

Lambda 函數的範例項目

以下是對 Lambda 函數之請求成功的範例日誌項目：

```
http 2018-11-30T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
192.168.131.39:2817 - 0.000 0.001 0.000 200 200 34 366
"GET http://www.example.com:80/ HTTP/1.1" "curl/7.46.0" - -
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337364-23a8c76965a2ef7629b185e3" "-" "-"
0 2018-11-30T22:22:48.364000Z "forward" "-" "-" "-" "-" "-" "-"
```

以下是對 Lambda 函數之請求失敗的範例日誌項目：

```
http 2018-11-30T22:23:00.186641Z app/my-loadbalancer/50dc6c495c0c9188
192.168.131.39:2817 - 0.000 0.001 0.000 502 - 34 366
"GET http://www.example.com:80/ HTTP/1.1" "curl/7.46.0" - -
arn:aws:elasticloadbalancing:us-east-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067
"Root=1-58337364-23a8c76965a2ef7629b185e3" "-" "-"
0 2018-11-30T22:22:48.364000Z "forward" "-" "LambdaInvalidResponse" "-" "-" "-" "-"
```

處理存取日誌檔

存取日誌檔已壓縮。如您利用 Amazon S3 主控台開啟檔案，則會解壓縮檔案並顯示資訊。如果您下載檔案，則必須先將其解壓縮才能看到資訊。

如果您的網站上有許多需求，負載平衡器產生的日誌檔可能有好幾 GB 的資料。您可能無法使用處理來處理如此大量的資 line-by-line 料。因此，您可能需要使用提供平行處理解決方案的分析工具。例如，您可以使用以下分析工具來分析和處理存取日誌：

- Amazon Athena 是一種互動式查詢服務，可讓您使用標準 SQL 輕鬆分析 Amazon S3 中的資料。如需詳細資訊，請參閱《Amazon Athena 使用者指南》中的[查詢 Application Load Balancer 日誌](#)。
- [Loggly](#)
- [Splunk](#)
- [Sumo Logic](#)

為 Application Load Balancer 啟用存取日誌

為負載平衡器啟用存取記錄時，必須指定供負載平衡器存放日誌的 S3 儲存貯體名稱。儲存貯體必須具有儲存貯體政策，能授予 Elastic Load Balancing 寫入儲存貯體的許可。

任務

- [步驟 1：建立 S3 儲存貯體](#)
- [步驟 2：連接政策到您的 S3 儲存貯體](#)
- [步驟 3：設定存取日誌](#)
- [步驟 4：確認儲存貯體許可](#)
- [故障診斷](#)

步驟 1：建立 S3 儲存貯體

啟用存取日誌時，必須為存取日誌指定 S3 儲存貯體。您可以使用現有儲存貯體，也可以建立專門用於存取日誌的儲存貯體。儲存貯體必須符合下列需求。

要求

- 儲存貯體與負載平衡器必須位於相同的 Region (區域)。儲存貯體和負載平衡器可以由不同的帳戶擁有。
- Amazon S3 受管金鑰 (SSE-S3) 是唯一支援的伺服器端加密選項。如需詳細資訊，請參閱 [Amazon S3 受管加密金鑰 \(SSE-S3\)](#)。

使用 Amazon S3 主控台建立 S3 儲存貯體

1. 前往 <https://console.aws.amazon.com/s3/> 開啟 Amazon S3 主控台。
2. 選擇建立儲存貯體。
3. 在 Create bucket (建立儲存貯體) 頁面上，執行下列操作：

- a. 針對 Bucket name (儲存貯體名稱)，輸入儲存貯體的名稱。該名稱在 Amazon S3 中所有現有的儲存貯體名稱之間，不得重複。在某些區域，可能會對儲存貯體的名稱進行其他限制。如需詳細資訊，請參閱 Amazon Simple Storage Service 使用者指南中的[儲存貯體限制與局限](#)。
- b. 針對 AWS 區域，選取您建立負載平衡器時所在的區域。
- c. 對於預設加密，選擇 Amazon S3 受管金鑰 (SSE-S3)。
- d. 選擇建立儲存貯體。

步驟 2：連接政策到您的 S3 儲存貯體

您的 S3 儲存貯體必須擁有儲存貯體政策，以授權 Elastic Load Balancing 將存取日誌寫入到儲存貯體。儲存貯體政策是以存取政策語言所編寫的 JSON 陳述式集合，可定義儲存貯體的存取許可。每個陳述式包含單一許可的相關資訊，且包含一系列的元素。

如果您目前使用的儲存貯體有已連接的政策，您可以將 Elastic Load Balancing 存取日誌的陳述式加入至政策中。若您這麼做，建議您評估所產生的一組許可，以確保它們適用於需要存取儲存貯體以取得存取日誌的使用者。

可用的儲存貯體政策

您將使用的值區政策取決於區域 AWS 區域 和類型。

2022 年 8 月或之後可用的區域

此政策會將許可授予指定的日誌交付服務。針對下列「區域」內的「可用區域」和「本地區域」中的負載平衡器使用此政策：

- 亞太區域 (海德拉巴)
- 亞太區域 (墨爾本)
- 加拿大西部 (卡加利)
- 歐洲 (西班牙)
- 歐洲 (蘇黎世)
- 以色列 (特拉維夫)
- 中東 (阿拉伯聯合大公國)

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "Service": "logdelivery.elasticloadbalancing.amazonaws.com"
    },
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3:::bucket-name/prefix/AWSLogs/aws-account-id/*"
  }
]
```

2022 年 8 月前可用的區域

此政策會將許可授予指定的 Elastic Load Balancing 帳戶 ID。針對下列清單中「區域」內的「可用區域」或「本地區域」中的負載平衡器使用此政策。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::elb-account-id:root"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::bucket-name/prefix/AWSLogs/aws-account-id/*"
    }
  ]
}
```

將 *elb-Account id* 替換 AWS 帳戶 為您所在地區的 Elastic Load Balancing 的 ID：

- 美國東部 (維吉尼亞北部) – 127311923021
- 美國東部 (俄亥俄) – 033677994240
- 美國西部 (加利佛尼亞北部) – 027434742980
- 美國西部 (奧勒岡) – 797873946194
- 非洲 (開普敦) – 098369216593
- 亞太區域 (香港) – 754344448648

- 亞太區域 (雅加達) – 589379963580
- 亞太區域 (孟買) – 718504428378
- 亞太區域 (大阪) – 383597477331
- 亞太區域 (首爾) – 600734575887
- 亞太區域 (新加坡) – 114774131450
- 亞太區域 (雪梨) – 783225319266
- 亞太區域 (東京) – 582318560864
- 加拿大 (中部) – 985666609251
- 歐洲 (法蘭克福) – 054676820928
- 歐洲 (愛爾蘭) – 156460612806
- 歐洲 (倫敦) – 652711504416
- 歐洲 (米蘭) – 635631232127
- 歐洲 (巴黎) – 009996457667
- 歐洲 (斯德哥爾摩) – 897822967062
- 中東 (巴林) – 076674570225
- 南美洲 (聖保羅) – 507241528517

將 *my-s3-arn* 替換為存取日誌所在位置的 ARN。您指定的 ARN 取決於您是否打算在 [步驟 3](#) 中啟用存取日誌時指定字首。

- 帶有字首的 ARN 範例

```
arn:aws:s3:::bucket-name/prefix/AWSLogs/aws-account-id/*
```

- 不帶字首的 ARN 範例

```
arn:aws:s3:::bucket-name/AWSLogs/aws-account-id/*
```

NotPrincipal何時使Effect用Deny。

如果 Amazon S3 儲存貯體政策Effect與值搭配使用，Deny且包含NotPrincipal如下列範例所示，請確定Service清單中logdelivery.elasticloadbalancing.amazonaws.com已包含該值。

```
{
```

```

"Effect": "Deny",
"NotPrincipal": {
  "Service": [
    "logdelivery.elasticloadbalancing.amazonaws.com",
    "example.com"
  ]
}
},

```

AWS GovCloud (US) Regions

此政策會將許可授予指定的 Elastic Load Balancing 帳戶 ID。針對下列清單中區域中可用區域或 Local Zones 中的負載平衡器使用此原則。AWS GovCloud (US)

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws-us-gov:iam::elb-account-id:root"
      },
      "Action": "s3:PutObject",
      "Resource": "my-s3-arn"
    }
  ]
}

```

將 *elb-Account id* 替換 AWS 帳戶 為您所在地區的 Elastic Load Balancing 的 ID : AWS GovCloud (US)

- AWS GovCloud (美國西部)
- AWS GovCloud (美國東部)

將 *my-s3-arn* 替換為存取日誌所在位置的 ARN。您指定的 ARN 取決於您是否打算在[步驟 3](#) 中啟用存取日誌時指定字首。

- 帶有字首的 ARN 範例

