AWS IoT

開發人員指南

AWS IoT: 開發人員指南

Copyright © 2019 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

什麼是 AWS IoT？ ... 1
- AWS IoT 元件 ... 1
- 如何開始使用 AWS IoT ... 2
- 存取 AWS IoT ... 2
- 相關服務 ... 3
- AWS IoT 運作方式 ... 3

AWS IoT 入門 ... 4
- 登入 AWS IoT 主控台。 ... 4
- 在登錄檔中登錄裝置 ... 5
  - 建立或啟用裝置憑證 ... 14
  - 建立 AWS IoT 政策 ... 17
  - 將 AWS IoT 政策連接至裝置憑證 ... 20
  - 將憑證連接到物件 ... 22
- 設定您的裝置 ... 25
- 以 AWS IoT MQTT 用戶端查看裝置的 MQTT 訊息。 ... 25
- 規則的設定與測試 ... 27
  - 建立 SNS 主題。 ... 27
  - 訂閱 Amazon SNS 主題 ... 30
  - 建立規則 ... 32
  - 測試 Amazon SNS 規則 ... 38
  - 後續步驟 ... 38
- 建立和追蹤的 AWS IoT 任務 ... 38
  - 連接至您的裝置至 AWS IoT。 ... 38
  - 執行任務範例 ... 39
  - 建立任務文件 ... 39
  - 建立任務 ... 39
  - 在裝置上執行任務 ... 47
  - 追蹤任務進度和任務執行活動 ... 48

AWS IoT 規則教學 ... 52
- 使用 DynamoDB 動作建立規則 ... 52
  - 使用 DynamoDB 動作測試規則 ... 62
- 使用 AWS Lambda 動作建立規則 ... 63
  - 建立 Lambda 函數 ... 63
  - 測試您的 Lambda 函式 ... 67
  - 使用 Lambda 動作建立規則 ... 70
  - 使用 Lambda 動作測試您的規則 ... 80
  - 使用 Lambda 動作疑難排解規則 ... 81
- 建立 Amazon SNS 規則 ... 82

在 Raspberry Pi 上使用 AWS IoT SDK ... 91
- 先決條件 ... 91
- 建立 Raspberry Pi 的 AWS IoT 實物 ... 91
- SDK 範例組態 ... 92
- 使用適用於 Embedded C 的 AWS IoT SDK ... 92
  - 安裝適用於 Embedded C 的 AWS IoT 裝置 SDK ... 92
  - 範例應用程式組態 ... 92
  - 執行範例應用程式 ... 93
- 使用適用於 JavaScript 的 AWS IoT 裝置 SDK 和節點 ... 94
  - 安裝最新版本的 Node.js ... 94
  - 安裝適用於 JavaScript 的 AWS IoT 裝置 SDK ... 94
  - 準備執行範例應用程式 ... 95
  - 執行範例應用程式 ... 96

AWS IoT 其他教學課程 ... 97
- 使用 AWS IoT 和 Raspberry Pi 監控土壤濕度 ... 97
  - 先決條件 ... 97

設定 AWS IoT 97
設定您的 Raspberry Pi 和濕度感應器 101
利用 AWS IoT 管理裝置 106
如何使用登錄檔管理物件 106
建立物件 106
列出物件 107
搜尋物件 107
更新物件 108
刪除物件 109
將主體連接至物件 109
將主體與物件分離 109
物件類型 109
建立物件類型 110
列出物件類型 110
描述物件類型 110
將物件類型與物件建立關聯 111
棄用物件類型 111
刪除物件類型 112
物件群組 112
建立物件群組 113
描述物件群組 114
將物件加入物件群組 114
從物件群組移除物件 115
列出物件群組中的物件 115
列出物件群組 115
列出物件的群組 117
更新物件群組 117
刪除物件群組 117
將政策連接至物件群組 118
從物件群組分離政策 118
列出連接至物件群組的政策 118
列出政策所在的群組 119
為物件取得有效政策 119
MQTT 動作的測試授權 120
動態物件群組 121
建立動態物件群組 121
描述動態物件群組 122
更新動態物件群組 123
刪除動態物件群組 123
限制和衝突 123
為您的 AWS IoT 資源建立標籤 125
標籤基本概念 125
標籤的限制與上限 126
搭配 IAM 政策使用標籤 126
帳單群組 127
檢視成本分配和用量資料 127
限制 128
安全性 129
AWS IoT 中的安全性 129
身分驗證 130
X.509 憑證概觀 130
伺服器驗證 130
用戶端身分驗證 132
自訂身分驗證 138
管理裝置憑證 145
伺服器驗證 130
授權 145

AWS IoT 政策 ... 147
授權直接呼叫 AWS 服務 ... 172
跨帳戶存取 ... 175
資料保護 ... 177
AWS IoT 中的傳輸安全性 ... 178
資料加密 ... 178
Identity and Access Management ... 179
對象 ... 179
使用 IAM 身分來驗證 ... 179
使用政策管理存取權 ... 181
AWS IoT 如何搭配 IAM 運作 ... 183
身分類型政策範例 ... 196
故障診斷 ... 198
記錄日誌和監控 ... 200
監控工具 ... 201
使用 Amazon CloudWatch 進行監控 ... 202
使用 CloudWatch Logs 進行監控 ... 210
使用 AWS CloudTrail 記錄 AWS IoT API 呼叫 ... 230
合規驗證 ... 232
彈性 ... 232
基礎設施安全 ... 233
漏洞分析 ... 233
安全最佳實務 ... 233
在 AWS IoT 中保護 MQTT 連線 ... 233
讓裝置的時鐘保持同步 ... 235
驗證伺服器憑證 ... 235
使用每個裝置單一身分 ... 236
使用及時佈建 ... 236
連接裝置 ... 237
可設定端點 (Beta 版) ... 237
建立和設定 AWS 受管網域 ... 238
建立和設定自訂網域 ... 238
管理網域組態 ... 240
訊息中介裝置 ... 242
通訊協定 ... 242
通訊協定、連接埠映射和身分驗證 ... 242
MQTT ... 242
HTTP ... 252
規則 ... 254
授與 AWS IoT 所需的存取權 ... 254
傳送角色許可 ... 256
建立 AWS IoT 規則 ... 256
檢視您的規則 ... 260
刪除規則 ... 260
AWS IoT規則動作 ... 260
CloudWatch 警示動作 ... 261
CloudWatch 指標動作 ... 262
DynamoDB 動作 ... 263
DynamoDBv2 動作 ... 264
Elasticsearch 動作 ... 265
Firehose 動作 ... 266
HTTP 動作 ... 266
IoT 分析動作 ... 268
IoT 事件動作 ... 269
IoT SiteWise 動作 ... 270
Kinesis 動作 ... 273
Lambda 動作 ... 273

Republish 動作 ... 275
S3 動作 ... 275
Salesforce 動作 ... 276
SNS 動作 ... 277
SQS 動作 ... 278
Step Functions 動作 ... 279
規則的故障診斷 ... 279
錯誤處理 (錯誤動作) ... 280
錯誤動作訊息格式 ... 280
錯誤動作範例 ... 281
使用主題規則目的地 ... 281
建立主題規則目的地 ... 282
確認主題規則目的地 ... 282
停用主題規則目的地 ... 283
啟用主題規則目的地 ... 283
傳送新的確認訊息 ... 283
刪除主題規則目的地 ... 283
AWS IoTSQL 參考 ... 283
資料類型 ... 284
運算子 ... 287
函數 ... 292
SELECT 子句 ... 323
FROM 子句 ... 325
WHERE 子句 ... 325
文字 ... 326
案例陳述式 ... 326
JSON Extensions ... 327
替代範本 ... 327
SQL 版本 ... 328
2016 年 3 月 23 日 SQL 規則引擎版本的新功能 ... 329
基本擷取 ... 331
若要使用基本擷取 ... 331
Device Shadow 服務 ... 332
Device Shadow 服務資料流程 ... 332
偵測物件是否連網 ... 338
Device Shadow 服務文件 ... 339
文件屬性 ... 339
Device Shadow 的版本控制 ... 340
用戶端符記 ... 340
範例文件 ... 340
空白部分 ... 341
陣列 ... 341
使用影子 ... 342
通訊協定支援 ... 342
更新影子 ... 343
擷取影子文件 ... 343
刪除資料 ... 346
刪除影子 ... 347
差量狀態 ... 347
觀察狀態變更 ... 349
訊息順序 ... 349
裁剪影子訊息 ... 350
RESTful API ... 350
GetThingShadow ... 351
UpdateThingShadow ... 352
DeleteThingShadow ... 352
MQTT Pub/Sub Topics ... 353

/update ... 353
/update/accepted ... 354
/update/documents ... 355
/update/rejected ... 355
/update/delta ... 356
/get ... 356
/get/accepted ... 357
/get/rejected ... 357
/delete ... 358
/delete/accepted ... 358
/delete/rejected ... 359
文件語法 ... 359
請求狀態文件 ... 360
回應狀態文件 ... 360
錯誤回應文件 ... 361
錯誤訊息 ... 362
任務 ... 363
Jobs 的重要概念 ... 363
管理工作 ... 365
建立和管理任務 (主控台) ... 366
建立和管理任務 (CLI) ... 367
裝置和任務 ... 374
設定裝置與 Jobs 合作 ... 376
使用 AWS IoT 任務 API ... 385
工作管理和控制 API ... 385
Jobs 裝置 MQTT 和 HTTPS API ... 436
任務推展和中止組態 ... 458
使用任務推展速率 ... 458
使用任務推展中止組態 ... 459
工作限制 ... 460
AWS IoT 安全通道 ... 461
安全通道概念 ... 461
安全通道教學課程 ... 461
先決條件 ... 461
開啟通道 ... 462
啟動本機代理 ... 462
啟動 SSH 工作階段 ... 462
關閉通道 ... 463
安全通道生命週期 ... 463
控制 Kibana 的存取 ... 463
通道存取先決條件 ... 463
iot:OpenTunnel ... 463
iot:DescribeTunnel ... 465
iot:ListTunnels ... 465
iot:ListTagsForResource ... 465
iot:CloseTunnel ... 465
iot:TagResource ... 466
iot:UntagResource ... 466
本機代理 ... 466
本機代理安全最佳實務 ... 467
IoT Agent Snippet ... 467
設定遠端裝置 ... 468
裝置佈建 ... 470
使用機群佈建來佈建沒有裝置憑證的裝置 ... 470
透過要求佈建 ... 470
由信任的使用者佈建 ... 471
機群佈建 API ... 472

佈建具有裝置憑證的裝置 475
單一物件佈建 475
即時佈建 475
大量註冊 478
佈建範本 478
參數部分 479
資源部分 479
JITP 和大量註冊的範本範例 482
機群佈建 484
機群索引服務 487
管理實物索引 487
啟用實物索引 487
描述實物索引 492
查詢實物索引 493
法規與限制 494
授權 496
管理實物群組索引 496
啟用實物群組索引 496
描述群組索引 497
查詢實物群組索引 497
授權 497
彙總資料的查詢 498
GetStatistics 498
GetCardinality 499
GetPercentiles 500
疑難排解彙總查詢 501
查詢語法 502
範例實物查詢 503
範例實物群組查詢 504
AWS IoT Device Defender 506
稽核 506
問題嚴重性 506
稽核檢查 507
如何執行稽核 528
通知 528
許可 531
服務限制 535
稽核命令 535
管理稽核設定 535
排定稽核時程 539
執行隨需稽核 547
管理稽核執行個體 548
檢查稽核結果 554
緩解動作 559
如何定義並管理緩解動作 562
套用緩解動作 565
許可 570
Service Limits 573
緩解行動命令 574
CreateMitigationAction 574
UpdateMitigationAction 578
ListMitigationActions 581
DescribeMitigationAction 582
DeleteMitigationAction 586
StartAuditMitigationActionsTask 586
CancelAuditMitigationActionsTask 589
ListAuditMitigationActionsExecutions 589

ListAuditMitigationActionsTasks ... 592
DescribeAuditMitigationActionsTask ... 595
偵測 ... 598
概念 ... 599
行為 ... 600
指標 ... 601
監控未註冊裝置的行為 ... 611
如何使用 AWS IoT Device Defender Detect ... 612
許可 ... 613
服務限制 ... 614
從裝置傳送指標 ... 614
偵測命令 ... 619
AttachSecurityProfile ... 619
CreateSecurityProfile ... 620
DeleteSecurityProfile ... 624
DescribeSecurityProfile ... 625
DetachSecurityProfile ... 628
ListActiveViolations ... 629
ListSecurityProfiles ... 633
ListSecurityProfilesForTarget ... 635
ListTargetsForSecurityProfile ... 636
ListViolationEvents ... 638
UpdateSecurityProfile ... 642
ValidateSecurityProfileBehaviors ... 648
裝置代理程式與 AWS IoT Greengrass 整合 ... 651
裝置代理程式的安全性最佳實務 ... 653
AWS IoT Device Defender 故障診斷指南 ... 654
事件訊息 ... 657
登錄檔事件 ... 658
Jobs 事件 ... 664
生命週期事件 ... 667
連線/中斷連線事件 ... 667
Subscribe/Unsubscribe Events ... 668
適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 ... 670
在 NXP 裝置上開始使用 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 ... 671
使用預先設定的 NXP 帳戶預覽 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 ... 672
使用您的 AWS 和 Alexa Voice Service 開發人員帳戶來設定適用於 AWS IoT 的 AVS 整合 ... 675
AWS IoT 軟體開發套件 ... 678
適用於 Android 的 AWS Mobile SDK ... 678
Arduino Yún SDK ... 678
適用於 Embedded C 的 AWS IoT 裝置 SDK ... 678
AWS IoT C++ 裝置 SDK ... 679
適用於 iOS 的 AWS Mobile SDK ... 679
適用於 Java 的 AWS IoT 裝置 SDK ... 679
適用於 JavaScript 的 AWS IoT 裝置 SDK ... 679
適用於 Python 的 AWS IoT 裝置 SDK ... 679
故障診斷 ... 681
診斷連線問題 ... 681
身分驗證 ... 681
授權 ... 681
診斷規則問題 ... 681
診斷影子的問題 ... 682
診斷 Salesforce 動作問題 ... 683
執行追蹤 ... 683
作業成功和故障 ... 683
AWS IoT 錯誤 ... 684
AWS IoT 限制 ... 686

物件限制 686
物件群組限制 686
訊息中介限制 686
通訊協定限制 689
裝置影子限制 690
安全性和身分限制 691
AWS IoT 調節限制 691
AWS IoT 規則引擎限制 695
AWS IoT 規則引擎 HTTP 動作限制 695
AWS IoT 工作限制 460
AWS IoT 安全通道限制 697
AWS IoT 裝置機群索引限制 697
AWS IoT 調節限制 698
AWS IoT 機群佈建限制 698
AWS IoT 大量實物註冊限制 698
AWS IoT Device Defender 限制 699
IOT 命令 700
AcceptCertificateTransfer 704
AddThingToBillingGroup 705
AddThingToThingGroup 706
AssociateTargetsWithJob 708
AttachPolicy 709
AttachPrincipalPolicy 710
AttachSecurityProfile 712
AttachThingPrincipal 713
CancelAuditMitigationActionsTask 714
CancelAuditTask 714
CancelCertificateTransfer 715
CancelJob 716
CancelJobExecution 718
ClearDefaultAuthorizer 720
CreateAuthorizer 721
CreateBillingGroup 722
CreateCertificateFromCsr 724
CreateDynamicThingGroup 726
CreateJob 729
CreateKeysAndCertificate 734
CreateMitigationAction 735
CreateOTAUpdate 738
CreatePolicy 743
CreatePolicyVersion 745
CreateRoleAlias 746
CreateScheduledAudit 748
CreateSecurityProfile 750
CreateStream 753
CreateThing 756
CreateThingGroup 758
CreateThingType 761
CreateTopicRule 763
DeleteAccountAuditConfiguration 776
DeleteAuthorizer 777
DeleteBillingGroup 778
DeleteCACertificate 779
DeleteCertificate 780
DeleteDynamicThingGroup 781
DeleteJob 782
DeleteJobExecution 783