```
arn:aws-us-gov:s3::bucket-name/prefix/AWSLogs/aws-account-id/*
```


- 不帶字首的 ARN 範例

```
arn:aws-us-gov:s3:::bucket-name/AWSLogs/aws-account-id/*
```

Outpost 區域

以下政策會將許可授予指定的日誌交付服務。將此政策用於 Outpost 區域中的負載平衡器。

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "logdelivery.elb.amazonaws.com"
  },
  "Action": "s3:PutObject",
  "Resource": "arn:aws:s3:::bucket-name/prefix/AWSLogs/your-aws-account-id/*",
  "Condition": {
    "StringEquals": {
      "s3:x-amz-acl": "bucket-owner-full-control"
    }
  }
}
```

使用 Amazon S3 主控台，將存取日誌的儲存貯體政策連接到儲存貯體

1. 前往 <https://console.aws.amazon.com/s3/> 開啟的 Amazon Simple Storage Service (Amazon S3) 主控台。
2. 選取儲存貯體的名稱，開啟其詳細資訊頁面。
3. 選擇 Permissions (許可)，然後選擇 Bucket policy (儲存貯體政策)、Edit (編輯)。
4. 更新儲存貯體政策，授予所需許可。
5. 選擇儲存變更。

步驟 3：設定存取日誌

使用以下程序設定存取日誌，擷取並交付日誌檔案到您的 S3 儲存貯體。

要求

儲存貯體必須符合 [步驟 1](#) 中所述的要求，且您必須按照 [步驟 2](#) 所述連接儲存貯體政策。如果您指定首碼，它不得包含字串 "AWSLogs"。

使用主控台為您的負載平衡器啟用存取日誌

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
3. 選取您負載平衡器的名稱來開啟其詳細資訊頁面。
4. 在屬性索引標籤中，選擇編輯。
5. 對於監控，請開啟存取日誌。
6. 針對 S3 URI，請輸入日誌檔案的 S3 URI。指定的 URI 取決於您是否使用字首。
 - 帶有字首的 URI：`s3://bucket-name/prefix`
 - 不帶字首的 URI：`s3://bucket-name`
7. 選擇儲存變更。

若要啟用存取記錄 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令。

管理存取日誌的 S3 儲存貯體

在刪除您為存取日誌設定的儲存貯體之前，請務必停用存取日誌。否則，如果新的儲存貯體有相同名稱，且所需的儲存貯體政策是在您未擁有的 AWS 帳戶中建立，則 Elastic Load Balancing 可能會將負載平衡器的存取日誌寫入這個新的儲存貯體。

步驟 4：確認儲存貯體許可

為負載平衡器啟用存取日誌之後，Load Balancing 會驗證 S3 儲存貯體，並建立測試檔案，以確保儲存貯體政策指定所需的許可。您可以使用 Amazon S3 主控台來確認是否已建立測試檔案。測試檔案不是實際的存取日誌檔案；它不包含範例記錄。

驗證 Elastic Load Balancing 已在 S3 儲存貯體中建立測試檔案

1. 前往 <https://console.aws.amazon.com/s3/> 開啟的 Amazon Simple Storage Service (Amazon S3) 主控台。
2. 選取您為存取日誌指定的儲存貯體名稱。
3. 導覽到測試檔案，ELBAccessLogTestFile。位置取決於您是否使用字首。
 - 帶字首的位置：`my-bucket/prefix/AWSLogs/123456789012/ELBAccessLogTestFile`
 - 不帶字首的位置：`my-bucket/AWSLogs/123456789012/ELBAccessLogTestFile`

故障診斷

如果您收到存取遭拒錯誤，則以下是可能的原因：

- 儲存貯體政策不會授權 Elastic Load Balancing 將存取日誌寫入儲存貯體。確認您正在使用適合該區域的正確儲存貯體政策。確認資源 ARN 使用您在啟用存取日誌時指定的相同儲存貯體名稱。如果啟用存取日誌時未指定字首，則請確認資源 ARN 不包含字首。
- 儲存貯體使用不支援的伺服器端加密選項。儲存貯體必須使用 Amazon S3 受管金鑰 (SSE-S3)。

停用 Application Load Balancer 的存取日誌

您可以隨時對負載平衡器停用存取日誌。停用存取日誌之後，存取日誌會保留在 S3 儲存貯體中，直到您刪除為止。如需詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的[使用儲存貯體](#)。

使用主控台停用存取日誌

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
3. 選取您負載平衡器的名稱來開啟其詳細資訊頁面。
4. 在屬性索引標籤中，選擇編輯。
5. 針對監控，請關閉存取日誌。
6. 選擇儲存變更。

若要停用存取記錄，請使用 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令。

Application Load Balancer 的連線記錄

Elastic Load Balancing 提供連線記錄，以擷取傳送至負載平衡器之要求的詳細資訊。每個記錄檔都包含諸如用戶端的 IP 位址和連接埠、監聽器連接埠、使用的 TLS 加密和通訊協定、TLS 交握延遲、連線狀態以及用戶端憑證詳細資料等資訊。您可以使用這些連線記錄來分析要求模式並疑難排解問題。

連線記錄是 Elastic Load Balancing 的選用功能，預設為停用。啟用負載平衡器的連線日誌後，Elastic Load Balancing 會擷取日誌，並將其作為壓縮檔存放在您指定的 Amazon S3 儲存貯體中。您可以隨時停用連線記錄。

您將需支付 Amazon S3 的儲存費用，但 Elastic Load Balancing 將日誌檔傳送到 Amazon S3 所使用的頻寬不需要付費。如需有關儲存費用的詳細資訊，請參閱 [Amazon S3 定價](#)。

目錄

- [連線記錄檔](#)
- [連線日誌項目](#)
- [範例日誌項目](#)
- [處理連線記錄檔](#)
- [啟用應用 Application Load Balancer 的連線記錄](#)
- [停用應用 Application Load Balancer 的連線記錄](#)

連線記錄檔

Elastic Load Balancing 每 5 分鐘發佈每個負載平衡器節點的日誌檔。日誌傳遞最終會達到一致。負載平衡器可能在相同期間傳遞多個日誌。這通常是在網站的流量很高時才會發生。

連線記錄檔的檔案名稱使用下列格式：

```
bucket[/prefix]/AWSLogs/aws-account-id/elasticloadbalancing/region/yyyy/mm/dd/  
conn_log.aws-account-id_elasticloadbalancing_region_app.load-balancer-id_end-time_ip-  
address_random-string.log.gz
```

bucket

S3 儲存貯體的名稱。

prefix

(選用) 儲存貯體的字首 (邏輯階層)。您指定的字首不得包含字串 AWSLogs。如需詳細資訊，請參閱 [使用字首組織物件](#)。

AWSLogs

我們在您指定的儲存貯體名稱和可選字首之後，增加了以 AWSLogs 開頭的檔案名稱部分。

aws-account-id

擁有者的 AWS 帳號 ID。

region

負載平衡器和 S3 儲存貯體的區域。

yyyy/mm/dd

傳遞日誌的日期。

load-balancer-id

負載平衡器的資源 ID。如果資源 ID 包含任何斜線 (/)，斜線會換成句點 (.)。

end-time

記錄間隔結束的日期和時間。例如，結束時間 20140215T2340Z 的日子檔案包含在 23:35 和 23:40 (UTC 或 Zulu 時間) 之間發出的請求項目。

ip-address

處理請求之負載平衡器節點的 IP 地址。對於內部負載平衡器，這是私有 IP 地址。

random-string

系統產生的隨機字串。

以下是含字首的日誌檔案名稱範例：

```
s3://my-bucket/my-prefix/AWSLogs/123456789012/elasticloadbalancing/us-east-2/2022/05/01/conn_log.123456789012_elasticloadbalancing_us-east-2_app.my-loadbalancer.1234567890abcdef_20220215T2340Z_172.160.001.192_20sg8hgm.log.gz
```

以下是不含字首的日誌檔案名稱範例：

```
s3://my-bucket/AWSLogs/123456789012/elasticloadbalancing/us-east-2/2022/05/01/conn_log.123456789012_elasticloadbalancing_us-east-2_app.my-loadbalancer.1234567890abcdef_20220215T2340Z_172.160.001.192_20sg8hgm.log.gz
```

日誌檔案可存放於儲存貯體任意長時間，但您也可以定義 Amazon S3 生命週期規則，自動封存或刪除日誌檔案。如需詳細資訊，請參閱 Amazon Simple Storage Service 使用者指南中的[物件生命週期管理](#)。

連線日誌項目

每次連線嘗試在連線記錄檔中都有一個項目。用戶端要求的傳送方式取決於連線為持續性或非持續性連線。非持續連線具有單一要求，會在存取記錄和連線記錄中建立單一項目。持續連線有多個要求，會在存取記錄檔中建立多個項目，並在連線記錄中建立單一項目。