DeleteMitigationAction 785
DeleteOTAUpdate 786
DeletePolicy 787
DeletePolicyVersion 788
DeleteRegistrationCode 789
DeleteRoleAlias 790
DeleteScheduledAudit 791
DeleteSecurityProfile 792
DeleteStream 793
DeleteThing 794
DeleteThingGroup 795
DeleteThingShadow 796
DeleteThingType 797
DeleteTopicRule 798
DeleteV2LoggingLevel 799
DeprecateThingType 800
DescribeAccountAuditConfiguration 801
DescribeAuditFinding 802
DescribeAuditMitigationActionsTask 806
DescribeAuditTask 810
DescribeAuthorizer 813
DescribeBillingGroup 815
DescribeCACertificate 816
DescribeCertificate 819
DescribeDefaultAuthorizer 821
DescribeEndpoint 823
DescribeEventConfigurations 824
DescribeIndex 825
DescribeJob 826
DescribeJobExecution 831
DescribeJobExecution 834
DescribeMitigationAction 836
DescribeRoleAlias 839
DescribeScheduledAudit 841
DescribeSecurityProfile 843
DescribeStream 846
DescribeThing 848
DescribeThingGroup 850
DescribeThingRegistrationTask 853
DescribeThingType 855
DetachPolicy 857
DetachPrincipalPolicy 858
DetachSecurityProfile 859
DetachThingPrincipal 860
DisableTopicRule 861
EnableTopicRule 862
GetEffectivePolicies 863
GetIndexingConfiguration 864
GetJobDocument 866
GetLoggingOptions 867
GetOTAUpdate 868
GetPendingJobExecutions 873
GetPolicy 875
GetPolicyVersion 876
GetRegistrationCode 878
GetStatistics 879
GetThingShadow 881

GetTopicRule 882
GetV2LoggingOptions 895
ListActiveViolations 896
ListAttachedPolicies 900
ListAuditFindings 902
ListAuditMitigationActionsExecutions 907
ListAuditMitigationActionsTasks 909
ListAuditTasks 911
ListAuthorizers 913
ListBillingGroups 915
ListCACertificates 917
ListCertificates 918
ListCertificatesByCA 920
ListIndices 922
ListJobExecutionsForJob 923
ListJobExecutionsForThing 925
ListJobs 928
ListMitigationActions 930
ListOTAUpdates 932
ListOutgoingCertificates 933
ListPolicies 935
ListPolicyPrincipals 937
ListPolicyVersions 938
ListPrincipalPolicies 940
ListPrincipalThings 941
ListRoleAliases 943
ListScheduledAudits 944
ListSecurityProfiles 946
ListSecurityProfilesForTarget 947
ListStreams 949
ListTagsForResource 951
ListTargetsForPolicy 952
ListTargetsForSecurityProfile 953
ListThingGroups 955
ListThingGroupsForThing 956
ListThingPrincipals 958
ListThingRegistrationTaskReports 959
ListThingRegistrationTasks 961
ListThingTypes 962
ListThings 964
ListThingsInBillingGroup 966
ListThingsInThingGroup 968
ListTopicRules 969
ListV2LoggingLevels 970
ListViolationEvents 972
發佈 976
RegisterCACertificate 977
RegisterCertificate 979
RegisterThing 981
RejectCertificateTransfer 982
RemoveThingFromBillingGroup 983
RemoveThingFromThingGroup 984
ReplaceTopicRule 985
SearchIndex 998
SetDefaultAuthorizer 1001
SetDefaultPolicyVersion 1003
SetLoggingOptions 1004

SetV2LoggingLevel 1005
SetV2LoggingOptions 1006
StartAuditMitigationActionsTask 1007
StartNextPendingJobExecution 1009
StartOnDemandAuditTask 1011
StartThingRegistrationTask 1013
StopThingRegistrationTask 1014
TagResource 1015
TestAuthorization 1016
TestInvokeAuthorizer 1020
TransferCertificate 1022
UntagResource 1023
UpdateAccountAuditConfiguration 1024
UpdateAuthorizer 1026
UpdateBillingGroup 1027
UpdateCACertificate 1029
UpdateCertificate 1030
UpdateDynamicThingGroup 1032
UpdateEventConfigurations 1034
UpdateIndexingConfiguration 1035
UpdateJob 1036
UpdateJobExecution 1039
UpdateMitigationAction 1042
UpdateRoleAlias 1045
UpdateScheduledAudit 1046
UpdateSecurityProfile 1048
UpdateStream 1055
UpdateThing 1057
UpdateThingGroup 1059
UpdateThingGroupsForThing 1061
UpdateThingShadow 1062
ValidateSecurityProfileBehaviors 1063

什麼是 AWS IoT？

AWS IoT 可在已與網際網路連線的裝置 (例如感測器、傳動器、嵌入式微控制器或智慧型設備) 和 AWS 雲端之間，提供安全的雙向通訊。您如此可從多項裝置收集遙測資料，及存放和分析資料。您也可以建立應用程式，讓您的使用者從手機或平板電腦控制這些裝置。

AWS IoT 元件

AWS IoT 包含以下元件：

裝置閘道

讓裝置以安全有效的方式與 AWS IoT 通訊。

訊息中介裝置

提供安全的機制，讓裝置和 AWS IoT 應用程式能夠互相發佈與接收訊息。您可以直接使用 MQTT 通訊協定或是經 WebSocket 的 MQTT 來發佈和訂閱。您可以使用 HTTP REST 介面發佈。如需詳細資訊，請參閱 AWS IoT 的訊息中介裝置 (p. 242)。

規則引擎

可處理訊息並與其他 AWS 服務整合。您可使用以 SQL 為基礎的語言從訊息承載選擇資料，然後處理資料並傳送至其他服務，例如 Amazon S3、Amazon DynamoDB 和 AWS Lambda。您也可以使用訊息中介裝置重新發佈訊息給其他訂閱者。如需詳細資訊，請參閱 AWS IoT 的規則 (p. 254)。

安全與身分服務

在 AWS 雲端對於安全提供共同的責任。您的裝置必須保護其登入資料，才可將資料安全地傳送至訊息中介裝置。訊息中介裝置和規則引擎使用 AWS 安全功能，將資料安全傳送至裝置或其他 AWS 服務。如需更多詳細資訊，請參閱 身分驗證 (p. 130)。

登錄

整理 AWS 雲端中與每項裝置相關聯的資源。您可登錄您的裝置，且每項裝置最多可與三個自訂屬性相關聯。您也可以建立憑證和 MQTT 用戶端 ID 與每項裝置的關聯，提升您管理裝置及排解其問題的能力。如需更多詳細資訊，請參閱 利用 AWS IoT 管理裝置 (p. 106)。

群組登錄檔

您可利用群組，將多個裝置分類至群組來同時管理。群組還可以包含群組—，這樣就能建立群組階層。您對父群組執行的任何動作，都會套用至其子群組，以及父群組和其所有子群組中的所有裝置。為群組授予許可，也會套用至該群組及其所有子群組中的所有裝置。如需詳細資訊，請參閱 利用 AWS IoT 管理裝置 (p. 106)。

裝置影子

用於存放及擷取裝置目前狀態資訊的 JSON 文件。

Device Shadow 服務

提供您裝置在 AWS 雲端中不變的表示方式。您可以對裝置影子發佈更新的狀態資訊，而您的裝置會在連線時同步其狀態。您的裝置也可以對影子發佈其目前的狀態，供應用程式或其他裝置使用。如需更多詳細資訊，請參閱 AWS IoT 適用的 Device Shadow 服務 (p. 332)。

裝置佈建服務

可讓您使用描述裝置所需資源 (「實物」、憑證、一或多項政策) 的範本，來佈建裝置。物件即登錄檔中的項目，內含描述裝置的屬性。裝置使用憑證向 AWS IoT 進行驗證。政策決定裝置在 AWS IoT 可執行的操作。

範本包含了被字典 (對應) 之值取代的變數。只要在字典中為範本的變數傳入不同的值，您就可以使用相同的範本來佈建多個裝置。如需更多詳細資訊，請參閱 裝置佈建 (p. 470)。

自訂身分驗證服務

您可以定義自訂授權方，使用自訂身分驗證服務和 Lambda 函數，藉以管理自己的身分驗證和授權策略。自訂授權方可讓 AWS IoT 使用承載符記與身分驗證策略來驗證您的裝置並授權操作。

自訂授權方可以實作多種身分驗證策略 (例如：JSON Web Token 驗證、OAuth 提供者呼叫等)，也必須傳回由裝置閘道用於授權 MQTT 操作的政策文件。如需更多詳細資訊，請參閱 自訂身分驗證 (p. 138)。

任務服務

可讓您定義一組遠端操作，這組操作會傳送到連接 AWS IoT 的一個或多個裝置並且執行。例如，您可以定義一個工作，指示一組裝置下載並安裝應用程式或韌體更新、重新啟動、輪換憑證，或者執行遠端故障排除操作。

欲建立工作，您必須指定要執行的遠端操作之描述，以及應執行操作的目標清單。目標可以是個別裝置、群組或兩者。如需更多詳細資訊，請參閱 任務 (p. 363)。

適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合

將 Alexa Voice 帶至任何連線裝置。適用於 AWS IoT 的 AVS 可降低整合 Alexa 的成本和複雜性。此功能運用 AWS IoT 將裝置中的密集運算和記憶體音訊任務卸載至雲端。由於工程用料表 (eBoM) 的成本降低，裝置製造商可以經濟實惠的方式將 Alexa 帶到資源有限的 IoT 裝置，讓消費者可以局部在其住家、辦公室或旅館房間直接與 Alexa 通話，以獲得環境體驗。

適用於 AWS IoT 的 AVS 在 MCU 上啟用 Alexa 內建功能，例如具有少於 1 MB 內嵌 RAM 的 ARM Cortex M 類別。若要這樣做，AVS 會將記憶體和運算任務卸載至雲端中的虛擬 Alexa 內建裝置。如此可以降低高達 50% 的 eBoM 成本。如需詳細資訊，請參閱 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 (p. 670)。

如需 AWS IoT 限制的詳細資訊，請參閱 AWS IoT。

如何開始使用 AWS IoT

- 要進一步了解 AWS IoT，請參閱 AWS IoT 運作方式 (p. 3)。
- 如需了解如何將裝置連線至 AWS IoT，請參閱「AWS IoT 入門 (p. 4)」。

存取 AWS IoT

AWS IoT 提供下列界面供您建立裝置並與其互動：

- AWS Command Line InterfaceAWS CLI—在 Windows、macOS 與 Linux 上執行 AWS IoT 的命令。這些命令可讓您建立並管理物件、憑證、規則和政策。若要開始使用，請參閱 AWS Command Line Interface User Guide。如需 AWS IoT 命令的詳細資訊，請參閱 AWS CLI Command Reference 中的 iot。
- AWS IoT API—使用 HTTP 或 HTTPS 請求建置您的 IoT 應用程式。這些 API 動作可讓您以程式設計的方式建立並管理物件、憑證、規則和政策。關於 AWS IoT 的 API 動作的詳細資訊，請參閱 API 參考中的 Actions (動作)AWS IoT。
- AWS SDK—使用語言特定的 API 建立您的 IoT 應用程式。這些 SDK 整合了 HTTP/HTTPS API，可讓您以任何支援的語言編寫程式。如需詳細資訊，請參閱 AWS 開發套件與工具。
- AWS IoT 裝置 SDK—建立在裝置上執行的應用程式，與 AWS IoT 互相傳送和接收訊息。如需詳細資訊，請參閱 AWS IoT SDK。

相關服務

AWS IoT 直接整合了下列 AWS 服務：

- Amazon Simple Storage Service—在 AWS 雲端提供可擴展的儲存。如需詳細資訊，請參閱 Amazon S3。
- Amazon DynamoDB—提供受管的 NoSQL 資料庫。如需詳細資訊，請參閱 Amazon DynamoDB。
- Amazon Kinesis—可即時處理大規模的串流資料。如需詳細資訊，請參閱 Amazon Kinesis。
- AWS Lambda—從 Amazon EC2 的虛擬伺服器執行您的程式碼以回應事件。如需詳細資訊，請參閱 AWS Lambda。
- Amazon Simple Notification Service—傳送或接收通知。如需詳細資訊，請參閱 Amazon SNS。
- Amazon Simple Queue Service—將資料存放於佇列以供應用程式擷取。如需詳細資訊，請參閱 Amazon SQS。

AWS IoT 運作方式

AWS IoT 可讓已與網際網路連線的裝置連線至 AWS 雲端，並讓雲端中的應用程式與其互動。一般的 IoT 應用程式會收集並處理裝置的遙測資料，或是讓使用者能夠遠端控制裝置。

連接至 AWS IoT 的每個裝置狀態會存放在裝置影子中。裝置影子服務會回應請求以擷取或更新裝置狀態資料，藉此管理裝置影子。裝置影子服務可讓裝置與應用程式通訊，以及讓應用程式與裝置通訊。

透過使用 X.509 憑證，保護裝置和 AWS IoT 之間的通訊。AWS IoT 可以產生憑證，您也可以使用自己的憑證。無論何種情況，憑證都必須透過 AWS IoT 登錄並啟動，然後複製到您的裝置上。當您的裝置與 AWS IoT 通訊時，裝置會向 AWS IoT 提供憑證做為登入資料。

建議所有連線至 AWS IoT 的裝置，在登錄檔中都要有一個項目。登錄檔會存放裝置與該裝置使用之憑證的相關資訊，以保護與 AWS IoT 之間的通訊。

你可以建立規則，定義根據訊息之資料應執行的一個或多個動作。例如，您可以插入、更新或查詢 DynamoDB 表格，或是叫用 Lambda 函數。規則使用表達式篩選訊息。規則與訊息相符時，規則引擎會使用選取的屬性觸發動作。規則也包含 IAM 角色，授與 AWS IoT 許可使用用於執行動作的 AWS 資源。

AWS IoT 入門

本教學課程說明如何使用 AWS IoT 建立所需的資源，以便使用裝置發送、接收、處理 MQTT 訊息。您使用 MQTT 用戶端以模擬 IoT 裝置。

如需 AWS IoT 的詳細資訊，請參閱 什麼是 AWS IoT (p. 1)？

主題

- 登入 AWS IoT 主控台。 (p. 4)
- 在登錄檔中登錄裝置 (p. 5)
- 設定您的裝置 (p. 25)
- 以 AWS IoT MQTT 用戶端查看裝置的 MQTT 訊息。 (p. 25)
- 規則的設定與測試 (p. 27)
- 建立和追蹤的 AWS IoT 任務 (p. 38)

登入 AWS IoT 主控台。

若您沒有 AWS 帳戶，請建立帳戶。

建立 AWS 帳戶:

1. 請開啟 AWS 首頁 並選擇 建立 AWS 帳戶。
2. 請遵循線上指示進行。部分註冊程序需接收來電，並使用電話鍵盤輸入 PIN 碼。
3. 登入 AWS 管理主控台，並開啟 AWS IoT 主控台。
4. 請選擇 Welcome (歡迎) 頁面上的 Get started (開始使用)。

若您是第一次使用 AWS IoT 主控台，則會出現 Welcome to the AWS IoT Console (歡迎使用 IoT 主控台) 頁面。

在登錄檔中登錄裝置

連線到 AWS IoT 的裝置在 AWS IoT 登錄檔中以「IoT 實物」表示。登錄檔可保留向您 AWS IoT 帳戶登錄之所有裝置的記錄。

在登錄檔中登錄您的裝置

1. 在導覽窗格中的 Welcome to the AWS IoT Console (歡迎使用 AWS IoT 主控台) 頁面，選擇 Manage (管理)。

2. 請在 You don't have any things yet (尚無任何實物) 的頁面上，選擇 Register a thing (登錄實物)。

3. 在 Creating AWS IoT things (建立 IoT 物件) 頁面上，選擇 Create a single thing (建立單一物件)。

4. 在 Create a thing (建立實物) 頁面的 Name (名稱) 欄位中，輸入您實物的名稱，例如 **`MyIotThing`**。選擇 Next (下一步)。

   Note

   我們不建議在實物名稱中使用個人識別資訊。

5. 在 Add a certificate for your thing (新增實物的憑證) 頁面上，選擇 Create certificate (建立憑證)。將產生 X.509 憑證和金鑰對。

6. 在 Certificate created! (憑證已建立！) 頁面上，下載您的公開與私密金鑰、憑證和根憑證授權單位 (CA)：

   a. 對您的憑證，選擇 Download (下載)。

   b. 對您的私密金鑰，選擇 Download (下載)。

   c. 對於 Amazon 根 CA，選擇 Download (下載)。新的網頁會顯示：選擇 RSA 2048 bit key: Amazon Root CA 1 (RSA 2048 位元金鑰：Amazon Root CA 1)。這會開啟另一個網頁，選擇 CA Certificates for Service Authentication (用於服務驗證的 CA 憑證)。在以下網頁上，選擇 Amazon Root CA 1 (Amazon 根 CA 1)，這將顯示一個頁面，其中包含根 CA 憑證的文字。將此文字複製並貼至名為 `Amazon_Root_CA_1.pem` 的檔案中。

多數的網頁瀏覽器會將下載的檔案儲存於 Downloads (下載) 目錄。執行範例應用程式時把這些檔案複製到不同的目錄。選擇 Activate (啟用) 來啟用 X.509 憑證，然後選擇 Attach a policy (連接政策)。

7. 在 Add a policy for your thing (新增實物的政策) 頁面上，選擇 Register Thing (註冊實物)。

   註冊實物後，建立新的政策並將其與憑證連接。

8. 在 AWS IoT 主控台的導覽窗格中，選擇 Secure (保護)，然後選擇 Policies (政策)。

   選擇 Create (建立)。

9. 在 Create policy (建立政策) 頁面上：
   a. 為政策輸入 Name (名稱)，例如 **`MyIotPolicy`**。
   b. 針對 Action (動作)，輸入 **`iot:*`**。針對 Resource ARN (資源 ARN)，輸入 **`*`**。
   c. 在 Effect (效果) 下方，選擇 Allow (允許)，然後選擇 Create (建立)。

   此政策可讓您的裝置對所有 AWS IoT 資源執行所有 AWS IoT 動作。

   > **Important**
   >
   > 這些設定太過寬鬆。在生產環境中，將許可範圍縮小至您裝置所需的範圍。如需更多詳細資訊，請參閱 授權 (p. 145)。

10. 選擇 Manage (管理)，然後選擇 AWS IoT 實物。

11. 選擇 Security (安全性)。

12. 選擇憑證。
13. 在憑證詳細資訊頁面中，選擇 Actions (動作)，然後選擇 Attach policy (連接政策)。

14. 選擇您建立的政策 (MyIotPolicy)，然後選擇 Attach (連接)。

建立或啟用裝置憑證

透過使用 X.509 憑證，保護裝置和 AWS IoT 之間的通訊。AWS IoT 可以產生為您憑證，您也可以使用自己的 X.509 憑證。本教學課程是由 AWS IoT 來產生 X.509 憑證。需先啟用憑證才可開始使用。

1. 選擇 Create certificate (建立憑證)。

2. Certificate created! (已建立憑證！) 頁面上，選擇憑證、私有金鑰和 AWS IoT 根 CA 的 Download (下載) (不需下載公開金鑰)。將各個項目儲存至電腦，再選擇 Activate (啟用) 繼續進行。

   Note

   AWS IoT Download (下載) 連結的根 CA 會帶您至 X.509 用戶端憑證 (p. 132) 頁面，您可以在這裡選擇 CA 憑證。與頁面上的其他 Download (下載) 連結不同，它不會直接下載檔案。

請注意，下載的檔案名稱可能與列在 Certificate created! (已建立憑證！) 頁面上的不同。例如：

- 2a540e2346-certificate.pem.crt
- 2a540e2346-private.pem.key
- 2a540e2346-public.pem.key

> **Note**
>
> 雖然可能性不大，但根 CA 憑證會有過期和遭撤銷的可能。若發生此種情況，請務必將新的根 CA 憑證複製到您的裝置上。

3. 選擇 Done (完成)，來返回 AWS IoT 主控台的主頁面。

需要使用裝置時，您必須將私有金鑰和 rootCA 憑證複製到裝置。本指南中的指示是以假設您不是使用裝置來撰寫的，且只需熟悉 AWS IoT 主控台。

建立 AWS IoT 政策

您的裝置與 AWS IoT 之間的驗證所用的是 X.509 憑證。會使用 AWS IoT 政策來授權您的裝置執行 AWS IoT 操作，例如訂閱或發佈至 MQTT 主題。您的裝置將在傳送訊息至 AWS IoT 時出示憑證。請務必建立 AWS IoT 政策並將之連接到裝置憑證，才能讓裝置執行 AWS IoT 的運作。

若要建立 AWS IoT 政策

1. 請在左側導覽窗格中選擇 Secure (安全)，然後選擇 Policies (原則)。請在 You don't have a policy yet (尚無任何政策) 頁面上，選擇 Create a policy (建立政策)。

2. 請在 Create a policy (建立政策) 頁面上的 Name (名稱) 欄位中輸入政策的名稱 (例如 **`MyIotPolicy`**)。

   Note

   我們不建議在政策名稱中使用個人識別資訊。

   請在 Action (動作) 欄位中輸入 **`iot:Connect`**。請在 Resource ARN (資源 ARN) 欄位中輸入 **`*`**。請選取 Allow (允許) 的核取方塊。如此能讓所有用戶端連線到 AWS IoT。

> **Note**
>
> 您可以將用戶端 ARN 指定為資源，以限制可以連線的用戶端 (裝置)。用戶端 ARN 遵循以下格式：
> `arn:aws:iot:your-region:your-aws-account:client/<my-client-id>`

請選擇 Add Statement (新增聲明) 按鈕，新增另一項政策聲明。請在 Action (動作) 欄位中輸入 **`iot:Publish`**。請在 Resource ARN (資源 ARN) 欄位中輸入您的裝置要發佈到的主題的 ARN。

> **Note**
>
> 主題 ARN 遵循以下格式：
> `arn:aws:iot:your-region:your-aws-account:topic/<your/topic>`
> 例如：
> `arn:aws:iot:us-east-1:123456789012:topic/my/topic`

最後請選擇 Allow (允許) 的核取方塊。如此可讓您的裝置發佈訊息到指定的主題。

3. 政策相關資訊輸入完畢後，請選擇 Create (建立)。

如需詳細資訊，請參閱管理 AWS IoT 政策。

將 AWS IoT 政策連接至裝置憑證

您已建立了一個政策，請務必將政策連接至裝置憑證。將 AWS IoT 政策連接至憑證，即可讓裝置獲得政策中指定的許可。

1. 請在左側導覽窗格中選擇 Secure (安全)，然後選擇 Certificates (憑證)。

2. 請在您建立的憑證欄位中，選擇 ...，以打開下拉式選單，然後選擇 Attach policy (連接政策)。

3. 請在 Attach policies to certificate(s) (將政策連接至憑證) 中，選取您在上一步中建立的政策旁的核取方塊，並選擇 Attach (連接)。

將憑證連接到物件

裝置必須擁有憑證、私有金鑰和根 CA 憑證，才能與 AWS IoT 進行驗證。建議您也將裝置憑證連接至在 AWS IoT 中代表您裝置的 IoT 實物。如此可建立 AWS IoT 政策，讓政策根據連接到物件的憑證來授與許可。如需詳細資訊，請參閱 物件原則變數 (p. 150)。

將憑證連接至登錄檔中代表您裝置的物件

1. 請在您建立的憑證欄位中，選擇 ...，以打開下拉式選單，再選擇Attach thing (連接物件)。

2. 請在 Attach things to certificate(s) (將物件連接至憑證) 中，選取您在上一步中註冊的物件，並選擇 Attach (連接)。

3. 請選取憑證的方塊，以驗證連接的物件。

4. 請在憑證的 Details (詳細資訊) 頁面左側的導覽窗格中，選擇 Things (物件)。

5. 若要驗證連接的政策，請在憑證的 Details (詳細資訊) 頁面左側的導覽窗格中，選擇 Policies (政策)。

設定您的裝置

所有裝置都必須安裝裝置憑證、私有金鑰和根 CA 憑證，才能與 AWS IoT 通訊。請參閱裝置的文件，連線到裝置，並將裝置的憑證、私有金鑰、根 CA 憑證複製到裝置上。

如果您沒有具備 IoT 功能的裝置，您可以使用 MQTT 用戶端、AWS IoT 裝置軟體開發套件或 AWS CLI。如需詳細資訊，請參閱 在 Raspberry Pi 上使用 AWS IoT SDK (p. 91) 節。教學課程會使用 Raspberry Pi，但您可以輕鬆調整，以用於其他類型的電腦。

以 AWS IoT MQTT 用戶端查看裝置的 MQTT 訊息。

您可以使用 AWS IoT MQTT 用戶端來更清楚地了解裝置所傳送的 MQTT 訊息。

裝置會發佈 MQTT 訊息至主題。您可以使用 AWS IoT MQTT 用戶端來訂閱主題，以查看這些訊息。

若要查看 MQTT 訊息

1. 請在 AWS IoT 主控台 左側的導覽窗格中，選擇 Test (測試)。

2. 請訂閱 IoT 實物發佈的主題。接續此範例，請在 Subscribe to a topic (訂閱主題) 的 Subscription topic (訂閱主題) 欄位中，輸入 **`my/topic`**，再選擇 Subscribe to topic (訂閱主題)。

Subscribe (訂閱) 欄位會顯示 **`my/topic`** 主題。

若要模擬傳送訊息的 IoT 實物

- 請在 MQTT 用戶端頁面的 Publish (發佈) 區塊中，於 Specify a topic and a message to publish (指定主題和欲發佈的訊息) 欄位中，輸入 **`my/topic`**。

  > **Note**
  >
  > 我們不建議在主題名稱中使用個人識別資訊。

  在訊息承載區段中，輸入下列 JSON：

```
{
    "message": "Hello, world",
    "clientType": "MQTT client"
}
```

請選擇 Publish to topic (發佈至主題)。您應該會在 AWS IoT MQTT 用戶端看到該訊息。(在 Subscription(訂閱) 中選擇 my/topic (我的/主題) 欄位來查看該訊息。)

規則的設定與測試

AWS IoT 規則引擎會接聽與規則相符的 MQTT 傳入訊息。接收到符合規則的訊息時，規則會對 MQTT 訊息中的資料採取某些動作 (例如寫入資料至 Amazon S3 儲存貯體、呼叫 Lambda 函數、發送訊息至 Amazon SNS 主題)。在此步驟中，您建立並設定一項規則，將從裝置接收到的資料傳送至 Amazon SNS 主題。具體而言，您：

- 建立 Amazon SNS 主題。
- 使用手機號碼訂閱 Amazon SNS 主題。
- 建立一項規則，當從裝置接收到訊息時，便傳送訊息至 Amazon SNS 主題。
- 使用 MQTT 用戶端測試規則。

建立 SNS 主題。

請使用 Amazon SNS 主控台建立 Amazon SNS 主題。

> **Note**
>
> 並非所有 AWS 區域皆可使用 Amazon SNS。

1. 開啟 Amazon SNS 主控台。
2. 請在左側窗格中選擇 Topics (主題)。

3. 選擇 Create topic (建立主題)。

4. 請輸入主題名稱和顯示名稱，再選擇 Create topic (建立主題)。

Note

我們不建議在 Amazon SNS 主題名稱中使用個人識別資訊。

5. 請記住您剛建立的主題 ARN。

訂閱 Amazon SNS 主題

若要在手機接收 SMS 簡訊，請訂閱 Amazon SNS 主題。

1. 請在 Amazon SNS 主控台中，選取您剛建立的主題旁的核取方塊。請在 Actions (動作) 選單中選擇 Subscribe to topic (訂閱主題)。

2. 請在 Create subscription (建立訂閱) 的 Protocol (協定) 下拉式清單中，選擇 SMS。

請在 Endpoint (端點) 欄位中，輸入具 SMS 功能的手機的號碼，再選擇 Create subscription (建立訂閱)。

> **Note**
>
> 輸入的電話號碼僅能含有數字和破折號。

Amazon SNS 主控台會顯示下列訊息，但您可能不會收到確認訊息。

建立規則

AWS IoT 規則是由主題篩選條件、規則動作所組成，在大多數情況下還包含 IAM 角色。若主題篩選條件觸發了規則，便會將訊息發佈至主題。規則動作定義了觸發規則時要採取的動作。IAM 角色包含了一個以上的 IAM 政策，用以判定規則可存取哪些 AWS 服務。您可以建立多個監聽同一主題的規則。同樣地，也可以建立一項會受多個主題觸發的規則。AWS IoT 規則引擎會持續處理符合規則所定義的主題篩選條件而發佈到主題上的訊息。

在此例中，建立一項規則，使用 Amazon SNS 傳送 SMS 通知至某個手機號碼。

1. 請在 AWS IoT 主控台左側的導覽窗格中，選擇 Act (動作)。

2. 請在 Act (動作) 頁面選擇 Create a rule (建立規則)。

3. 在 Create a rule (建立規則) 頁面，於 Name (名稱) 欄位輸入規則的名稱。

   Note

   我們不建議在規則名稱中使用個人識別資訊。

   在 Description (描述) 欄位輸入對該項規則的描述。

4. 向下捲動到 Rule query statement (規則查詢陳述式)。請在 Using SQL version (使用的 SQL 版本) 的下拉式清單選擇最新版本。在 Rule query statement (規則查詢陳述式) 欄位中，輸入 **`SELECT * FROM 'my/topic'`**。

   `SELECT *` 表示您需要傳送觸發規則的整個 MQTT 訊息。主題篩選條件是 `FROM 'my/topic'`。規則引擎使用主題篩選條件決定收到 MQTT 訊息時觸發的規則。

5. 請在 Set one or more actions (設定一個或多個動作) 中選擇 Add action (新增動作)。

6. 請在 Select an action (選取動作) 頁面中，選擇 Send a message as an SNS push notification (傳送 SNS 推送通知形式的訊息)，並選擇 Configure action (設定動作)。

Select an action

Select an action.