目錄

- [語法](#)
- [錯誤原因代碼](#)

語法

連線記錄項目使用下列格式：

```
[timestamp] [client_ip] [client_port] [listener_port] [tls_protocol] [tls_cipher]
[tls_handshake_latency] [leaf_client_cert_subject] [leaf_client_cert_validity]
[leaf_client_cert_serial_number] [tls_verify_status]
```

下表依序說明連線記錄項目的欄位。所有欄位以空格分隔。引進的新欄位會新增到日誌項目尾端。您應該忽略日誌項目尾端任何非預期的欄位。

欄位	描述
timestamp	負載平衡器成功建立或無法建立連線時的時間 (使用 ISO 8601 格式)。
用戶端 IP	要求用戶端的 IP 位址。
用戶端口	要求用戶端的連接埠。
監聽器端口	接收用戶端要求之負載平衡器接聽程式的連接埠。
tls_ 通訊協定	[HTTPS 接聽程式] 握手期間使用的 SSL/TLS 通訊協定。-針對非 SSL/TLS 要求，此欄位會設定為。
tls_cipher	[HTTPS 接聽程式] 握手期間使用的 SSL/TLS 通訊協定。-針對非 SSL/TLS 要求，此欄位會設定為。
tls_ 手動延遲	[HTTPS 接聽程式] 建立成功交握時經過的總時間 (以秒為單位)，精確度為毫秒。此欄位設定為下列-時間： <ul style="list-style-type: none"> • 傳入的要求不是 SSL/TLS 要求。 • 握手未成功建立。
葉客戶證書主旨	[HTTPS 接聽程式] 分葉用戶端憑證的主旨名稱。此欄位設定為下列-時間：

欄位	描述
	<ul style="list-style-type: none"> 傳入的要求不是 SSL/TLS 要求。 未在啟用 MTL 的情況下設定負載平衡器接聽程式。 伺服器無法載入/剖析分葉用戶端憑證。
葉客戶證書有效性	<p>[HTTPS 接聽程式] 分葉用戶端憑證not-after 的有效性，以not-before 及 ISO 8601 格式的有效性。此欄位設定為下列-時間：</p> <ul style="list-style-type: none"> 傳入的要求不是 SSL/TLS 要求。 未在啟用 MTL 的情況下設定負載平衡器接聽程式。 伺服器無法載入/剖析分葉用戶端憑證。
葉客戶端 _ 證書 _ 序列號	<p>[HTTPS 接聽程式] 分葉用戶端憑證的序號。此欄位設定為下列-時間：</p> <ul style="list-style-type: none"> 傳入的要求不是 SSL/TLS 要求。 未在啟用 MTL 的情況下設定負載平衡器接聽程式。 伺服器無法載入/剖析分葉用戶端憑證。
驗證狀態	<p>[HTTPS 接聽程式] 連線要求的狀態。Success如果連線已成功建立，則此值為。在不成功的連線上，值為Failed:\$error_code 。</p>
連續追蹤識別碼	<p>連接追蹤性 ID 是用於識別每個連接的唯一不透明 ID。與用戶端建立連線之後，來自此用戶端的後續要求會在其各自的存取記錄項目中包含此 ID。此 ID 充當外鍵，可在連線和存取記錄之間建立連結。</p>

錯誤原因代碼

如果負載平衡器無法建立連線，負載平衡器會在連線記錄中儲存下列其中一個原因代碼。

代碼	描述
ClientCertificateMaxChainDepthExceeded	已超過最大用戶端憑證鏈結深度

代碼	描述
ClientCertificateMaxSizeExceeded	已超過用戶端憑證大小上限
ClientCertificateCrlHit	用戶端憑證已被 CA 撤銷
ClientCertificateCrlProcessingError	CRL 處理錯誤
ClientCertificateUntrusted	用戶端憑證不受信任
ClientCertificateNotYetValid	用戶端憑證尚無效
ClientCertificateExpired	用戶端憑證已過期
ClientCertificateTypeUnsupported	不支援用戶端憑證類型
ClientCertificateInvalid	用戶端憑證無效
ClientCertificateRejected	客戶端證書被自定義服務器驗證拒絕
UnmappedConnectionError	未映射的運行時連接錯誤

範例日誌項目

以下是連線記錄項目的範例。

以下是成功連線至 HTTPS 接聽程式的範例記錄項目，且已在連接埠 443 上啟用相互 TLS 驗證模式：

```
2023-10-04T17:05:15.514108Z 203.0.113.1 36280 443 TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 4.036 "CN=amazondomains.com,O=endEntity,L=Seattle,ST=Washington,C=US"
NotBefore=2023-09-21T22:43:21Z;NotAfter=2026-06-17T22:43:21Z FEF257372D5C14D4 Success
```

以下是在連接埠 443 上啟用相互 TLS 驗證模式的 HTTPS 接聽程式連線失敗的範例記錄項目。：

```
2023-10-04T17:05:15.514108Z 203.0.113.1 36280 443 TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 - "CN=amazondomains.com,O=endEntity,L=Seattle,ST=Washington,C=US"
NotBefore=2023-09-21T22:43:21Z;NotAfter=2026-06-17T22:43:21Z FEF257372D5C14D4
Failed:ClientCertUntrusted
```

處理連線記錄檔

連線記錄檔即會壓縮。如您利用 Amazon S3 主控台開啟檔案，則會解壓縮檔案並顯示資訊。如果您下載檔案，則必須先將其解壓縮才能看到資訊。

如果您的網站上有許多需求，負載平衡器產生的日誌檔可能有好幾 GB 的資料。您可能無法使用處理來處理如此大量的資料。因此，您可能需要使用提供平行處理解決方案的分析工具。例如，您可以使用下列分析工具來分析和處理連線記錄：

- Amazon Athena 是一種互動式查詢服務，可讓您使用標準 SQL 輕鬆分析 Amazon S3 中的資料。
- [Loggly](#)
- [Splunk](#)
- [Sumo Logic](#)

啟用應用 Application Load Balancer 的連線記錄

啟用負載平衡器的連線日誌時，您必須指定負載平衡器將在其中存放日誌的 S3 儲存貯體的名稱。儲存貯體必須具有儲存貯體政策，能授予 Elastic Load Balancing 寫入儲存貯體的許可。

任務

- [步驟 1：建立 S3 儲存貯體](#)
- [步驟 2：連接政策到您的 S3 儲存貯體](#)
- [步驟 3：設定連線記錄](#)
- [步驟 4：確認儲存貯體許可](#)

• [故障診斷](#)

步驟 1：建立 S3 儲存貯體

啟用連線記錄時，您必須為連線記錄指定 S3 儲存貯體。您可以使用現有值區，也可以針對連線記錄建立儲存貯體。儲存貯體必須符合下列需求。

要求

- 儲存貯體與負載平衡器必須位於相同的 Region (區域)。儲存貯體和負載平衡器可以由不同的帳戶擁有。
- Amazon S3 受管金鑰 (SSE-S3) 是唯一支援的伺服器端加密選項。如需詳細資訊，請參閱 [Amazon S3 受管加密金鑰 \(SSE-S3\)](#)。

使用 Amazon S3 主控台建立 S3 儲存貯體

1. 前往 <https://console.aws.amazon.com/s3/> 開啟 Amazon S3 主控台。
2. 選擇建立儲存貯體。
3. 在 Create bucket (建立儲存貯體) 頁面上，執行下列操作：
 - a. 針對 Bucket name (儲存貯體名稱)，輸入儲存貯體的名稱。該名稱在 Amazon S3 中所有現有的儲存貯體名稱之間，不得重複。在某些區域，可能會對儲存貯體的名稱進行其他限制。如需詳細資訊，請參閱 Amazon Simple Storage Service 使用者指南中的 [儲存貯體限制與局限](#)。
 - b. 針對 AWS 區域，選取您建立負載平衡器時所在的區域。
 - c. 對於預設加密，選擇 Amazon S3 受管金鑰 (SSE-S3)。
 - d. 選擇建立儲存貯體。

步驟 2：連接政策到您的 S3 儲存貯體

您的 S3 儲存貯體必須具有授與 Elastic Load Balancing 權限的儲存貯體政策，才能將連線日誌寫入儲存貯體。儲存貯體政策是以存取政策語言所編寫的 JSON 陳述式集合，可定義儲存貯體的存取許可。每個陳述式包含單一許可的相關資訊，且包含一系列的元素。

如果您使用的是已附加政策的現有值區，則可以將 Elastic Load Balancing 連線記錄的陳述式新增至原則。如果您這麼做，我們建議您評估產生的一組權限，以確保它們適合需要存取值區以存取連線記錄的使用者。