- Insert a message into a DynamoDB table
  DYNAMODB
- Split message into multiple columns of a DynamoDB table (DynamoDBv2)
  DYNAMODBV2
- Send a message to a Lambda function
  LAMBDA
- Send a message as an SNS push notification
  SNS
- Send a message to an SQS queue
  SQS
- Send a message to an Amazon Kinesis Stream
  AMAZON KINESIS
- Republish a message to an AWS IoT topic
  AWS IOT REPUBLISH
- Store a message in an Amazon S3 bucket
  S3
- Send a message to an Amazon Kinesis Firehose stream
  AMAZON KINESIS FIREHOSE
- Send message data to CloudWatch
  CLOUDWATCH METRICS
- Change the state of a CloudWatch alarm
  CLOUDWATCH ALARMS
- Send a message to the Amazon Elasticsearch Service
  AMAZON ELASTICSEARCH
- Send a message to a Salesforce IoT Input Stream
  SALESFORCE IOT
- Send a message to IoT Analytics
  IOT ANALYTICS
- Send a message to an IoT Events Input
  IOT EVENTS
- Start a Step Functions state machine execution
  STEP FUNCTIONS

Cancel

Configure action

7. 在 Configure action (設定動作) 頁面的 SNS target (SNS 目標) 下，選擇 Select (選取) 以展開 SNS 主題。接著在您先前建立的 Amazon SNS 主題旁選擇 Select (下一步)。在 Message format (訊息格式) 下，選擇 JSON。

8. 請授與 AWS IoT 許可，以在觸發規則時代表您發佈至 Amazon SNS 主題。選擇 Create a new role (建立新角色)。在 IAM role name (IAM 角色名稱) 中，輸入新的角色名稱，然後選擇 Create a new role (建立新的角色)。

9. 在 IAM role name (IAM 角色名稱) 下，選擇 Update role (更新角色) 將許可套用到新建立的角色。選擇該角色，然後選擇 Add action (新增動作)。

10. 請在 Create a Rule (建立規則) 頁面上，選擇 Create rule (建立規則)。

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

MyIoTRule

Description

A rule for AWS IoT Getting Started

Rule query statement

Indicate the source of the messages you want to process with this rule.

Using SQL version

2016-03-23

Rule query statement

SELECT <Attribute> FROM <Topic Filter> WHERE <Condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature > 50. To learn more, see AWS IoT SQL Reference.

```
1 SELECT * FROM 'mytopic'
```

Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required)

Send a message as an SNS push notification
My_IoT_SNS_Topic

Remove Edit

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

Add action

Tags

Apply tags to your resources to help organize and identify them. A tag consists of a case-sensitive key-value pair. Learn more about tagging your AWS resources.

Tag name

Provide a tag name, e.g. Manufacturer

Value

Provide a tag value, e.g. Acme-Corporation

Clear

Add another

Cancel

Create rule

更多關於建立規則的資訊，請參閱 AWS IoT規則。

測試 Amazon SNS 規則

您可以使用 AWS IoT MQTT 用戶端來測試您的規則。

1. 請在 AWS IoT 主控台 左側的導覽窗格中，選擇 Test (測試)。
2. 請在 MQTT 用戶端頁面的 Publish (發佈) 區塊中，於 Specify a topic and a message to publish (指定主題和欲發佈的訊息) 中輸入 **`my/topic`** 或規則中所使用的主題。在訊息承載區段中，輸入下列 JSON：

```
{
    "default": "Hello, from AWS IoT console",
    "message": "Hello, from AWS IoT console"
}
```

3. 請選擇 Publish to topic (發佈至主題)。您的手機上應該會收到 Amazon SNS 簡訊。

恭喜您！ 您已成功建立和設定規則，能夠將從裝置接收到的資料傳送至 Amazon SNS 主題。

後續步驟

如需更多 AWS IoT 規則的相關資訊，請參閱 AWS IoT 規則教學 (p. 52)和 AWS IoT 規則 (p. 254)。

建立和追蹤的 AWS IoT 任務

AWS IoT 任務可讓您部署和追蹤裝置機群中的管理任務。您可以使用任務來一次傳送遠端動作到一或多個裝置，控制部署任務到您的裝置，以及針對每個裝置追蹤任務執行目前和過去的狀態。

此主題向您說明如何建立和部署範例任務到裝置。它會逐步帶領您完成建立任務所需的步驟，以及追蹤設定為與 AWS IoT 通訊的裝置上的事件。這些指示是假設您使用 Raspberry Pi 所撰寫的，但它們可以修改以順應其他以 Linux 為基礎的裝置。

以下是使用任務的可能情況：

- 更新裝置韌體、軟體或檔案，例如安全憑證。
- 執行管理任務，例如重新啟動裝置或執行診斷。
- 還原裝置為原廠設定或其他已知良好的狀態。

連接至您的裝置至 AWS IoT。

請執行下列步驟來連接 Raspberry Pi 到 AWS IoT。

1. 完成 連接您的 Raspberry Pi 教學課程。當您完成後，您將在您的名為 `MyRaspberryPi` 的 AWS 帳戶中有一個註冊的 AWS IoT 實物。此外，也會在您的裝置上有完整設定的安全憑證。
2. 完成 使用適用於 JavaScript 的 AWS IoT 裝置開發套件教學課程。當您完成後，您的裝置連接到 AWS IoT，而且您可以執行 AWS IoT Device SDK for JavaScript 隨附的範本程式碼。

您的裝置現在已就緒可使用 AWS IoT 任務。

執行任務範例

AWS IoT Device SDK for JavaScript 包含名為 jobs-example.js 的範例。此範例可以從 AWS IoT 主控台 接收訊息以驗證連接。它也可以接收和處理源自 AWS IoT 服務的任務執行。

您可以使用下列命令來執行此範例。使用 Raspberry Pi 的 REST 端點當做 `-H` 參數值。

```
node examples/jobs-example.js -f ~/certs -H <PREFIX>.iot.<REGION>.amazonaws.com -
T thingName
```

如果您已建立設定檔，其中包含實物名稱和主機端點 (您的裝置的 REST 端點)，您可以使用以下命令。

```
node examples/jobs-example.js -f ./certs -F your config file name.json
```

建立任務文件

JSON 文件的任務文件提供您的裝置執行任務所需的所有資訊。AWS IoT Device SDK for JavaScript 使用名為 `operation` 的屬性來路由任務文件到特定的處理常式。該 `jobs-example.js` 計劃具有名為 customJob 操作的範例處理常式。若要為這個處理常式建立名為 `example-job.json` 的任務文件，則檔案應該包含下列 JSON 物件。

```
{
    "operation":"customJob",
    "otherInfo":"someValue"
}
```

如需更多範例任務文件，請參閱文件的 jobs-agent.js 範例。

建立任務

現在您已準備好建立任務，傳遞任務文件至您指定的所有裝置。若要建立任務，可使用 AWS IoT 主控台、AWS IoT 軟體開發套件或 AWS IoT CLI。

以下範例顯示如何使用 AWS IoT CLI 來建立任務。

```
 aws iot create-job \
--job-id "example-job-01" \
--targets "arn:aws:iot:::thing/MyRaspberryPi" \
--document file:///example-job.json \
--description "My First test job" \
--target-selection SNAPSHOT
```

如果您將任務文件存放在 Amazon Simple Storage Service 中，請使用 `-document-source` 參數而不是 `-document` 參數來指定任務文件的 Amazon S3 URL。

如果您偏好使用 AWS IoT 主控台，請依照以下步驟建立任務。

1. 將任務文件上傳到 Amazon S3 儲存貯體。如需詳細資訊，請參閱 Amazon Simple Storage Service Console User Guide 中 如何將檔案與資料夾上傳到 S3 儲存貯體？

2. 在 AWS IoT 主控台中，選擇 Manage (管理)，然後選擇 Jobs (任務)。
3. 選擇 Create a job (建立任務)。

4. 在 Select a job (選取任務) 頁面上，選擇 Create custom job (建立自訂任務)。

CREATE JOB
Select a job
AWS IoT Device Management job orchestration and notification
set of remote operations called jobs that are sent to and execut
connected to AWS IoT.
Create a custom job
Send a request to acquire an executable job file from one of you
devices connected to AWS IoT.
Create an Amazon FreeRTOS Over-the-air (OTA) up
This OTA update job will send your firmware image securely ove
FreeRTOS-based devices
Create a Greengrass Core update job
Create a snapshot job to update one or more Greengrass Core d
Greengrass Core or OTA agent version.

5. 在 Create a job (建立任務) 頁面上，輸入唯一的任務 ID。

   Note

   我們不建議在任務 ID 中使用個人識別資訊。

   在 Select devices to update (選擇要更新的裝置) 下，選擇已連接到 AWS IoT 的裝置。

CREATE JOB
Create a job
Job ID
example-job-01
Description
Select devices to update
Browse and select the devices you want to include in this job.
1 thing(s) and 0 thing group(s) selected.
Things
Thing Groups
Summary
Search
MyRaspberryPi

6. 向下捲動到 Add a job file (新增任務檔案)，然後選擇已上傳到 Amazon S3 的任務文件檔案。在 Job type (任務類型) 下，選擇 Your job will complete after deploying to the selected devices/groups (snapshot) (您的任務將會在部署到選定的裝置/群組 (快照) 之後完成)。(另一選項 Your job will continue deploying to any devices added to the selected groups (continuous)(您的任務將會繼續部署到任何新增至選定群組 (連續) 的裝置)，是用於當有裝置新增到每個群組時部署任務到裝置群組)。 保持 Job executions rollout configuration (任務執行推展組態) 不變。選擇 Create (建立)。

Add a job file

Upload a job file that defines what your job should do.

example-job.json

Pre-sign resource URLs

For an extra layer of security, you can pre-sign URLs that refer to re

Cannot find pre-sign url placeholder in the job file. Skip pre-sign co

Job type

A job can run on the devices and/or groups selected, or remain ope

Your job will complete after deploying to the selected devic

Your job will continue deploying to any devices added to th

Job executions rollout configuration

Specify how quickly devices will be notified of a pending job execut

Maximum per minute (1-1000)

1000

7. 您的新任務會顯示在 Jobs (任務) 頁面上。

如需有關建立和部署任務的詳細資訊，請參閱 AWS IoT 任務。

在裝置上執行任務

建立任務之後，任務服務會傳送待處理任務的通知到您的裝置。您的裝置可透過 NextJobExecutionChanged API 取得任務詳細資訊和任務文件。您已執行的 `jobs-example.js` 範例，會在裝置上執行任務。當任務完成之後，範例會使用 UpdateJobExecution API 發佈其完成狀態。當您在裝置上執行範例時，會看到下列輸出。

```
node examples/jobs-example.js -f ./certs -F config.json
connect
startJobNotifications completed for thing: MyRaspberryPi
customJob operation handler invoked, jobId: example-job-01
```

當您重新整理 Jobs (任務) 頁面時，您可以看到您的任務已成功完成。

追蹤任務進度和任務執行活動

您可以使用 `Job` 事件和 `JobExecution` 事件來追蹤任務的進度。

這是一個實用的方式來提醒使用者、系統管理員或系統其他成員任務完成，或任務執行已變更其狀態。例如，您可以提醒使用者有關裝置上的韌體更新，或通知系統管理員其裝置機群有關問題需要進行調查和加以解決。

此範例中之任務的任務事件，會在任務完成或取消時發佈到下列主題。

```
$aws/events/job/example-job-01/completed
$aws/events/job/example-job-01/canceled
```

此範例中之任務的任務執行事件，會在任務執行達到可能的最終狀態之一時，傳送到下列主題。

```
$aws/events/jobExecution/example-job-01/succeeded
$aws/events/jobExecution/example-job-01/failed
$aws/events/jobExecution/example-job-01/rejected
$aws/events/jobExecution/example-job-01/canceled
$aws/events/jobExecution/example-job-01/removed
```

當裝置上的任務執行成功時，AWS IoT 會發佈 JobExecution 已成功的事件。您可導覽到 AWS IoT Test (測試) 頁面並訂閱 MQTT 用戶端中的 `$aws/events/jobExecution/example-job-01/succeeded` 主題，來查看此事件。

當裝置的任務執行已成功完成時，會顯示以下訊息。

AWS IoT 也會發佈完成的 `job` 事件。您可以訂閱 MQTT 用戶端中的 `$aws/events/job/example-job-01/completed` 主題，來查看此事件。

Publish

Specify a topic and a message to publish with a QoS of 0.

$aws/events/job/example-job-01/completed

```
1 {
2   "message": "Hello from AWS IoT console"
3 }
```

$aws/events/job/example-job-01/completed Dec 19, 201


```
{
  "eventType": "JOB",
  "eventId": "a1817d74-0f1d-42b3-b03f-accfc90af41e",
  "timestamp": "1513709645",
  "operation": "completed",
  "jobId": "example-job-01",
  "status": "COMPLETED",
  "targetSelection": "SNAPSHOT",
  "targets": [
    "arn:aws:iot:us-east-1:[illegible]:thing/MyRaspb
  ],
  "description": "Job example-job-01 for Thing MyRaspb
  "completedAt": "1513709645105",
  "createdAt": "1513709615488",
  "lasetUpdatedAt": "1513709645105",
  "jobProgressDetails": {
    "numberOfCanceledThings": 0,
    "numberOfRejectedThings": 0,
    "numberOfFailedThings": 0,
    "numberOfRemovedThings": 0,
    "numberOfSucceededThings": 1
  }
```

AWS IoT 規則教學

以下教學說明如何建立與測試 AWS IoT 規則。開始之前，請務必完成 AWS IoT 入門教學 (p. 4)。它說明如何建立 AWS 帳戶並在 AWS IoT 中登錄裝置，此兩個動作是演練這些教學的前提。

此教學中的案例含有植物資料列的溫室。每個植物都有濕氣感應器。濕氣感應器會依預定間隔，將資料傳送至 AWS IoT。AWS IoT 規則引擎會接收此資料並將其寫入 DynamoDB 資料表。您可以建立一個規則，使用 AWS IoT MQTT 用戶端將資料寫入 DynamoDB 和模擬感應器。

AWS IoT 規則包含了 SQL SELECT 陳述式、主題篩選條件和規則動作。裝置藉由發佈訊息至 MQTT 主題以傳送資訊給 AWS IoT。SQL SELECT 陳述式讓您可從傳入的 MQTT 訊息擷取資料。AWS IoT 規則的主題篩選條件可指定一個或多個 MQTT 主題。在主題上收到符合主題篩選條件的 MQTT 訊息時，就會觸發規則。規則動作可讓您獲得從 MQTT 訊息擷取的資訊並傳送至另一個 AWS 服務。規則動作是針對如 Amazon DynamoDB、AWS Lambda、Amazon SNS、Amazon S3 等 AWS 服務而定義。透過使用 Lambda 規則，您可以呼叫其他 AWS 或第三方 Web 服務。如需規則動作的完整清單，請參閱 AWS IoT 規則動作 (p. 260)。

在這些教學中，我們假設您使用 AWS IoT MQTT 用戶端，也使用 `my/greenhouse` 做為規則中的主題篩選條件。

您也可以使用自己的裝置，但您必須知道裝置會發佈什麼 MQTT 主題，這樣才能在規則中將其指定為主題篩選條件。如需詳細資訊，請參閱 AWS IoT 規則 (p. 254)。

使用 DynamoDB 動作建立規則

DynamoDB 動作可讓您從傳入的 MQTT 訊息獲得資訊並寫入 DynamoDB 表格。

建立 DynamoDB 規則

1. 在 AWS IoT 主控台的導覽窗格中，選擇 Act (動作)。

2. 在 Rules (規則) 頁面，選擇 Create (建立)。

3. 在 Create a rule (建立規則) 頁面中，輸入規則名稱和描述。

   Note

   我們不建議在規則名稱或描述中使用個人識別資訊。

4. 在 Rule query statement (規則查詢陳述式) 下，請從 Using SQL version (使用 SQL 版本) 清單中選擇最新版本。針對 Rule query statement (規則查詢陳述式)，輸入：

```
SELECT * FROM 'my/greenhouse'
```

   ("`SELECT *`" 會指定您要傳送觸發規則的整個 MQTT 訊息。"`FROM 'my/greenhouse'`" 會告知規則引擎，當收到的 MQTT 訊息主題符合主題篩選條件時觸發此規則。選擇 Add action (新增動作)。

5. 在 Select an action (選取動作) 上，選擇 Insert a message into a DynamoDB table (插入訊息至 DynamoDB 資料表)，並選擇 Configure action (設定動作)。

6. 在 Configure action (設定動作) 上，選擇 Create a new resource (建立新的資源)。

7. 在 Amazon DynamoDB 頁面，選擇 Create table (建立表格)。

8. 在 Create DynamoDB table (建立 DynamoDB 資料表)上，輸入名稱。在 Partition key (分區索引鍵)，輸入 **`Row`**。選擇 Add sort key (新增排序鍵)，然後在 Sort key (排序鍵) 欄位中輸入 **`PositionInRow`**。`Row` 代表溫室中的植物列數。`PositionInRow` 代表植物在列中的位置。選擇分割區和排序索引鍵同時適用於 String (字串)，然後選擇 Create (建立)。建立您的 DynamoDB 表格需要幾秒鐘時間。將開啟 Amazon DynamoDB 主控台的瀏覽器標籤關閉。若不關閉標籤，您的 DynamoDB 資料表就不會顯示於 AWS IoT 主控台 Configure action (設定動作) 頁面上的 Table name (資料表名稱) 清單。

9. 在 Configure action (設定動作) 上，從 Table name (資料表名稱) 清單選擇您的新資料表。在 Partition key value (分割區索引鍵值) 中，輸入 **`${row}`**。這樣會指示規則從 MQTT 訊息取得 `row`屬性值，並將

其寫入 DynamoDB 資料表中的 Row 欄。在 Sort key value (排序索引鍵值) 中，輸入 **`${pos}`**。這會將 `pos` 屬性的值寫入 PositionInRow 欄。在 Write message data to this column (寫入訊息資料至此欄) 中輸入 **`Payload`**。這會將訊息承載插入 `Payload` 欄。將 Operation (操作) 保持空白。此欄位可讓您指定觸發動作時要執行的操作 (INSERT、UPDATE 或 DELETE)。選擇 Create a new role (建立新角色)。

10. 在 Create a new role (建立新角色) 中輸入唯一名稱，然後選擇 Create role (建立角色)。

11. 選擇 Add action (新增動作)。

12. 選擇 Create rule (建立規則)。

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

GreenhouseRule

Description

A DynamoDB rule for a greenhouse

Rule query statement

Indicate the source of the messages you want to process with this rule.

Using SQL version

2016-03-23

Rule query statement

SELECT <Attribute> FROM <Topic Filter> WHERE <Condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature > 50. To learn more, see AWS IoT SQL Reference.

```
1 SELECT * FROM 'my/greenhouse'
```

Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required)

Insert a message into a DynamoDB table
GreenhouseTable

Remove Edit

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

Add action

Tags

Apply tags to your resources to help organize and identify them. A tag consists of a case-sensitive key-value pair. Learn more about tagging your AWS resources.

Tag name

Provide a tag name, e.g. Manufacturer

Value

Provide a tag value, e.g. Acme-Corporation

Clear

Add another

Cancel

Create rule

使用 DynamoDB 動作測試規則

1. 若要測試規則，請開啟 AWS IoT 主控台，接著從導覽窗格中，選擇 Test (測試)。
2. 請選擇 Publish to a topic (發佈至主題)。在 Publish (發佈) 區段中，輸入 **my/greenhouse**。在訊息區域中，輸入下列 JSON：

```
{
    "row" : "0",
    "pos" : "0",
    "moisture" : "75"
  }
```

回到 DynamoDB 主控台，然後選擇 Tables (資料表)。

選取 GreenhouseTable，然後選擇 Items (項目)。您的資料會顯示在 Items (項目) 索引標籤中。

使用 AWS Lambda 動作建立規則

您可以定義規則來呼叫 Lambda 函式，從觸發規則的 MQTT 訊息傳入資料。這可讓您從傳入的訊息擷取資料，然後呼叫另一個 AWS 或第三方服務。在此教學中，我們假設您已完成了 AWS IoT 入門教學課程 (p. 4)，在此教學課程中您已經建立並訂閱了 Amazon SNS 主題。此時，您會建立一個 Lambda 函數，以將訊息發佈至您在 AWS IoT 入門教學 (p. 4) 中建立的 Amazon SNS 主題。您也會建立 Lambda 規則來呼叫 Lambda 函式，從觸發規則的 MQTT 訊息傳入一些資料。

在此教學中，您會使用 AWS IoT MQTT 用戶端來傳送可觸發規則的訊息。

建立 Lambda 函數

1. 在 AWS Lambda 主控台中，選擇 Create function (建立函數)。

2. 在 Create function (建立函數) 頁面上，選擇 Use a blueprint (使用藍圖)。在 Blueprints (藍圖) 篩選條件欄位中，輸入 **`hello-world`**，然後按 Enter 鍵。選擇 hello-world-python (hello-world-python) 藍圖，然後選擇 Configure (設定)。

3. 在 Basic information (基本資訊) 中，輸入您函數的名稱。

   Note

   我們不建議在規則名稱或描述中使用個人識別資訊。

4. 從 Execution Role (執行角色) 選擇 Create a new role from AWS policy templates (從 AWS 政策範本建立新角色)。輸入角色的名稱。從 Policy templates (政策範本) 選擇 Amazon SNS publish policy (Amazon SNS 發佈政策)。在下拉式選單外按一下以將其關閉。

5. 選擇 Create function (建立函式)。

6. 在 AWS Lambda 主控台中，選擇您 Lambda 函數的名稱。有關您 Lambda 函數的資訊將會顯示。向下捲動至 Function code (函數程式碼) 區段，並將現有的程式碼以下列內容取代：

```
from __future__ import print_function

import json
import boto3

print('Loading function')

def lambda_handler(event, context):

    # Parse the JSON message
    eventText = json.dumps(event)

    # Print the parsed JSON message to the console; you can view this text in the
 Monitoring tab in the Lambda console or in the CloudWatch Logs console
    print('Received event: ', eventText)

    # Create an SNS client
    sns = boto3.client('sns')

    # Publish a message to the specified topic
    response = sns.publish (
      TopicArn = 'arn:aws:iam::123456789012:role/service-role/myLambdaFunctionRole',
      Message = eventText
    )

    print(response)
```

Note

以您先前建立的 Amazon SNS 主題之 ARN 取代 `TopicArn` 的值。

選擇 Save (儲存)。

測試您的 Lambda 函式

1. 在 Lambda 函數詳細資訊頁面右上方的 Select a test event (選取測試事件) 中，選擇 Configure test events (設定測試事件)。

2. 在 Configure test event (設定測試事件) 上，輸入您測試事件的名稱，並以下列內容取代訊息 JSON：

```
{
  "message" : "Hello, world"
}
```

選擇 Create (建立)。

3. 在 Lambda 函數詳細資訊頁面的右上方，選擇 Test (測試) 以搭配您在測試事件中指定的訊息來測試您的 Lambda 函數。

4. 在 Lambda 函數程式碼下的 Execution result (執行結果) 標籤上，您會看到 Lambda 函數的輸出。

使用 Lambda 動作建立規則

此節提供您搭配 Lambda 動作和錯誤動作來建立規則的步驟。Lambda 動作會呼叫您的 Lambda 函數。如果在呼叫 Lambda 函數時發生錯誤，則錯誤動作會發佈訊息到 `lambda/error` MQTT 主題。在您測試規則時，這會十分實用。

1. 瀏覽至 AWS IoT 主控台，然後在導覽窗格中選擇 Act (動作)。

2. 選擇 Create (建立) 以建立一個 AWS IoT 規則。

3. 在 Create a rule (建立規則) 頁面上，輸入您的規則名稱。

4. 在 Rule query statement (規則查詢陳述式) 中輸入以下查詢：

```
SELECT * FROM "my/lambda/topic"
```

5. 請在 Set one or more actions (設定一個或多個動作) 中選擇 Add action (新增動作)。

6. 在 Select an action (選取動作) 下選擇 Send a message to a Lambda function (傳送訊息至 Lambda 函數)，然後再選擇 Configure action (設定動作)。

7. 在 Configure action (設定動作) 上選擇 Select (選取)。

8. 選擇您的 Lambda 函數。

9. 選擇 Add action (新增動作)。

10. 在 Create a rule (建立規則) 頁面上的 Error action (錯誤動作) 中，選擇 Add action (新增動作)。

11. 在 Set an action as error action (將動作設為錯誤動作) 上，選擇 Republish a message to an AWS IoT topic (將訊息重新發佈至 AWS IoT 主題)，然後選擇 Configure action (設定動作)。

12. 在 Configure action (設定動作) 的 Topic (主題) 下輸入 `lambda/error`。

13. 在 Choose or create a role to grant AWS IoT access to perform this action (選擇或建立角色，授與 AWS IoT 存取權限以執行此動作) 下，選擇 Create Role (建立角色)。

14. 在 Create a new role (建立新角色) 中輸入角色名稱，然後選擇 Create role (建立角色)。

15. 在 Configure action (設定動作) 中，選擇 Add action (新增動作)。

16. 在 Create a rule (建立規則) 中，選擇 Create rule (建立規則)。

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

myLambdaRule

Description

Rule query statement

Indicate the source of the messages you want to process with this rule.

Using SQL version

2016-03-23

Rule query statement

SELECT <Attribute> FROM <Topic Filter> WHERE <Condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature > 50. To learn more, see AWS IoT SQL Reference.

```
1 SELECT * FROM "my/lambda/topic"
```

Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required)

Send a message to a Lambda function
myLambdaFunction
Remove Edit

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

Republish a message to an AWS IoT topic
lambda/error
Remove Edit

Tags

Apply tags to your resources to help organize and identify them. A tag consists of a case-sensitive key-value pair. Learn more about tagging your AWS resources.

Tag name	Value	
Provide a tag name, e.g. Manufacturer	Provide a tag value, e.g. Acme-Corporation	Clear

Add another

Cancel Create rule

使用 Lambda 動作測試您的規則

1. 若要測試 Lambda 動作，請開啟 AWS IoT 主控台，接著從導覽窗格中，選擇 Test (測試)。

2. 在 MQTT 用戶端中的 Subscription topic (訂閱主題) 下輸入 `lambda/error`，然後選擇 Subscribe to topic (訂閱至主題)。

3. 在 Publish (發佈) 下輸入 `my/lambda/topic`，然後選擇 Publish to topic (發佈至主題) 以發佈預設 JSON 訊息。

發佈此訊息應觸發規則並呼叫您的 Lambda 函數。您的 Lambda 函數會推送 Amazon SNS 訊息至已訂閱您 Amazon SNS 主題的電話號碼。如果您未收到文字訊息，請在 MQTT 用戶端中檢查是否有任何發佈至 `lambda/error` 的訊息。

使用 Lambda 動作疑難排解規則

若您的 Lambda 函數已被呼叫，但您並沒有到文字訊息，請務必確認您的電話號碼已訂閱 Amazon SNS 主題。如果您的電話號碼已訂閱，請檢查您 Lambda 函數的 CloudWatch 記錄。AWS Lambda 會將記錄寫入至 CloudWatch，以便您查看 Lambda 函數的輸出。

檢視 CloudWatch 日誌

1. 在 Lambda 主控台的導覽窗格中選擇 Functions (函數)。

2. 選擇您的 Lambda 函數。

3. 在 Lambda 函數詳細資訊頁面上選擇 Monitoring (監控) 標籤。

4. 選擇 View logs in CloudWatch (檢視 CloudWatch 中的日誌)。

5. 選擇最新的記錄串流。

6. 記錄串流會顯示由您的 Lambda 函數所寫入的記錄。

建立 Amazon SNS 規則

您可以定義規則將訊息資料傳送至 Amazon SNS 主題。

在此教學中，您會建立一個規則，將觸發規則的 AWS IoT 實物名稱傳送給 Amazon SNS 主題的所有訂閱者。

建立具 SNS 動作的規則

1. 在 AWS IoT 主控台的導覽窗格中，選擇 Act (動作)。

2. 在 Rules (規則) 頁面，選擇 Create (建立)。

3. 輸入規則的名稱與簡短描述。

   Note

   我們不建議在規則名稱或描述中使用個人識別資訊。

4. 在 Rule query statement (規則查詢陳述式) 編輯器中，輸入以下內容：

```
SELECT *, topic(3) as thing FROM '$aws/things/+/shadow/update/accepted'
```

(接在 "FROM" 之後的主題篩選條件會指定主題在訊息發佈時觸發規則的動作。主題篩選條件使用的加號 (+) 是萬用字元，可與任何實物名稱配對。接在 "SELECT" 之後的 "topic(3)" 屬性會附加實物的名稱，其為訊息內容的第三個主題欄位。)

5. 請在 Set one or more actions (設定一個或多個動作) 中選擇 Add action (新增動作)。

6. 在 Select an action (選取動作) 下方，選擇 Send a message as an SNS push notification (以 SNS 推播通知形式傳送訊息)，並選擇 Configure action (設定動作)。

Select an action
Select an action.
Insert a message into a DynamoDB table
DYNAMODB
Split message into multiple columns of a DynamoDB table (DynamoDBv2)
DYNAMODBV2
Send a message to a Lambda function
LAMBDA
Send a message as an SNS push notification
SNS
Send a message to an SQS queue
SQS
Send a message to an Amazon Kinesis Stream
AMAZON KINESIS
Republish a message to an AWS IoT topic
AWS IOT REPUBLISH
Store a message in an Amazon S3 bucket
S3
Send a message to an Amazon Kinesis Firehose stream
AMAZON KINESIS FIREHOSE
Send message data to CloudWatch
CLOUDWATCH METRICS
Change the state of a CloudWatch alarm
CLOUDWATCH ALARMS
Send a message to the Amazon Elasticsearch Service
AMAZON ELASTICSEARCH
Send a message to a Salesforce IoT Input Stream
SALESFORCE IOT
Send a message to IoT Analytics
IOT ANALYTICS
Send a message to an IoT Events Input
IOT EVENTS
Start a Step Functions state machine execution
STEP FUNCTIONS
Cancel
Configure action

7. 在 Configure action (設定動作) 頁面上，針對 SNS target (SNS 目標)，選擇 Create (建立)。

8. 在對話方塊中輸入主題名稱，然後選擇 Create (建立)。

9. 在 Configure action (設定動作) 頁面上，針對 SNS target (SNS 目標)，選擇您剛建立的 SNS 主題。在 Message format (訊息格式) 部分，選擇 RAW。在 Choose or create a role to grant AWS IoT access to perform this action (選擇或建立角色，授與 AWS IoT 存取權限以執行此動作) 下，選擇 Create Role (建立角色)。

10. 輸入角色名稱，然後選擇 Create role (建立角色)。

11. 在 Configure action (設定動作) 中，選擇 Add action (新增動作)。

12. 選擇 Create rule (建立規則)。

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

MySNSRule

Description

A more complex SNS rule.