可用的儲存貯體政策

您將使用的值區政策取決於區域 AWS 區域 和類型。

2022 年 8 月或之後可用的區域

此政策會將許可授予指定的日誌交付服務。針對下列「區域」內的「可用區域」和「本地區域」中的負載平衡器使用此政策：

- 亞太區域 (海德拉巴)
- 亞太區域 (墨爾本)
- 歐洲 (西班牙)
- 歐洲 (蘇黎世)
- 以色列 (特拉維夫)
- 中東 (阿拉伯聯合大公國)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "logdelivery.elasticloadbalancing.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::bucket-name/prefix/AWSLogs/aws-account-id/*"
    }
  ]
}
```

2022 年 8 月前可用的區域

此政策會將許可授予指定的 Elastic Load Balancing 帳戶 ID。針對下列清單中「區域」內的「可用區域」或「本地區域」中的負載平衡器使用此政策。

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

```
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::elb-account-id:root"
    },
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3::bucket-name/prefix/AWSLogs/aws-account-id/*"
  }
]
```

將 *elb-Account id* 替換 AWS 帳戶 為您所在地區的 Elastic Load Balancing 的 ID :

- 美國東部 (維吉尼亞北部) – 127311923021
- 美國東部 (俄亥俄) – 033677994240
- 美國西部 (加利佛尼亞北部) – 027434742980
- 美國西部 (奧勒岡) – 797873946194
- 非洲 (開普敦) – 098369216593
- 亞太區域 (香港) – 754344448648
- 亞太區域 (雅加達) – 589379963580
- 亞太區域 (孟買) – 718504428378
- 亞太區域 (大阪) – 383597477331
- 亞太區域 (首爾) – 600734575887
- 亞太區域 (新加坡) – 114774131450
- 亞太區域 (雪梨) – 783225319266
- 亞太區域 (東京) – 582318560864
- 加拿大 (中部) – 985666609251
- 歐洲 (法蘭克福) – 054676820928
- 歐洲 (愛爾蘭) – 156460612806
- 歐洲 (倫敦) – 652711504416
- 歐洲 (米蘭) – 635631232127
- 歐洲 (巴黎) – 009996457667
- 歐洲 (斯德哥爾摩) – 897822967062
- 中東 (巴林) – 076674570225

- 南美洲 (聖保羅) – 507241528517
- AWS GovCloud (美國西部)
- AWS GovCloud (美國東部) — 一九五六九 1635

將 *my-s3-arn* 替換為連接日誌的位置的 ARN。您指定的 ARN 取決於您是否打算在 [步驟 3](#) 中啟用連線記錄時指定前置詞。

- 帶有字首的 ARN 範例

```
arn:aws:s3::bucket-name/prefix/AWSLogs/aws-account-id/*
```

- 不帶字首的 ARN 範例

```
arn:aws:s3::bucket-name/AWSLogs/aws-account-id/*
```

NotPrincipal何時使Effect用Deny。

如果 Amazon S3 儲存貯體政策Effect與值搭配使用，Deny並且包含NotPrincipal如以下範例所示，請確保logdelivery.elasticloadbalancing.amazonaws.com該值包含在Service清單中。

```
{
  "Effect": "Deny",
  "NotPrincipal": {
    "Service": [
      "logdelivery.elasticloadbalancing.amazonaws.com",
      "example.com"
    ]
  },
}
```

使用 Amazon S3 主控台將連線日誌的儲存貯體政策附加到儲存貯體

1. 前往 <https://console.aws.amazon.com/s3/> 開啟的 Amazon Simple Storage Service (Amazon S3) 主控台。
2. 選取儲存貯體的名稱，開啟其詳細資訊頁面。
3. 選擇 Permissions (許可)，然後選擇 Bucket policy (儲存貯體政策)、Edit (編輯)。
4. 更新儲存貯體政策，授予所需許可。
5. 選擇儲存變更。

步驟 3：設定連線記錄

使用下列程序設定連線日誌，以擷取日誌檔並將其傳遞到 S3 儲存貯體。

要求

儲存貯體必須符合**步驟 1**中所述的要求，且您必須按照**步驟 2**所述連接儲存貯體政策。如果您指定首碼，它不得包含字串 "AWSLogs"。

使用主控台為負載平衡器啟用連線記錄

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
3. 選取您負載平衡器的名稱來開啟其詳細資訊頁面。
4. 在屬性索引標籤中，選擇編輯。
5. 對於監控，請開啟連線記錄。
6. 針對 S3 URI，請輸入日誌檔案的 S3 URI。指定的 URI 取決於您是否使用字首。
 - 帶有字首的 URI : `s3://bucket-name/prefix`
 - 不帶字首的 URI : `s3://bucket-name`
7. 選擇儲存變更。

若要啟用連線記錄 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令。

管理連線日誌的 S3 儲存貯體

在刪除您為連線記錄設定的值區之前，請務必停用連線記錄。否則，如果存在具有相同名稱和所需值區策略的新值區，但在您不擁有 AWS 帳戶 的值區中建立，則 Elastic Load Balancing 可能會將負載平衡器的連線記錄寫入此新值區。

步驟 4：確認儲存貯體許可

為負載平衡器啟用連線日誌後，Elastic Load Balancing 會驗證 S3 儲存貯體並建立測試檔案，以確保儲存貯體政策指定了必要的許可。您可以使用 Amazon S3 主控台來確認是否已建立測試檔案。測試檔案不是實際的連線記錄檔；它不包含範例記錄。

驗證 Elastic Load Balancing 已在 S3 儲存貯體中建立測試檔案

1. 前往 <https://console.aws.amazon.com/s3/> 開啟的 Amazon Simple Storage Service (Amazon S3) 主控台。
2. 選取您為連線記錄指定的值區名稱。
3. 導覽到測試檔案，ELBConnectionLogTestFile。位置取決於您是否使用字首。
 - 帶字首的位置：*my-bucket/prefix/AWSLogs/123456789012/ELBConnectionLogTestFile*
 - 不帶字首的位置：*my-bucket/AWSLogs/123456789012/ELBConnectionLogTestFile*

故障診斷

如果您收到存取遭拒錯誤，則以下是可能的原因：

- 值區政策未授與 Elastic Load Balancing 寫入值區的連線記錄的權限。確認您正在使用適合該區域的正確儲存貯體政策。確認資源 ARN 使用您啟用連線記錄時指定的相同值區名稱。如果啟用連線記錄時未指定前置詞，請確認資源 ARN 不包含前置詞。
- 儲存貯體使用不支援的伺服器端加密選項。儲存貯體必須使用 Amazon S3 受管金鑰 (SSE-S3)。

停用應用 Application Load Balancer 的連線記錄

您可以隨時停用負載平衡器的連線記錄。停用連線日誌後，您的連線日誌會保留在 S3 儲存貯體中，直到您將其刪除為止。如需詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的[使用儲存貯體](#)。

使用主控台停用連線記錄

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Load Balancers (負載平衡器)。
3. 選取您負載平衡器的名稱來開啟其詳細資訊頁面。
4. 在屬性索引標籤中，選擇編輯。
5. 對於監控，請關閉連線記錄。
6. 選擇儲存變更。

若要停用連線記錄 AWS CLI

使用 [modify-load-balancer-attributes](#) 命令。

Application Load Balancer 上的請求追蹤

當負載平衡器收到用戶端的請求時，在將請求傳送到目標之前，它會新增或更新 X-Amzn-Trace-Id 標頭。負載平衡器和目標之間的任何服務或應用程式也可以新增或更新此標頭。

您可以使用請求追蹤來追蹤從用戶端到目標或其他服務的 HTTP 請求。如果您啟用存取日誌，則會記錄 X-Amzn-Trace-Id 標頭的內容。如需詳細資訊，請參閱 [Application Load Balancer 的存取日誌](#)。

語法

X-Amzn-Trace-Id 標頭包含如下格式的欄位：

```
Field=version-time-id
```

欄位

欄位的名稱。支援的值為 Root 和 Self。

應用程式可以新增任意欄位供自己使用。負載平衡器會保留這些欄位，但不使用。

version

版本號碼。

time

epoch 時間 (以秒為單位)。

id

追蹤識別符。

範例

如果傳入請求上沒有 X-Amzn-Trace-Id 標頭，負載平衡器會產生含有 Root 欄位的標頭，然後轉送請求。例如：

```
X-Amzn-Trace-Id: Root=1-67891233-abcdef012345678912345678
```

如果 X-Amzn-Trace-Id 標頭存在且有 Root 欄位，負載平衡器會插入 Self 欄位，然後轉送請求。例如：


```
X-Amzn-Trace-Id: Self=1-67891233-12456789abcdef012345678;Root=1-67891233-abcdef012345678912345678
```

如果應用程式新增含有 Root 欄位和自訂欄位的標頭，負載平衡器會保留這兩個欄位、插入 Self 欄位，然後轉送請求：

```
X-Amzn-Trace-Id: Self=1-67891233-12456789abcdef012345678;Root=1-67891233-abcdef012345678912345678;CalledFrom=app
```

如果 X-Amzn-Trace-Id 標頭存在且有 Self 欄位，負載平衡器會更新 Self 欄位的值。

限制

- 負載平衡器會在收到傳入請求時，而不是在收到回應時更新標頭。
- 如果 HTTP 標頭大於 7 KB，負載平衡器會使用 Root 欄位重寫 X-Amzn-Trace-Id 標頭。
- 使用時 WebSockets，您只能追蹤升級要求成功為止。

使用 AWS CloudTrail 記錄 Application Load Balancer 的 API 呼叫

Elastic Load Balancing 與提供使用者 AWS CloudTrail、角色或服務在 Elastic Load Balancing 中採取的動作記錄的 AWS 服務整合。CloudTrail 將 Elastic Load Balancing 的所有 API 呼叫擷取為事件。擷取的呼叫包括來自 Elastic Load Balancing API 作業的呼叫 AWS Management Console 和程式碼呼叫。如果您建立追蹤，您可以啟用持續向 Amazon S3 儲存貯體傳遞 CloudTrail 事件，包括 Elastic Load Balancing 的事件。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。使用收集的資訊 CloudTrail，您可以判斷對 Elastic Load Balancing 提出的要求、提出要求的 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱 [AWS CloudTrail 用戶指南](#)。

若要監控負載平衡器的其他動作，例如當用戶端對負載平衡器提出請求時，請使用存取日誌。如需詳細資訊，請參閱 [Application Load Balancer 的存取日誌](#)。

Elastic Load Balancing 資訊 CloudTrail

CloudTrail 在您創建 AWS 帳戶時，您的帳戶已啟用。當活動在 Elastic Load Balancing 中發生時，該活動會與 CloudTrail 事件歷史記錄中的其他 AWS 服務事件一起記錄在事件中。您可以在帳戶中查看，搜索和下載最近的事 AWS 件。如需詳細資訊，請參閱 [使用 CloudTrail 事件歷程記錄檢視事件](#)。

如需 AWS 帳戶中持續的事件記錄 (包括 Elastic Load Balancing 的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。根據預設，當您在主控台中建立追蹤時，追蹤會套用至所有 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務與整合](#)
- [設定 Amazon SNS 通知 CloudTrail](#)
- [從多個區域接收 CloudTrail 日誌文件並從多個帳戶接收 CloudTrail 日誌文件](#)

應用程式負載平衡器的所有 Elastic Load Balancing 動作都會記錄下來，CloudTrail 並記錄在 [Elastic Load Balancing API 參考版本 2015-12-01](#) 中。例如，呼叫 `CreateLoadBalancer` 和 `DeleteLoadBalancer` 動作會在 CloudTrail 記錄檔中產生項目。

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：

- 是否使用根或憑證發出請求。
- 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。
- 該請求是否由另一項 AWS 服務服務提出。

如需詳細資訊，請參閱 [CloudTrail 使用 userIdentity 元素](#)。

了解 Elastic Load Balancing 日誌檔案項目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。CloudTrail 記錄檔包含一或多個記錄項目。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 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

日誌文件包括您的所有 AWS API 調用的事件 AWS 帳戶，而不僅僅是 Elastic Load Balancing API 調用。您可以透過 `elasticloadbalancing.amazonaws.com` 值檢查 `eventSource` 元素，將呼叫定位至 Elastic Load Balancing API。若要檢視關於特定動作的紀錄，例如 `CreateLoadBalancer`，請透過動作名稱檢查 `eventName` 元素。

以下是建立應 CloudTrail 用程式負載平衡器，然後使用將其刪除的使用者的 Elastic Load Balancing 範例記錄 AWS CLI。您可以使用 `userAgent` 元素來識別 CLI。您可以使用 `eventName` 元素來識別請求的 API 呼叫。使用者 (Alice) 的相關資訊則可在 `userIdentity` 元素中找到。

Example 範例：CreateLoadBalancer

```
{
  "eventVersion": "1.03",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:user/Alice",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Alice"
  },
  "eventTime": "2016-04-01T15:31:48Z",
  "eventSource": "elasticloadbalancing.amazonaws.com",
  "eventName": "CreateLoadBalancer",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "198.51.100.1",
  "userAgent": "aws-cli/1.10.10 Python/2.7.9 Windows/7 botocore/1.4.1",
  "requestParameters": {
    "subnets": ["subnet-8360a9e7","subnet-b7d581c0"],
    "securityGroups": ["sg-5943793c"],
    "name": "my-load-balancer",
    "scheme": "internet-facing"
  },
  "responseElements": {
    "loadBalancers": [{
      "type": "application",
      "loadBalancerName": "my-load-balancer",
      "vpcId": "vpc-3ac0fb5f",
      "securityGroups": ["sg-5943793c"],
      "state": {"code": "provisioning"},
      "availabilityZones": [
        {"subnetId": "subnet-8360a9e7", "zoneName": "us-west-2a"},
        {"subnetId": "subnet-b7d581c0", "zoneName": "us-west-2b"}
      ],
      "dnsName": "my-load-balancer-1836718677.us-west-2.elb.amazonaws.com",
      "canonicalHostedZoneId": "Z2P70J7HTTTPLU",
      "createdTime": "Apr 11, 2016 5:23:50 PM",
```

```

        "loadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/ffcddace1759e1d0",
        "scheme": "internet-facing"
    }]
},
"requestID": "b9960276-b9b2-11e3-8a13-f1ef1EXAMPLE",
"eventID": "6f4ab5bd-2daa-4d00-be14-d92efEXAMPLE",
"eventType": "AwsApiCall",
"apiVersion": "2015-12-01",
"recipientAccountId": "123456789012"
}

```

Example 範例 : DeleteLoadBalancer