Rule query statement

Indicate the source of the messages you want to process with this rule.

Using SQL version

2016-03-23

Rule query statement

SELECT <Attribute> FROM <Topic Filter> WHERE <Condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature > 50. To learn more, see AWS IoT SQL Reference.

```
1 SELECT *, topic(3) as thing FROM '$aws/things/+/shadow/update/accepted'
```

Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required)

Send a message as an SNS push notification
MySNSTopic
Remove Edit

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

Add action

Tags

Apply tags to your resources to help organize and identify them. A tag consists of a case-sensitive key-value pair. Learn more about tagging your AWS resources.

Tag name | **Value**

Provide a tag name, e.g. Manufacturer | Provide a tag value, e.g. Acme-Corporation | Clear

Add another

Cancel | Create rule

若要測試規則，請對您所建立的 SNS 主題新增訂閱，然後更新所有 AWS IoT 實物的影子。

您可以使用 AWS IoT 主控台尋找物件、開啟其詳細資訊頁面，以及變更裝置的影子。當裝置影子服務收到變更的通知時，會將訊息發佈至 `$aws/things/MySNSThing/shadow/update/accepted`。您的規則會被觸發，而您的 SNS 主題訂閱者都會收到包含物件名稱的訊息。

在 Raspberry Pi 上使用 AWS IoT SDK

AWS IoT 裝置軟體開發套件可協助您輕鬆快速地將裝置連線至 AWS IoT。AWS IoT 裝置 SDK 包含開放原始碼程式庫、開發人員指南與範例以及移植指南，因此您能夠在自選硬體平台上，建置創新的 IoT 產品或解決方案。

> **Important**
>
> 開始本教學課程之前，請先完成AWS IoT 入門 (p. 4)中的步驟。

這些教學提供逐步指示，讓您使用 AWS IoT Device SDK for Embedded C 和 AWS IoT Device SDK for JavaScript，將 Raspberry Pi 連線至 AWS IoT 的訊息中介裝置 (p. 242)。完成這些教學課程中的步驟後，您可以連接到 AWS IoT 平台並執行 AWS IoT 裝置 SDK 隨附的範例應用程式。

內容

- 先決條件 (p. 91)
- 建立 Raspberry Pi 的 AWS IoT 實物 (p. 91)
- SDK 範例組態 (p. 92)
- 使用適用於 Embedded C 的 AWS IoT 裝置 SDK (p. 92)
- 使用適用於 JavaScript 的 AWS IoT 裝置 SDK 和節點 (p. 94)

先決條件

為了完成本教學，您需要以下項目：

- Raspberry Pi 2 代 B 型或更新的型號。
- Raspbian Wheezy 或更新版本。我們建議使用最新版本的 Raspbian，這可從 Raspberry Pi 網站取得。
- 乙太網路或 Wi-Fi 連線。
- 一個 AWS 帳戶。如果您還沒有 AWS 帳戶，您可以透過前往 Amazon AWS 入門資源中心來免費獲得。

建立 Raspberry Pi 的 AWS IoT 實物

物件代表其狀態存放在 AWS 雲端的裝置。裝置的狀態會存放在 JSON 文件中，亦即我們所謂的裝置影子。影子是用於存放和擷取狀態資訊。裝置影子服務會為每個在 AWS IoT 中註冊的裝置，維護一個影子。

向 AWS IoT 註冊您的 Raspberry Pi

1. 在您的 Raspberry Pi 中，瀏覽到 AWS IoT 主控台。
2. 請在導覽窗格中選擇 Secure (安全)，然後選擇 Policies (政策)。
3. 在 Policies (政策) 頁面上，選擇 Create a policy (建立政策)。
4. 在 Create policy (建立政策) 頁面上：
   a. 輸入政策的名稱 (例如 **`RaspberryPi-Policy`**)。
   b. 針對 Action (動作)，輸入 **`iot:*`**。

c. 針對 Resource ARN (資源 ARN)，輸入 `*`。
d. 在 Effect (效果) 下方，選擇 Allow (允許)，然後選擇 Create (建立)。

此政策可允許您的 Raspberry Pi 發佈訊息至 AWS IoT。

> **Important**
>
> 這些設定太過寬鬆。在生產環境中，將許可範圍縮小至您裝置所需的範圍。如需更多詳細資訊，請參閱 授權 (p. 145)。

5. 在主控台的導覽窗格中，選擇 Manage (管理)，然後選擇 Things (實物)。
6. 選擇 Create (建立)。
7. 在 Creating AWS IoT things (建立 IoT 物件) 頁面上，選擇 Create a single thing (建立單一物件)。
8. 在 Add your device to the thing registry (將您的裝置新增至實物登錄) 頁面中，輸入 **`RaspberryPi`** 然後選擇 Next (下一步)。
9. 在 Add a certificate for your thing (新增實物的憑證) 頁面上，選擇 Create certificate (建立憑證)。
10. 在 Certificate created (憑證已建立) 頁面，下載您AWS IoT 的私有金鑰、憑證和根憑證授權單位 (CA)。(為每個項目選擇 Download (下載) 連結。) 這些檔案會儲存在您的 `/home/pi/Downloads` 目錄中。
11. 選擇 Activate (啟用) 來啟用 X.509 憑證，然後選擇 Attach a policy (連接政策)。
12. 在 Add a policy for your thing (新增您的實物的政策) 頁面上，選擇 RaspberryPi-policy，然後選擇 Register Thing (註冊實物)。

SDK 範例組態

使用適用於 Embedded C 的 AWS IoT 裝置 SDK

本節說明如何執行適用於 Embedded C 的 AWS IoT 裝置 SDK。

安裝適用於 Embedded C 的 AWS IoT 裝置 SDK

從 GitHub 將 AWS IoT SDK for Embedded C 下載至 Raspberry Pi：

```
git clone https://github.com/aws/aws-iot-device-sdk-embedded-c.git -b release
```

這會在目前目錄中建立名為 `aws-iot-device-sdk-embedded-c` 的目錄。

從 mbed TLS 網站將 mbed TLS 下載到您的 Raspberry Pi。

cd 到 `/home/pi/Downloads` 目錄中。使用 tar -xvf mbedtls-2.16.3-apache.tgz 展開 `mbedtls-2.16.3-apache.tgz` 檔案。

使用以下命令將 `mbedtls-2.16.3` 的內容複製到 `aws-iot-device-sdk-embedded-C/external_libs/mbedTLS` 目錄中：

```
mv ~/Downloads/mbedtls-2.16.3 ~/aws-iot-device-sdk-embedded-c/external_libs/mbedTLS
```

範例應用程式組態

AWS IoT SDK for Embedded C 包含您可試用的範例應用程式。為求簡化，本教學課程使用 `subscribe_publish_sample` 應用程式，說明如何連線到 AWS IoT 訊息中介裝置，以及訂閱和發佈到 MQTT 主題。

1. 將您在建立 Raspberry Pi 的 AWS IoT 實物 (p. 91)中建立的憑證、私有金鑰和根 CA 憑證複製到 `aws-iot-device-sdk-embedded-C/certs` 目錄中。

   Note

   裝置和根 CA 憑證會有過期或遭撤銷的可能。如果您的憑證過期或遭到撤銷，您必須將新的 CA 憑證或私有金鑰和裝置憑證複製到裝置上。

2. 您必須使用個人 AWS IoT 端點、私有金鑰、憑證和根 CA 憑證來設定範例。請前往 `aws-iot-device-sdk-embedded-c/samples/linux/subscribe_publish_sample` 目錄。

   如果已安裝 AWS CLI，則可以使用 aws iot describe-endpoint --endpoint-type iot:Data-ATS 命令來尋找個人端點 URL。如果尚未安裝 AWS CLI，請開啟 AWS IoT 主控台。在導覽窗格中，選擇 Manage (管理)，然後選擇 Things (實物)。為您的 Raspberry Pi 選擇 IoT 物件，然後選擇 Interact (互動)。您的端點會顯示在物件詳細資料頁面的 HTTPS 區段中。

3. 開啟 `aws_iot_config.h`檔案，並在 `Get from console` 部分更新下列值：

   AWS_IOT_MQTT_HOST

   您的個人端點。

   AWS_IOT_MY_THING_NAME

   您的物件名稱。

   AWS_IOT_ROOT_CA_FILENAME

   您的根 CA 憑證。

   AWS_IOT_CERTIFICATE_FILENAME

   您的憑證。

   AWS_IOT_PRIVATE_KEY_FILENAME

   您的私有金鑰。

   例如：

```
// Get from console
// ================================================
#define AWS_IOT_MQTT_HOST              "a22j5sm6o3yzc5.iot.us-east-1.amazonaws.com"
#define AWS_IOT_MQTT_PORT              8883
#define AWS_IOT_MQTT_CLIENT_ID         "MyRaspberryPi"
#define AWS_IOT_MY_THING_NAME          "MyRaspberryPi"
#define AWS_IOT_ROOT_CA_FILENAME       "root-CA.crt"
#define AWS_IOT_CERTIFICATE_FILENAME   "device.pem.crt"
#define AWS_IOT_PRIVATE_KEY_FILENAME   "private.pem.key"
// ================================================
```

執行範例應用程式

執行 AWS IoT Device SDK for Embedded C 範例應用程式

1. 使用隨附的 makefile 來編譯 `subscribe_publish_sample_app`。

   ```
   make -f Makefile
   ```

   將產生執行檔。

2. 執行 `subscribe_publish_sample_app`。您應該會看到類似下列的輸出：

現已透過 AWS IoT Device SDK for Embedded C 將您的 Raspberry Pi 連接至 AWS IoT。

使用適用於 JavaScript 的 AWS IoT 裝置 SDK 和節點

此教學課程將示範如何在 Raspberry Pi 上安裝 Node.js、npm 套件管理員，和適用於 JavaScript 的 AWS IoT 裝置開發套件，以及如何執行範本應用程式。

安裝最新版本的 Node.js

若要使用適用於 JavaScript 的 AWS IoT 裝置開發套件，請在您的 Raspberry Pi 上安裝 Node.js 和 npm 套件管理員。

1. 若要下載最新版本的 Node 儲存庫，請開啟終端機視窗並執行下列命令：

   curl -sL https://deb.nodesource.com/setup_12.x | sudo -E bash -
2. 請執行以下指令以安裝節點與 npm。

   sudo apt-get install -y nodejs
3. 請執行以下指令以驗證節點與 npm 的安裝。

   node -v

   以及

   npm -v

   如果顯示版本編號，表示節點和 npm 安裝正確。

安裝適用於 JavaScript 的 AWS IoT 裝置 SDK

在您的 Raspberry Pi 上安裝適用於 JavaScript 的 AWS IoT 裝置 SDK。

1. 在終端機視窗中，使用下列命令，將適用於 JavaScript 存放庫的 AWS IoT 裝置 SDK 複製到您的主目錄 (依預設為 `/home/pi`)。

   ```
   git clone https://github.com/aws/aws-iot-device-sdk-js.git
   ```

2. cd 到 `aws-iot-device-sdk-js` 目錄中。
3. 使用 npm 以安裝軟體開發套件：

   ```
   npm install
   ```

準備執行範例應用程式

在 `aws-iot-device-sdk-js` 下方，建立 `certs` 目錄並將您的私有金鑰、裝置憑證和根 CA 憑證複製到其中。

如要執行適用於 JavaScript 的 AWS IoT 裝置開發套件範本，您需要以下資訊：

您的 AWS 區域

若要尋找您正在使用的區域，請開啟 AWS IoT 主控台並查看 URL。區域緊跟在 `https://` 的後方。例如：

```
https://us-west-2.console.aws.amazon.com/iot/home?region=us-west-2#/dashboard
```

如需詳細資訊，請參閱 AWS IoT 服務端點。

用戶端 ID

用於識別連接至 AWS IoT 的裝置或應用程式的任意英數字串。這應該與本教學課程中您的 IoT 物件名稱一樣。

您的私密金鑰

您 Raspberry Pi 的私有金鑰完整路徑 (例如 `/home/pi/aws-iot-device-sdk-js/certs/private.pem.key`)。

您的 AWS IoT X.509 憑證

您 Raspberry Pi 的 AWS IoT 憑證完整路徑 (例如 `/home/pi/aws-iot-device-sdk-js/certs/device.pem.crt`)。

STS Amazon 根 CA

您 Raspberry Pi 的 Amazon 根 CA 完整路徑 (例如 `/home/pi/aws-iot-device-sdk-js/certs/Amazon-root-CA-1.pem`)。

您的 AWS IoT 端點

如果您已在 Raspberry Pi 上安裝 AWS CLI，則可以執行 describe-endpoint CLI 命令來尋找您的端點：

```
aws iot describe-endpoint --endpoint-type iot:Data-ATS
```

如果尚未安裝 AWS CLI，請開啟 AWS IoT 主控台。在導覽窗格中，選擇 Manage (管理)，然後選擇 Things (實物)。為您的 Raspberry Pi 選擇 IoT 物件，然後選擇 Interact (互動)。您的端點會顯示在物件詳細資料頁面的 HTTPS 區段中。

AWS IoT 訊息中介裝置監聽的連接埠

一律為 `8883`。

您的 IoT 物件名稱

此為您在向 AWS IoT 註冊 Raspberry Pi 所使用的名稱。

執行範例應用程式

適用於 JavaScript 的 AWS IoT 裝置開發套件在 `aws-iot-device-sdk-js/examples` 目錄中包含了一些範本。我們建議您從 `device-example.js` 開始。此範例將以兩種模式執行。在模式 1 中，它將訂閱 MQTT 主題 `topic_1`，並在 `topic_2` 上每 4 秒發佈一次訊息。在模式 2 中，它將訂閱 `topic_2`，並在 `topic_1` 上每 4 秒發佈一次訊息。您可執行兩個 `device-example.js` 的執行個體，一個在模式 1，另一個在模式 2，並查看傳送和接收的訊息。

從 `aws-iot-device-sdk-js/examples` 目錄中，執行以下指令以啟動範本的執行個體：

node device-example -k "../certs/private.pem.key" -c "../certs/device.pem.crt" -i "***raspberry-pi-1***" -a "../certs/Amazon-root-CA-1" -H "***<your-iot-endpoint>***" -p 8883 -T "***your-thing-name***" --test-mode 1