```

{
  "eventVersion": "1.03",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:user/Alice",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Alice"
  },
  "eventTime": "2016-04-01T15:31:48Z",
  "eventSource": "elasticloadbalancing.amazonaws.com",
  "eventName": "DeleteLoadBalancer",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "198.51.100.1",
  "userAgent": "aws-cli/1.10.10 Python/2.7.9 Windows/7 boto-core/1.4.1",
  "requestParameters": {
    "loadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/ffcddace1759e1d0"
  },
  "responseElements": null,
  "requestID": "349598b3-000e-11e6-a82b-298133eEXAMPLE",
  "eventID": "75e81c95-4012-421f-a0cf-babdaEXAMPLE",
  "eventType": "AwsApiCall",
  "apiVersion": "2015-12-01",
  "recipientAccountId": "123456789012"
}

```

為 Application Load Balancer 進行疑難排解

以下資訊有助於您就 Application Load Balancer 的問題進行疑難排解。

問題

- [已註冊目標處於非服務中狀態](#)
- [用戶端無法連接到面向網際網路的負載平衡器](#)
- [負載平衡器不會收到傳送至自訂域的請求](#)
- [傳送至負載平衡器的 HTTPS 要求會傳回 "NET::ERR_CERT_COMMON_NAME_INVALID"](#)
- [負載平衡器顯示處理時間延長](#)
- [負載平衡器會傳送 000 的回應代碼](#)
- [負載平衡器產生 HTTP 錯誤](#)
- [目標產生了 HTTP 錯誤](#)
- [AWS Certificate Manager 憑證無法使用](#)
- [不支援多行標頭](#)
- [使用資源對應疑難排解狀況不良的目標](#)

已註冊目標處於非服務中狀態

如果目標進入 InService 狀態所花的時間超過預期，表示該目標可能未通過運作狀態檢查。您的目標將處於非服務中狀態，除非通過一次運作狀態檢查。如需詳細資訊，請參閱 [目標群組運作狀態檢查](#)。

確認您的執行個體是否未通過運作狀態檢查，然後檢查以下問題：

安全群組不允許流量

與執行個體相關聯的安全群組，必須允許來自負載平衡器使用運作狀態檢查連接埠和運作狀態檢查通訊協定傳來的流量。您可以將規則新增到執行個體安全群組，以允許來自負載平衡器安全群組的所有流量。此外，負載平衡器的安全群組必須允許對執行個體的流量。

網路存取控制清單 (ACL) 不允許流量

與執行個體的子網路相關聯的網路 ACL 必須允許透過運作狀態檢查連接埠傳送對內流量，以及透過暫時性連接埠 (1024-65535) 傳送對外流量。與負載平衡器節點的子網路相關聯的網路 ACL 必須允許透過暫時性連接埠傳送對內流量，以及透過運作狀態檢查連接埠和暫時性連接埠傳送對外流量。

ping 路徑不存在

針對運作狀態檢查建立目標頁面，並指定其路徑做為 ping 路徑。

連線逾時

首先，驗證您可以使用目標的私有 IP 地址和運作狀態檢查通訊協定，從網路內直接連接到目標。如果無法連接，請檢查是否過度利用該執行個體，如果它太忙碌而無法回應，便將更多目標新增到您的目標群組。如果您可以連接，則在運作狀態檢查逾時期間之前，目標頁面可能都沒有回應。為運作狀態檢查選擇較簡單的目標頁面，或調整運作狀態檢查設定。

目標未傳回成功的回應代碼

在預設情況下，成功代碼是 200，但您可以在設定運作狀態檢查時選擇性地指定額外的成功代碼。確認負載平衡器預期的成功代碼，並且您的應用程式已設定為在成功時傳回這些代碼。

目標回應代碼錯誤或連線至目標時發生錯誤

確認應用程式是否回應負載平衡器的運作狀態檢查請求。某些應用程式需要額外的組態才能回應運作狀態檢查，例如需要有虛擬主機組態才能回應負載平衡器傳送的 HTTP 主機標頭。主機標頭值包含目標的私人 IP 位址，不使用預設連接埠時，接著健全狀況檢查連接埠。如果目標使用預設健全狀況檢查連接埠，則主機標頭值僅包含目標的私人 IP 位址。例如，如果目標的私有 IP 位址是 10.0.0.10 且其健康狀態檢查連接埠為 8080，則負載平衡器在健康狀態檢查中傳送的 HTTP Host 標頭為 Host: 10.0.0.10:8080。如果您的目標的私有 IP 地址是 10.0.0.10 並且它的健康檢查端口是 80 那麼負載平衡器在運行狀態檢查中發送的 HTTP Host 標頭是 Host: 10.0.0.10。可能需要虛擬主機組態來回應該主機，或是預設組態，才能順利進行應用程式的運作狀態檢查。運作狀態檢查請求具有下列屬性：User-Agent 設定為 ELB-HealthChecker/2.0，訊息標頭欄位的行結束字元是 CRLF 序列，標頭會在第一個空白行終止，後面接著 CRLF。

用戶端無法連接到面向網際網路的負載平衡器

如果負載平衡器未回應請求，則請檢查下列問題：

您的面向網際網路的負載平衡器已連接到私有子網路

您必須為負載平衡器指定公有子網路。公有子網路具有適用您虛擬私有雲端 (VPC) 對網際網路開道的路由。

安全群組或網路 ACL 不允許流量

負載平衡器的安全群組和負載平衡器子網路的任何網路 ACL，必須允許來自用戶端的傳入流量和連至接聽程式連接埠上用戶端的傳出流量。

負載平衡器不會收到傳送至自訂域的請求

如果負載平衡器未收到傳送至自訂域的請求，則請檢查下列問題：

自訂域名稱未解析為負載平衡器 IP 地址

- 使用命令列介面確認自訂域名稱解析的目標 IP 地址。
 - Linux、macOS 或 Unix – 您可以在終端內使用 `dig` 命令。例如 `dig example.com`
 - Windows – 您可以在命令提示內使用 `nslookup` 命令。例如 `nslookup example.com`
- 使用命令列介面確認負載平衡器 DNS 名稱解析的目標 IP 地址。
- 比較兩種輸出的結果。IP 地址必須相符。

如果使用 Route 53 託管自訂網域，請參閱《Amazon Route 53 開發人員指南》中的[我的網域在網際網路不可用](#)。

傳送至負載平衡器的 HTTPS 要求會傳回

"NET::ERR_CERT_COMMON_NAME_INVALID"

如果 HTTPS 請求從負載平衡器接收 `NET::ERR_CERT_COMMON_NAME_INVALID`，則請檢查下列可能的原因：

- HTTPS 請求中使用的域名稱與在關聯 ACM 憑證之接聽程式中指定的替代名稱不相符。
- 正在使用負載平衡器預設 DNS 名稱。預設 DNS 名稱無法用於提出 HTTPS 請求，因為無法針對 `*.amazonaws.com` 域請求公有憑證。

負載平衡器顯示處理時間延長

負載平衡器會根據組態以不同的方式計算處理時間。

- 如果 AWS WAF 與您的 Application Load Balancer 相關聯，且用戶端傳送 HTTP POST 要求，則傳送 POST 要求資料的時間會反映在負載平衡器存取記錄中的 `request_processing_time` 欄位中。HTTP POST 請求預期會發生這種行為。
- 如果 AWS WAF 與您的 Application Load Balancer 沒有關聯，而且用戶端傳送 HTTP POST 要求，則傳送 POST 要求資料的時間會反映在負載平衡器存取記錄中的 `target_processing_time` 欄位中。HTTP POST 請求預期會發生這種行為。

負載平衡器會傳送 000 的回應代碼

對於 HTTP/2 連線，如果任何標頭的壓縮長度超過 8 K 位元組，或透過一個連線提供的請求數量超過 10,000，則負載平衡器會傳送 GOAWAY 框架並關閉與 TCP FIN 的連線。

負載平衡器產生 HTTP 錯誤

以下 HTTP 錯誤是由負載平衡器產生。負載平衡器會將 HTTP 程式碼傳送到用戶端，將請求儲存到存取日誌，並遞增 HTTPCode_ELB_4XX_Count 或 HTTPCode_ELB_5XX_Count 指標。

錯誤

- [HTTP 400：錯誤的請求](#)
- [HTTP 401：未經授權](#)
- [HTTP 403：禁止](#)
- [HTTP 405：方法不允許](#)
- [HTTP 408：請求逾時](#)
- [HTTP 413：承載過大](#)
- [HTTP 414：URI 過長](#)
- [HTTP 460](#)
- [HTTP 463](#)
- [HTTP 464](#)
- [HTTP 500：內部伺服器錯誤](#)
- [HTTP 501：未導入](#)
- [HTTP 502：無效的閘道](#)
- [HTTP 503：服務無法使用](#)
- [HTTP 504：閘道逾時](#)
- [HTTP 505：不支援的版本](#)
- [儲存空間不足](#)
- [HTTP 561：未經授權](#)

HTTP 400：錯誤的請求

可能原因：

- 用戶端傳送不符合 HTTP 規格的格式錯誤請求。
- 請求標頭超過每個請求行 16 K、每個單一標頭 16 K，或整個請求標頭 64 K 的限制。
- 用戶端在傳送完整請求內文之前關閉了連線。

HTTP 401：未經授權