啟動在模式 2 中執行的另一個 `device-example.js` 執行個體：

node device-example -k "../certs/private.pem.key" -c "../certs/device.pem.crt" -i "***raspberry-pi-2***" -a "../certs/Amazon-root-CA-1" -H "***<your-iot-endpoint>***" -p 8883 -T "***your-thing-name***" --test-mode 2

> **Important**
>
> 在您執行兩個 `device-example.js` 的執行個體時，請務必確認您使用了不同的用戶端 ID。兩個不同的用戶端 (裝置或應用程式) 不得使用相同的用戶端 ID 來連接至 AWS IoT。第一個用戶端的連線已終止，而第二個用戶端的連線已建立。
> 只有在您建立 IoT 物件特定的政策時，物件名稱才特別重要。在 AWS IoT 入門教學課程中，您不會建立這種政策，因此您可以在兩個執行個體都使用相同的物件名稱。

您的 Raspberry Pi 現已透過適用於 JavaScript 的 AWS IoT 裝置 SDK 連接至 AWS IoT。

如果範本執行個體正確執行，則在模式 1 中執行的執行個體輸出應如下所示：

```
substituting 250ms dely for truems...
connect
message topic_1 {"mode1Process":1}
message topic_1 {"mode1Process":2}
message topic_1 {"mode1Process":3}
message topic_1 {"mode1Process":4}
...
```

在模式 2 中執行的執行個體輸出應如下所示：

```
substituting 250ms dely for truems...
connect
message topic_2 {"mode2Process":1}
message topic_2 {"mode2Process":2}
message topic_2 {"mode2Process":3}
message topic_2 {"mode2Process":4}
...
```

如果範例無法正確執行，請嘗試新增 `-d` 選項以顯示偵錯資訊。

AWS IoT 其他教學課程

本節中的教學課程為您介紹如何同時使用多個 AWS IoT 服務來達成任務。這些教學課程專注於服務的整合，而不是完整的逐步解說 AWS IoT 功能。本節中的教學課程可能彼此為基礎建置，來為您介紹如何從較小的服務開始，再根據您公司需求的變更或增長發展解決方案。

每個教學課程包含先決條件清單，包括特定的硬體。可能的話，教學課程會在您沒有所有必要的硬體時提供替代方案。

使用 AWS IoT 和 Raspberry Pi 監控土壤濕度

本教學課程說明如何使用 Raspberry Pi、濕度感應器和 AWS IoT 來監控室內盆栽植物或花園的土壤濕度等級。Raspberry Pi 執行的程式碼，會從感應器讀取濕度等級和溫度，然後將資料傳送到 AWS IoT。您會在 AWS IoT 中建立當濕度等級低於閾值時，會傳送電子郵件到訂閱 Amazon SNS 主題地址的規則。

內容

- 先決條件 (p. 97)
- 設定 AWS IoT (p. 97)
  - 建立 AWS IoT 政策 (p. 98)
  - 建立 AWS IoT 實物、憑證和私有金鑰 (p. 99)
  - 建立 Amazon SNS 主題和訂閱 (p. 99)
  - 建立 AWS IoT 規則來傳送電子郵件 (p. 100)
- 設定您的 Raspberry Pi 和濕度感應器 (p. 101)

先決條件

為完成此教學課程您需要：

- 一個 AWS 帳戶。
- 擁有管理員許可的 IAM 使用者。
- 執行 Windows、macOS、Linux 或 Unix 的開發電腦，用來存取 AWS IoT 主控台。
- 執行 Raspbian Buster 的 Raspberry Pi 3B 或 4B。如需安裝指示，請參閱 Rasberry Pi 網站上的 Installing operating system images (安裝作業系統映像)。
- 用於 Raspberry Pi 的顯示器、鍵盤、滑鼠和 Wi-Fi 網路或乙太網路連線。
- 與 Raspberry Pi 相容的濕度感應器。本教學課程中使用的感應器是 Adafure STEMMA I2C 電容式濕度感應器搭配 JST 4 針腳轉母頭插槽纜線接頭。

設定 AWS IoT

若要完成此教學課程，您需要建立以下資源。若要將裝置連線至 AWS IoT，請建立 IoT 實物、裝置憑證和 AWS IoT 政策。

- AWS IoT 實物。

  代表實體裝置 (在此案例中為 Rasberry Pi)，包含有關裝置的靜態中繼資料的實物。
- 裝置憑證。

  所有裝置都必須擁有裝置憑證，才能連接至並向 AWS IoT 驗證。
- AWS IoT 政策。

每個裝置憑證都有與其相關聯的一或多個 AWS IoT 政策。這些政策決定裝置可存取哪些 AWS IoT 資源。

- AWS IoT 根 CA 憑證。

  裝置和其他用戶端會使用 AWS IoT 根 CA 憑證來驗證與其通訊的 AWS IoT 伺服器。如需詳細資訊，請參閱伺服器身分驗證。

- AWS IoT 規則。

  規則包含查詢和一或多個規則動作。查詢會從裝置訊息擷取資料，以判斷是否應處理訊息資料。規則動作會指定當資料符合查詢時要採取的動作。

- Amazon SNS 主題和主題訂閱。

  此規則會監聽來自 Raspberry Pi 的濕度資料。如果值低於閾值，它會傳送訊息到 Amazon SNS 主題。Amazon SNS 會將該訊息傳送到訂閱該主題的所有電子郵件地址。

建立 AWS IoT 政策

建立會允許您的 Raspberry Pi 連接和傳送訊息至 AWS IoT 的 AWS IoT 政策。

1. 在 AWS IoT 主控台中，如果 Get started (開始) 按鈕出現，請選擇它。否則，請在導覽窗格中展開 Secure (安全)，然後選擇 Policies (政策)。
2. 如果 You don't have any policies yet (您尚未有任何政策) 對話方塊出現，請選擇 Create a policy (建立政策)。否則，請選擇 Create (建立)。
3. 輸入 AWS IoT 政策的名稱 (例如，**MoistureSensorPolicy**)。
4. 在 Add statements (新增陳述式) 區段中，將現有政策取代為下列 JSON。將 *<region>* 和 *<account>* 取代為您 AWS 的區域和 AWS 帳戶號碼。

```
{
   "Version": "2012-10-17",
   "Statement": [{
         "Effect": "Allow",
         "Action": "iot:Connect",
         "Resource": "arn:aws:iot:<region>:<account>:client/RaspberryPi"
      },
      {
         "Effect": "Allow",
         "Action": "iot:Publish",
         "Resource": [
            "arn:aws:iot:us-west-2:504350838278:topic/$aws/things/RaspberryPi/shadow/
update",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
delete",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/get"
         ]
      },
      {
         "Effect": "Allow",
         "Action": "iot:Receive",
         "Resource": [
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
update/accepted",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
delete/accepted",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/get/
accepted",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
update/rejected",
            "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
delete/rejected"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iot:Subscribe",
            "Resource": [
                "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
update/accepted",
                "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
delete/accepted",
                "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/get/
accepted",
                "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
update/rejected",
                "arn:aws:iot:<region>:<account>:topic/$aws/things/RaspberryPi/shadow/
delete/rejected"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:GetThingShadow",
                "iot:UpdateThingShadow",
                "iot:DeleteThingShadow"
            ],
            "Resource": "arn:aws:iot:<region>:<account>:thing/RaspberryPi"

        }
    ]
}
```

5. 選擇 Create (建立)。

建立 AWS IoT 實物、憑證和私有金鑰

在 AWS IoT 登錄中建立實物來代表您的 Raspberry Pi。

1. 在 AWS IoT 主控台的導覽窗格中，選擇 Manage (管理)，然後選擇 Things (實物)。
2. 如果顯示 You don't have any things yet (您尚未有任何實物) 對話方塊，請選擇 Register a thing (註冊實物)。否則，請選擇 Create (建立)。
3. 在 Creating AWS IoT things (建立 IoT 物件) 頁面上，選擇 Create a single thing (建立單一物件)。
4. 在 Add your device to the device registry (將您的裝置新增至裝置登錄) 頁面上，輸入您 IoT 實物的名稱 (例如 **RaspberryPi**)，然後選擇 Next (下一步)。
5. 在 Add a certificate for your thing (新增實物的憑證) 頁面上，選擇 Create certificate (建立憑證)。
6. 選擇 Download (下載) 連結來下載憑證、私有金鑰和根 CA 憑證。

   Important

   這是您可以下載憑證和私有金鑰唯一機會。
7. 選擇 Activate (啟用)。
8. 選擇 Attach a policy (連接政策)。
9. 針對 Add a policy for your thing (新增您的實物的政策)，選擇 MoistureSensorPolicy，然後選擇 Register Thing (註冊實物)。

建立 Amazon SNS 主題和訂閱

建立 Amazon SNS 主題和訂閱。

1. 從 AWS IoT 主控台，選擇 Services (服務)，輸入 **SNS**，然後選擇 Simple Notification Service (簡單通知服務)。
2. 在導覽窗格中選擇 Topics (主題)，然後選擇 Create topic (建立主題)。
3. 輸入主題的名稱 (例如 **MoistureSensorTopic**)。
4. 輸入主題的顯示名稱 (例如，**Moisture Sensor Topic**)。這是在 Amazon SNS 主控台中針對您的主題顯示的名稱。
5. 選擇 Create topic (建立主題)。
6. 在 Amazon SNS 主題詳細資訊頁面中，選擇 Create Subscription (建立訂閱)。
7. 對於 通訊協定 ，選擇 電子郵件 。
8. 針對 Endpoint (端點)，輸入您的電子郵件位址。
9. 選擇 Create subscription (建立訂閱)。
10. 開啟您的電子郵件用戶端，並尋找主旨為 **MoistureSensorTopic** 的訊息。開啟電子郵件，然後按一下 Confirm subscription (確認訂閱) 連結。

    Important

    在您確認訂閱之前，不會收到來自此 Amazon SNS 主題的任何電子郵件提醒。

您應該會收到含有您輸入文字的電子郵件訊息。

建立 AWS IoT 規則來傳送電子郵件

AWS IoT 規則可定義從裝置接收訊息時要採取的查詢和一或多個動作。AWS IoT 規則引擎會監聽裝置傳送的訊息，並使用訊息中的資料來判斷是否應該採取某些動作。如需詳細資訊，請參閱 AWS IoT 的規則 (p. 254)。

在此教學中，您的 Raspberry Pi 會將訊息發佈至 `aws/things/RaspberryPi/shadow/update`。這是內部的 MQTT 主題，供裝置和實物影子服務使用。Raspberry Pi 發佈的訊息會具備以下格式：

```
{
    "reported": {
        "moisture" : <moisture-reading>,
        "temp" : <temperature-reading>
    }
}
```

您會建立一個查詢，從傳入的訊息擷取濕度和溫度資料。您也可以建立 Amazon SNS 動作，在濕度讀數低於閾值時，取得資料並將其傳送給 Amazon SNS 主題訂閱者。

建立 Amazon SNS 規則

1. 在 AWS IoT 主控台的導覽窗格中，選擇 Act (動作)。如果 You don't have any rules yet (您尚未有任何規則) 對話方塊出現，請選擇 Create a rule (建立規則)。否則，請選擇 Create (建立)。
2. 在 Create a rule (建立規則) 頁面中，輸入規則的名稱 (例如 **MoistureSensorRule**)。
3. 針對 Description (描述)，為此規則提供簡短的描述，例如，**Sends an alert when soil moisture level readings are too low**。
4. 在 Rule query statement (規則查詢陳述式) 下，選擇 SQL 版本 2016-03-23，然後輸入下列 AWS IoT SQL 查詢陳述式：

```
SELECT * FROM '$aws/things/RaspberryPi/shadow/update/accepted' WHERE
 state.reported.moisture < 400
```

此陳述式會在 `moisture` 讀數小於 `400` 時觸發規則動作。

> **Note**
>
> 您可能需要使用不同的值。在您的 Raspberry Pi 上執行程式碼之後，您可以碰觸感應器、將其放入水中，或將其放入花盆中，以查看從感應器取得的值。

5. 在 Set one or more actions (設定一或多個動作) 下，選擇 Add action (新增動作)。
6. 在 Select an action (選取動作) 頁面上，選擇 Send a message as an SNS push notification (傳送 SNS 推送通知形式的訊息)。
7. 向下捲動到頁面底部，然後選擇 Configure action (設定動作)。
8. 在 Configure action (設定動作) 頁面上，針對 SNS target (SNS 目標)，選擇 Select (選取)，然後選擇 LowMoistureTopic。
9. 在 Message format (訊息格式) 部分，選擇 RAW。
10. 在 Choose or create a role to grant AWS IoT access to perform this action (選擇或建立角色，授與 AWS IoT 存取權限以執行此動作) 下，選擇 Create role (建立角色)。輸入角色的名稱 (例如 **`LowMoistureTopicRole`**)，然後選擇 Create role (建立角色)。
11. 選擇 Add action (新增動作)。
12. 選擇 Create rule (建立規則)。

設定您的 Raspberry Pi 和濕度感應器

將您的 Micro SD 卡插入到 Raspberry Pi，連接螢幕、鍵盤、滑鼠，以及乙太網路纜線 (如果您未使用 Wi-Fi)。還不要連接電源線。

將 JST 跳線纜線連接至濕度感應器。跳線的另一端有四條配線：

- 綠色：I2C SCL
- 白色：I2C SDA
- 紅色：功率 (3.5 V)
- 黑色：接地

按住 Raspberry Pi 與右側的乙太網路插孔。在此方向中，上方有兩列 GPIO 接腳。按照以下順序，將濕度感應器的配線連接到接腳的下方排。從最左側接腳開始，連接紅色 (電源)、白色 (SDA) 和綠色 (SCL)。略過一個接腳，然後連接黑色 (接地) 配線。如需詳細資訊，請參閱 Python Computer Wiring。

將電源線連接至 Raspberry Pi，並將另一端插入牆上插座，以將其開啟。

設定您的 Raspberry Pi

1. 在 Welcome to Raspberry Pi (歡迎使用 Raspberry Pi) 上，選擇 Next (下一步)。
2. 選擇您的國家/地區、語言、時區及鍵盤配置。選擇 Next (下一步)。
3. 輸入您的 Raspberry Pi 的密碼，然後選擇 Next (下一步)。
4. 選擇您的 Wi-Fi 網路，然後選擇 Next (下一步)。如果您不是使用 Wi-Fi 網路，請選擇 Skip (略過)。
5. 選擇 Next (下一步) 以檢查軟體更新。更新完成時，選擇 Restart (重新啟動) 以重新啟動您的 Raspberry Pi。

在您的 Raspberry Pi 啟動後，啟用 I2C 界面。

1. 在 Raspbian 桌面的左上角，按一下 Raspberry 圖示，選擇 Preferences (偏好設定)，然後選擇 Raspberry Pi Configuration (Raspberry Pi 組態)。
2. 在 Interfaces (界面) 標籤上，針對 I2C，選擇 Enable (啟用)。

3. 選擇 OK (確定)。

Adafruit STEMMA 濕度感應器的程式庫是針對 CircuitPython 所撰寫。若要在 Raspberry Pi 上執行它們，您需要安裝最新版本的 Python 3。