您設定了接聽程式規則來驗證使用者，但下列其中一項成立：

- 您已將 `OnUnauthenticatedRequest` 設定為拒絕未經身分驗證的使用者，或是 IdP 拒絕了存取。
- IdP 傳回的宣告大小超過負載平衡器支援的大小上限。
- 用戶端提交了不含主機標頭的 HTTP/1.0 請求，負載平衡器無法產生重新導向 URL。
- 請求的範圍不會傳回 ID 字符。
- 您沒有在用戶端登入逾時到期之前完成登入程序。如需詳細資訊，請參閱 [Client login timeout](#)。

HTTP 403：禁止

您已設定 AWS WAF Web 存取控制清單 (Web ACL) 來監視對 Application Load Balancer 的要求，並封鎖要求。

HTTP 405：方法不允許

用戶端使用了 TRACE 方法，該方法不受 Application Load Balancer 支援。

HTTP 408：請求逾時

用戶端在閒置逾時期間過期前不會傳送資料。傳送 TCP 持續作用無法防止此逾時。在每個閒置逾時期間經過前，先傳送至少 1 位元組的資料。視需要提高閒置逾時期間的長度。

HTTP 413：承載過大

可能原因：

- 目標是 Lambda 函數，請求內文超過 1 MB。
- 請求標頭超過每個請求行 16 K、每個單一標頭 16 K，或整個請求標頭 64 K 的限制。

HTTP 414 : URI 過長

請求 URL 或查詢字串參數太大。

HTTP 460

負載平衡器收到來自用戶端的請求，但用戶端在閒置逾時期間經過之前關閉了與負載平衡器的連線。

檢查用戶端逾時期間是否大於負載平衡器的閒置逾時期間。確保您的目標在用戶端逾時期間經過之前向用戶端提供回應，或如果用戶端支援的話，增加用戶端逾時期間以符合負載平衡器閒置逾時。

HTTP 463

負載平衡器收到具有過多 IP 地址的 X-Forwarded-For 請求標頭。IP 地址的數量上限為 30。

HTTP 464

負載平衡器收到的傳入請求通訊協定與目標群組通訊協定的版本組態不相容。

可能原因：

- 請求通訊協定是 HTTP/1.1，而目標群組通訊協定版本是 gRPC 或 HTTP/2。
- 請求通訊協定是 gRPC，而目標群組通訊協定版本是 HTTP/1.1。
- 請求通訊協定是 HTTP/2，請求不是 POST，而目標群組通訊協定版本是 gRPC。

HTTP 500 : 內部伺服器錯誤

可能原因：

- 您已設定 AWS WAF Web 存取控制清單 (Web ACL)，執行 Web ACL 規則時發生錯誤。
- 負載平衡器無法與 IdP 字符端點或 IdP 使用者資訊端點通訊。
 - 確認 IdP 的 DNS 是否可公開解析。
 - 驗證您的負載平衡器的安全群組和您的 VPC 的網路 ACL 允許對這些端點的傳出存取。
 - 驗證您的 VPC 具有網際網路存取。如果您有面對內部的負載平衡器，請使用 NAT 閘道來啟用網際網路存取。
- 從 IdP 收到的使用者宣告大小大於 11KB。

HTTP 501：未導入

負載平衡器收到的 Transfer-Encoding 標頭具有不支援的值。Transfer-Encoding 的支援值為 chunked 和 identity。或者，您可以使用 Content-Encoding 標頭。

HTTP 502：無效的閘道

可能原因：

- 在嘗試建立連線之前，負載平衡器從目標接收 TCP RST。
- 在嘗試建立連線之前，負載平衡器從目標收到意外的回應，例如「ICMP 目的地無法連線 (主機無法連線)」。檢查是否允許從負載平衡器子網路對目標連接埠上之目標的流量。
- 當負載平衡器有對目標的未完成請求時，目標關閉了具有 TCP RST 或 TCP FIN 的連線。檢查目標的持續作用持續期間是否短於負載平衡器的閒置逾時值。
- 目標回應的格式錯誤或包含無效的 HTTP 標頭。
- 目標回應標頭超過整個回應標頭 32 K 的限制。
- 由已取消註冊目標處理的請求取消註冊延遲期間已經過。增加延遲期間，使得耗時操作可以完成。
- 目標是 Lambda 函數，回應內文超過 1 MB。
- 目標是一個 Lambda 函數，它沒有在到達設定的逾時之前回應。
- 目標是傳回錯誤的 Lambda 函數，或是受 Lambda 服務限流的函數。
- 負載平衡器在連線至目標時遇到 SSL 交握錯誤。

如需詳細資訊，請參閱[如何疑難排解 Sup AWS port 知識中心中的 Application Load Balancer HTTP 502 錯誤](#)。

HTTP 503：服務無法使用

負載平衡器的目標群組沒有已註冊的目標。

HTTP 504：閘道逾時

可能原因：

- 負載平衡器無法在連線逾時過期 (10 秒) 之前建立對目標的連線。
- 負載平衡器建立了對目標的連線，但目標未在閒置逾時期間經過之前回應。
- 子網路的網路 ACL 不允許從目標到暫時性連接埠 (1024-65535) 上負載平衡器節點的流量。

- 目標傳回的內容長度標頭大於實體主體。負載平衡器等候遺失的位元組時逾時。
- 目標是 Lambda 函數，且 Lambda 服務未在連線逾時期間經過之前回應。
- 連線至目標時，負載平衡器遇到 SSL 交握逾時 (10 秒)。

HTTP 505：不支援的版本

負載平衡器收到非預期的 HTTP 版本請求。例如，負載平衡器建立了 HTTP/1 連線，但收到 HTTP/2 請求。

儲存空間不足

重新導向網址太長。

HTTP 561：未經授權

您設定了接聽程式規則來驗證使用者，但 IdP 在驗證使用者時傳回錯誤碼。檢查您的存取日誌，以獲取相關的[錯誤原因代碼](#)。

目標產生了 HTTP 錯誤

負載平衡器將來自目標的有效 HTTP 回應轉送至用戶端，包括 HTTP 錯誤。目標產生的 HTTP 錯誤會記錄在 HTTPCode_Target_4XX_Count 和 HTTPCode_Target_5XX_Count 指標中。

AWS Certificate Manager 憑證無法使用

決定在應用程式負載平衡器搭配使用 HTTPS 接聽程式時，AWS Certificate Manager 需要您在發行憑證之前驗證網域擁有權。如果在設定期間遺漏此步驟，則憑證會保持在 Pending Validation 狀態，且在驗證之前無法使用。

- 如果使用電子郵件驗證，請參閱《AWS Certificate Manager 使用者指南》中的[電子郵件驗證](#)。
- 如果使用 DNS 驗證，請參閱《AWS Certificate Manager 使用者指南》中的[DNS 驗證](#)。

不支援多行標頭

Application Load Balancer 不支援多行標頭，包括 message/http 媒體類型標頭。如果收到多行標頭，Application Load Balancer 會在將其傳遞至目標之前附加冒號字元 ":"。

使用資源對應疑難排解狀況不良的目標

如果您的 Application Load Balancer 目標未通過健康狀態檢查，您可以使用資源對應來尋找狀態不良的目標，並根據失敗原因程式碼採取動作。如需詳細資訊，請參閱 [Application Load Balancer 資源對應](#)。

資源對應提供兩種檢視：「概觀」和「狀況不良目標對映」。預設情況下會選取「概觀」，並顯示所有負載平衡器的資源。選取狀態不良的目標對映檢視將只會顯示與「Application Load Balancer」相關聯之每個目標群組中狀況不良的目標。

Note

您必須啟用 [顯示資源詳細資料]，才能檢視資源對映中所有適用資源的健全狀況檢查摘要和錯誤訊息。未啟用時，您必須選取每個資源以檢視其詳細資訊。

「目標群組」資料欄會顯示每個目標群組之狀況良好和狀況不良目標的摘要。這有助於判斷所有目標是否都未通過健康狀態檢查，或是只有特定目標失敗。如果目標群組中的所有目標都未通過健全狀況檢查，請檢查目標群組的組態。選取目標群組名稱，以在新索引標籤中開啟其詳細資訊頁面。

「目標」資料欄會顯示 TargetID 和每個目標的目前健康狀況檢查狀態。當目標狀況不良時，會顯示健全狀況檢查失敗原因代碼。當單一目標未通過健全狀況檢查時，請確認目標具有足夠的資源，並確認目標上執行的應用程式是否可用。選取目標 ID，以在新索引標籤中開啟其詳細資訊頁面。

選取 [匯出] 可讓您選擇將應用程式負載平衡器資源對映的目前檢視匯出為 PDF。

確認您的執行個體未通過運作狀態檢查，然後根據失敗原因程式碼檢查下列問題：

- 健康狀況不相符：HTTP 回應不符
 - 確認在目標上執行的應用程式是否正確傳送正確的 HTTP 回應至應用程式負載平衡器的健康狀態檢查要求。
 - 或者，您也可以更新應用程式負載平衡器的健全狀況檢查要求，以符合目標上執行之應用程式的回應。
- 健康狀況不良：請求逾時
 - 確認與目標相關聯的安全群組和網路存取控制清單 (ACL)，以及「Application Load Balancer」未封鎖連線。
 - 確認目標有足夠的資源可用來接受來自 Application Load Balancer 的連線。
 - 驗證目標上執行的任何應用程式的狀態。

- 您可以在每個目標的應用程式記錄中檢視應用程式負載平衡器的健全狀況檢查回應。如需詳細資訊，請參閱 [Health 檢查原因代碼](#)。
- 不健康：FailedHealthChecks
 - 驗證目標上執行的任何應用程式的狀態。
 - 確認目標正在接聽健全狀況檢查連接埠上的流量。

使用 HTTPS 接聽程式時

您可以選擇用於前端連線的安全性原則。系統會根據使用中的前端安全性原則，自動選取用於後端連線的安全性原則。

- 如果您的 HTTPS 接聽程式使用 TLS 1.3 安全性原則進行前端連線，則ELBSecurityPolicy-TLS13-1-0-2021-06安全性原則會用於後端連線。
 - 如果您的 HTTPS 接聽程式未針對前端連線使用 TLS 1.3 安全性原則，則ELBSecurityPolicy-2016-08安全性原則會用於後端連線。
- 如需詳細資訊，請參閱[安全性原則](#)。

- 確認目標是否以安全性原則指定的正確格式提供伺服器憑證和金鑰。
- 確認目標支援一或多個相符的密碼，以及 Application Load Balancer 提供用於建立 TLS 交握的通訊協定。

Application Load Balancer 的配額

對於每個 AWS 服務，您的 AWS 帳戶有預設配額，先前稱為限制。除非另有說明，否則每個配額都是區域特定規定。您可以請求提高某些配額，而其他配額無法提高。

若要檢視 Application Load Balancer 的配額，請開啟 [Service Quotas 主控台](#)。在導覽窗格中，選擇 AWS services (AWS 服務)，然後選取 Elastic Load Balancing。您也可以使用 [describe-account-limits](#)(AWS CLI) 命令進行 Elastic Load Balancing。

若要請求增加配額，請參閱 Service Quotas 使用者指南中的 [請求提高配額](#)。如果 Service Quotas 中尚未提供配額，請使用 [Elastic Load Balancing 限制增加表單](#)。

負載平衡器

AWS 帳戶具有下列與 Application Load Balancer 相關的配額。

名稱	預設	可調整
每個區域的 Application Load Balancer	50	是
每個 Application Load Balancer 的憑證 (不含預設憑證)	25	是
每個 Application Load Balancer 的接聽程式	50	是
每個 Application Load Balancer 每個動作的目標群組	5	否
每個 Application Load Balancer 的目標群組	100	否
每個 Application Load Balancer 的目標	1,000	是

目標群組

下列配額適用於目標群組。

名稱	預設	可調整
每個區域的目標群組	3,000 *	是

名稱	預設	可調整
每個區域每個目標群組的目標數 (執行個體或 IP 地址)	1,000	是
每個區域每個目標群組的目標數 (Lambda 函數)	1	否
每個目標群組的負載平衡器	1	否

* 此配額由 Application Load Balancer 和 Network Load Balancer 共用。

規則

下列配額適用於規則。

名稱	預設	可調整
每個 Application Load Balancer 的規則 (不含預設規則)	100	是
每個規則的條件值	5	否
每個規則的條件萬用字元	5	否
每個規則的比對評估次數	5	否

信託商店

下列配額適用於信任存放區。

名稱	預設	可調整
每個帳戶的信任商店	20	是
每個負載平衡器在驗證模式下使用 MTL 的接聽程式數目。	2	否

憑證機構憑證

下列是 CA 憑證的配額。

名稱	預設	可調整
每個信任存放區的 CA 憑證	25	是
CA 憑證大小	16 KB	否
最大憑證鏈結深度	4	否

憑證撤銷清單

以下是憑證撤銷清單的配額。

名稱	預設	可調整
每個信任存放區的撤銷清單	30	是
每個信任存放區的撤銷項目	500,000	是
撤銷清單檔案大小	50 毫升	否

HTTP 標頭

HTTP 標頭的大小限制如下。

名稱	預設	可調整
請求行	16 K	否
單一標頭	16 K	否
整個回應標頭	32 K	否
整個請求標頭	64 K	否

Application Load Balancer 的文件歷史記錄

下表說明 Application Load Balancer 各版本。

變更	描述	日期
資源圖	此版本新增了以視覺化格式檢視負載平衡器資源和關係的支援。	2024年3月8日
一鍵式 WAF	此版本新增了對設定負載平衡器行為的支援 (如果與按一下滑鼠整合) AWS WAF。	2024年2月6日
相互 TLS	此版本新增了對相互 TLS 驗證的支援。	2023 年 11 月 26 日
自動目標權重	此版本新增了對自動目標權重演算法的支援。	2023 年 11 月 26 日
TLS 終端機	此版本新增了在終止 TLS 連線時使用 FIPS 140-3 流量圖形模組的安全性原則。	2023 年 11 月 20 日
使用 IPv6 註冊目標	此版本新增了在 IPv6 處理時將執行個體註冊為目標的支援。	2023 年 10 月 2 日
支援 TLS 1.3 的安全性原則	此版本新增對 TLS 1.3 預先定義安全性原則的支援。	2023 年 3 月 22 日
區域移位	此版本新增支援，透過與 Amazon Route 53 Application Recovery Controller.	2022 年 11 月 28 日
關閉跨區域負載平衡	此版本增加了關閉跨區域負載平衡的支援。	2022 年 11 月 28 日
目標群組運作狀態	此版本新增的支援，可讓您設定必須處於運作狀態良好之目	2022 年 11 月 28 日

	標的最小計數或百分比，以及不符合閾值時負載平衡器採取的動作。	
跨區域負載平衡	此版本新增了目標群組層級設定跨區域負載平衡的支援。	2022 年 11 月 17 日
IPv6 目標群組	此版本新增支援，可讓您為 Application Load Balancer 設定 IPv6 目標群組。	2021 年 11 月 23 日
IPv6 內部負載平衡器	此版本新增支援，可讓您為 Application Load Balancer 設定 IPv6 目標群組。	2021 年 11 月 23 日
AWS PrivateLink 和靜態 IP 位址	此版本透過將流量直接從網路負載平衡器轉送至應用程式負載平衡器，以增加對使用 AWS PrivateLink 和公開靜態 IP 位址的支援。	2021 年 9 月 27 日
用戶端連接埠保留	此版本新增屬性，可保留用戶端用來連線到負載平衡器的來源連接埠。	2021 年 7 月 29 日
TLS 標頭	此版本新增了一個屬性，指出 TLS 標頭 (其中包含有關交涉的 TLS 版本和加密套件的資訊) 已新增至用戶端要求，然後再將其傳送至目標。	2021 年 7 月 21 日
其他 ACM 憑證	此版本支援具有 2048、3072 和 4096 位元金鑰長度的 RSA 憑證，以及所有 ECDSA 憑證。	2021 年 7 月 14 日
應用程式型粘性	此版本新增應用程式型 Cookie，以支援負載平衡器的粘性會話。	2021 年 2 月 8 日

支援 TLS 1.2 版之 FS 的安全政策	此版本新增支援 TLS 1.2 版向前保密 (FS) 的安全政策。	2020 年 11 月 24 日
WAF 故障開啟支援	此版本新增了對設定負載平衡器行為 (如果與之整合) 的支援 AWS WAF。	2020 年 11 月 13 日
gRPC 和 HTTP/2 支援	此版本新增了對 gRPC 工作負載和 end-to-end HTTP/2 的支援。	2020 年 10 月 29 日
Outpost 支援	您可 Application Load Balancer 在 AWS Outposts.	2023 年 9 月 8 日
去同步緩解模式	此版本新增對非同步緩和模式的支援。	2020 年 8 月 17 日
最少未完成的請求	此版本增加對最少未完成請求演算法的支援。	2019 年 11 月 25 日
加權目標群組	此版本增加對多個目標群組轉送動作的支援。請求會根據您為每個目標群組指定的權重分配至這些目標群組。	2019 年 11 月 19 日
New attribute (新建屬性)	此版本新增對路由 http.drop_invalid_header_fields.enabled 屬性的支援。	2019 年 11 月 15 日
FS 的安全性原則	此版本新增了對三個額外預先定義的正向保密安全性原則的支援。	2019 年 10 月 8 日
進階請求路由	此版本對接聽程式規則增加支援其他條件類型。	2019 年 3 月 27 日
Lambda 函數作為目標	此版本增加將 IP 函數註冊為目標的支援。	2018 年 11 月 29 日

重新導向動作	此版本增加對負載平衡器將請求重新導向不同 URL 的支援。	2018 年 7 月 25 日
固定回應動作	此版本增加對負載平衡器傳回自訂 HTTP 回應的支援。	2018 年 7 月 25 日
FS 和 TLS 1.2 的安全政策	此版本增加對額外兩個預先定義安全政策的支援。	2018 年 6 月 6 日
使用者身分驗證	此版本增加對負載平衡器在路由傳送請求之前，使用企業或社交身分來驗證應用程式使用者的支援。	2018 年 5 月 30 日
資源層級許可	此版本增加對資源層級許可和標記條件金鑰的支援。	2018 年 5 月 10 日
慢啟動模式	此版本增加對慢速啟動模式的支援，該模式可在負載平衡器預備時，逐漸增加負載平衡器傳送到新註冊目標的請求量。	2018 年 3 月 24 日
SNI 支援	此版本增加對伺服器名稱指示 (SNI) 的支援。	2017 年 10 月 10 日
IP 地址即目標	此版本新增了支援註冊 IP 地址做為目標。	2017 年 8 月 31 日
以主機為基礎的路由	此版本增加根據主機標頭中主機名稱來路由請求的支援。	2017 年 4 月 5 日
TLS 1.1 及 TLS 1.2 的安全性原則	此版本增加 TLS 1.1 和 TLS 1.2 的安全政策。	2017 年 2 月 6 日
IPv6 支援	此版本增加對 IPv6 地址的支援。	2017 年 1 月 25 日
請求追蹤	此版本增加對請求追蹤的支援。	2016 年 11 月 22 日

[量度的百分位數支援
TargetResponseTime](#)

此版本增加了對 Amazon 支援的新百分位數統計資料的支援。 CloudWatch

2016 年 11 月 17 日

[新的負載平衡器類型](#)

此 Elastic Load Balancing 版本推出 Application Load Balancer。

2016 年 8 月 11 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。