1. 從命令提示執行下列命令，以更新您的 Raspberry Pi 軟體：

   ```
   sudo apt-get update
   ```

   ```
   sudo apt-get upgrade
   ```

2. 執行以下命令來更新您的 Python 3 安裝：

   ```
   sudo pip3 install --upgrade setuptools
   ```

3. 執行以下命令來安裝 Raspberry Pi GPIO 程式庫：

   ```
   pip3 install RPI.GPIO
   ```

4. 執行以下命令來安裝 Adafruit Blinka 程式庫：

   ```
   pip3 install adafruit-blinka
   ```

   如需詳細資訊，請參閱 在 Raspberry Pi 上安裝 CircuitPython 程式庫。

5. 執行以下命令來安裝 Adafruit Seesaw 程式庫：

   ```
   sudo pip3 install adafruit-circuitpython-seesaw
   ```

6. 執行以下命令來安裝適用於 Python 的 AWS IoT 裝置軟體開發套件：

   ```
   pip3 install AWSIoTPythonSDK
   ```

您的 Raspberry Pi 現在擁有所有必要的程式庫。建立名為 **moistureSensor.py** 的檔案，並將下列 Python 程式碼複製到檔案：

```
from adafruit_seesaw.seesaw import Seesaw
from AWSIoTPythonSDK.MQTTLib import AWSIoTMQTTShadowClient
from board import SCL, SDA

import logging
import time
import json
import argparse
import busio

# Shadow JSON schema:
#
# {
#   "state": {
#       "desired":{
#           "moisture":<INT VALUE>,
#           "temp":<INT VALUE>
#       }
#   }
# }

# Function called when a shadow is updated
def customShadowCallback_Update(payload, responseStatus, token):

    # Display status and data from update request
    if responseStatus == "timeout":
        print("Update request " + token + " time out!")
```

```
    if responseStatus == "accepted":
        payloadDict = json.loads(payload)
        print("~~~~~~~~~~~~~~~~~~~~~~~~")
        print("Update request with token: " + token + " accepted!")
        print("moisture: " + str(payloadDict["state"]["reported"]["moisture"]))
        print("temperature: " + str(payloadDict["state"]["reported"]["temp"]))
        print("~~~~~~~~~~~~~~~~~~~~~~~~\n\n")

    if responseStatus == "rejected":
        print("Update request " + token + " rejected!")

# Function called when a shadow is deleted
def customShadowCallback_Delete(payload, responseStatus, token):

     # Display status and data from delete request
    if responseStatus == "timeout":
        print("Delete request " + token + " time out!")

    if responseStatus == "accepted":
        print("~~~~~~~~~~~~~~~~~~~~~~~~")
        print("Delete request with token: " + token + " accepted!")
        print("~~~~~~~~~~~~~~~~~~~~~~~~\n\n")

    if responseStatus == "rejected":
        print("Delete request " + token + " rejected!")


# Read in command-line parameters
def parseArgs():

    parser = argparse.ArgumentParser()
    parser.add_argument("-e", "--endpoint", action="store", required=True, dest="host",
 help="Your AWS IoT custom endpoint")
    parser.add_argument("-r", "--rootCA", action="store", required=True, dest="rootCAPath",
 help="Root CA file path")
    parser.add_argument("-c", "--cert", action="store", dest="certificatePath",
 help="Certificate file path")
    parser.add_argument("-k", "--key", action="store", dest="privateKeyPath", help="Private
 key file path")
    parser.add_argument("-p", "--port", action="store", dest="port", type=int, help="Port
 number override")
    parser.add_argument("-n", "--thingName", action="store", dest="thingName",
 default="Bot", help="Targeted thing name")
    parser.add_argument("-id", "--clientId", action="store", dest="clientId",
 default="basicShadowUpdater", help="Targeted client id")

    args = parser.parse_args()
    return args


# Configure logging
# AWSIoTMQTTShadowClient writes data to the log
def configureLogging():

    logger = logging.getLogger("AWSIoTPythonSDK.core")
    logger.setLevel(logging.DEBUG)
    streamHandler = logging.StreamHandler()
    formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
    streamHandler.setFormatter(formatter)
    logger.addHandler(streamHandler)


# Parse command line arguments
args = parseArgs()
```

```
if not args.certificatePath or not args.privateKeyPath:
    parser.error("Missing credentials for authentication.")
    exit(2)

# If no --port argument is passed, default to 8883
if not args.port:
    args.port = 8883


# Init AWSIoTMQTTShadowClient
myAWSIoTMQTTShadowClient = None
myAWSIoTMQTTShadowClient = AWSIoTMQTTShadowClient(args.clientId)
myAWSIoTMQTTShadowClient.configureEndpoint(args.host, args.port)
myAWSIoTMQTTShadowClient.configureCredentials(args.rootCAPath, args.privateKeyPath,
 args.certificatePath)

# AWSIoTMQTTShadowClient connection configuration
myAWSIoTMQTTShadowClient.configureAutoReconnectBackoffTime(1, 32, 20)
myAWSIoTMQTTShadowClient.configureConnectDisconnectTimeout(10) # 10 sec
myAWSIoTMQTTShadowClient.configureMQTTOperationTimeout(5) # 5 sec

# Initialize Raspberry Pi's I2C interface
i2c_bus = busio.I2C(SCL, SDA)

# Intialize SeeSaw, Adafruit's Circuit Python library
ss = Seesaw(i2c_bus, addr=0x36)

# Connect to AWS IoT
myAWSIoTMQTTShadowClient.connect()

# Create a device shadow handler, use this to update and delete shadow document
deviceShadowHandler = myAWSIoTMQTTShadowClient.createShadowHandlerWithName(args.thingName,
 True)

# Delete curent shadow JSON doc
deviceShadowHandler.shadowDelete(customShadowCallback_Delete, 5)

# Read data from moisture sensor and update shadow
while True:

    # read moisture level through capacitive touch pad
    moistureLevel = ss.moisture_read()

    # read temperature from the temperature sensor
    temp = ss.get_temp()

    # Display moisture and temp readings
    print("Moisture Level: {}".format(moistureLevel))
    print("Temperature: {}".format(temp)

    # Create message payload
    payload = {"state":{"reported":{"moisture":str(moistureLevel),"temp":str(temp)}}}

    # Update shadow
    deviceShadowHandler.shadowUpdate(json.dumps(payload), customShadowCallback_Update, 5)
    time.sleep(1)
```

將檔案儲存到您可以找到的位置。從命令列搭配下列參數執行 `moistureSensor.py`：

endpoint

您的自訂 AWS IoT 端點。

rootCA

您的 AWS IoT 根 CA 憑證的完整路徑。

cert

您的 AWS IoT 裝置憑證的完整路徑。

金鑰

您的 AWS IoT 裝置憑證私有金鑰的完整路徑。

thingName

您的實物名稱 (在此案例中為 `RaspberryPi`)。

clientId

MQTT 用戶端 ID。請使用 `RaspberryPi`。

命令列看起來應該如下：

```
python3 moistureSensor.py --endpoint --rootCA ~/certs/Amazon-Root-CA-1.pem --
cert ~/certs/raspberrypi.crt --key ~/certs/raspberrypi-private.key --thing-name
RaspberryPi --clientId RaspberryPi
```

嘗試碰觸感應器、將其放入花盆中，或將其放入杯水中，以查看感應器如何回應各種濕度。如有需要，您可以在 `MoistureSensorRule` 中變更閾值。當濕度感應器讀數低於規則的 SQL 查詢陳述式中指定的值時，AWS IoT 就會將訊息發佈至 Amazon SNS 主題。您應該會收到包含濕度和溫度資料的電子郵件訊息。

在您驗證收到來自 Amazon SNS 的電子郵件訊息後，請按 CTRL+C 來停止 Python 程式。Python 程式不太可能傳送足夠的訊息因而衍生費用，但最佳實務是在完成時停止程式。

利用 AWS IoT 管理裝置

AWS IoT 會提供登錄檔，協助您管理「物件」。物件是特定裝置或邏輯實體的代表。它可以是實體裝置或感應器 (例如燈泡或牆上的開關)。它也可以是邏輯實體 (例如應用程式的執行個體)，或是未連線至 AWS IoT 但與其他有連線裝置有關聯的物理實體 (例如具有引擎感應器或控制面板的汽車)。

物件的相關資訊會以 JSON 資料的形式存放於登錄檔中。以下為物件的範例：

```
{
     "version": 3,
    "thingName": "MyLightBulb",
    "defaultClientId": "MyLightBulb",
    "thingTypeName": "LightBulb",
    "attributes": {
        "model": "123",
        "wattage": "75"
    }
}
```

物件以名稱識別。物件可以具有屬性，亦即名稱/值對，可讓您用來存放關於物件的資訊，例如其序號或製造商。

典型的裝置使用案例包括使用物件名稱做為預設 MQTT 用戶端 ID。雖然我們並不會強制對應物件登錄檔名稱及其使用的 MQTT 用戶端 ID、憑證或影子狀態，但仍建議您選擇物件名稱，並用其作為登錄檔與 Device Shadow 服務的 MQTT 用戶端 ID。這能為您的 IoT 裝置機群帶來秩序與便利，同時仍能保有基礎裝置憑證模型或影子的彈性。

您不必在登錄檔中建立物件，即可能將裝置連線至 AWS IoT。將物件新增至登錄檔，可更輕鬆地管理及搜尋裝置。

如何使用登錄檔管理物件

您可以使用 AWS IoT 主控台或 AWS CLI 與登錄檔互動。下列章節說明如何利用 CLI 使用登錄檔。

建立物件

以下命令示範如何從 CLI 使用 AWS IoTCreateThing 命令建立物件：

```
$ aws iot create-thing --thing-name "MyLightBulb" --attribute-payload "{\"attributes\":
 {\"wattage\":\"75\", \"model\":\"123\"}}"
```

CreateThing命令會顯示新物件的名稱和 ARN：

```
{
    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyLightBulb",
    "thingName": "MyLightBulb"
    "thingId": "12345678abcdefgh12345678ijklmnop12345678"
}
```

> **Note**
>
> 我們不建議在實物名稱中使用個人識別資訊。

列出物件

您可以使用 ListThings命令列出您帳戶中的所有物件：

```
$ aws iot list-things
{
    "things": [
       {
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1,
            "thingName": "MyLightBulb"
        },
        {
            "attributes": {
                "numOfStates":"3"
             },
            "version": 11,
            "thingName": "MyWallSwitch"
        }
    ]
}
```

搜尋物件

您可以使用 DescribeThing命令列出關於物件的資訊：

```
$ aws iot describe-thing --thing-name "MyLightBulb"
{
    "version": 3,
    "thingName": "MyLightBulb",
    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyLightBulb",
    "thingId": "12345678abcdefgh12345678ijklmnop12345678"
    "defaultClientId": "MyLightBulb",
    "thingTypeName": "StopLight",
    "attributes": {
        "model": "123",
        "wattage": "75"
    }
}
```

您可以使用 ListThings命令搜尋與物件類型名稱相關的所有物件：

```
$  aws iot list-things --thing-type-name "LightBulb"
```

```
{
    "things": [
        {
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1,
            "thingName": "MyRGBLight"
        },
```

```
        {
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1,
            "thingName": "MySecondLightBulb"
        }
    ]
}
```

您可以使用 ListThings命令搜尋屬性為特定值的所有物件：

```
$  aws iot list-things --attribute-name "wattage" --attribute-value "75"
```

```
{
    "things": [
        {
            "thingTypeName": "StopLight",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 3,
            "thingName": "MyLightBulb"
        },
        {
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1,
            "thingName": "MyRGBLight"
        },
        {
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1,
            "thingName": "MySecondLightBulb"
        }
    ]
}
```

更新物件

您可以使用 UpdateThing命令更新物件：

```
$ aws iot update-thing --thing-name "MyLightBulb" --attribute-payload "{\"attributes\":
 {\"wattage\":\"150\", \"model\":\"456\"}}"
```

UpdateThing命令不會產生輸出。您可以使用 DescribeThing命令來查看結果：

```
$ aws iot describe-thing --thing-name "MyLightBulb"
{
```

```
    "attributes": {
        "model": "456",
        "wattage": "150"
    },
    "version": 2,
    "thingName": "MyLightBulb"
}
```

刪除物件

您可以使用 DeleteThing命令來刪除物件：

```
$ aws iot delete-thing --thing-name "MyThing"
```

如果刪除成功或您指定的物件不存在，則此命令會成功傳回且不含錯誤。

將主體連接至物件

實體裝置必須擁有 X.509 憑證才能與 AWS IoT 通訊。您可以將裝置上的憑證與登錄檔中代表您裝置的物件相關聯。若要將憑證連接至物件，請使用 AttachThingPrincipal命令：

```
$ aws iot attach-thing-principal --thing-name "MyLightBulb" --principal "arn:aws:iot:us-
east-1:123456789012:cert/a0c01f5835079de0a7514643d68ef8414ab739a1e94ee4162977b02b12842847"
```

AttachThingPrincipal命令不會產生任何輸出。

將主體與物件分離

您可以使用 DetachThingPrincipal命令從物件分離出憑證：

```
$ aws iot detach-thing-principal --thing-name "MyLightBulb" --principal "arn:aws:iot:us-
east-1:123456789012:cert/a0c01f5835079de0a7514643d68ef8414ab739a1e94ee4162977b02b12842847"
```

DetachThingPrincipal命令不會產生任何輸出。

物件類型

物件類型能讓您存放與相同物件類型關聯之所有物件通用的描述和組態資訊。如此可簡化登錄檔中物件的管理。例如，您可以定義 LightBulb 物件類型。所有與 LightBulb 物件類型關聯的物件具有同一組屬性：序號、製造商、瓦特數。當您建立 LightBulb 類型的物件 (或是變更現有物件的類型為 LightBulb)，可以指定在 LightBulb 物件類型中所定義的每個屬性之值。

雖然物件類型為選用，但使用時可讓您探索物件更輕鬆。

- 有物件類型的物件最多可具有 50 個屬性。
- 無物件類型的物件最多可具有三個屬性。
- 一個物件只能與一個物件類型建立關聯。
- 在您的帳戶中，可以建立的物件類型數量不限。

物件類型不可變。物件類型建立之後，您就無法變更其名稱。您可以隨時棄用物件類型，避免新的物件與其建立關聯。您也可以刪除未和任何物件關聯的物件類型。

建立物件類型

您可以使用 CreateThingType命令來建立物件類型：

```
$ aws iot create-thing-type

                --thing-type-name "LightBulb" --thing-type-properties
 "thingTypeDescription=light bulb type, searchableAttributes=wattage,model"
```

CreateThingType命令的回應會包含物件類型及其 ARN：

```
{
    "thingTypeName": "LightBulb",
    "thingTypeId": "df9c2d8c-894d-46a9-8192-9068d01b2886",
    "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb"
}
```

列出物件類型

您可以使用 ListThingTypes命令列出物件類型：

```
$ aws iot list-thing-types
```

ListThingTypes命令會傳回一份清單，列出在您 AWS 帳戶中所定義的物件類型：

```
{
    "thingTypes": [
        {
            "thingTypeName": "LightBulb",
            "thingTypeProperties": {
                "searchableAttributes": [
                    "wattage",
                    "model"
                ],
                "thingTypeDescription": "light bulb type"
            },
            "thingTypeMetadata": {
                "deprecated": false,
                "creationDate": 1468423800950
            }
        }
    ]
}
```

描述物件類型

您可以使用 DescribeThingType命令獲得關於物件類型的資訊：

```
$ aws iot describe-thing-type --thing-type-name "LightBulb"
```

DescribeThingType 命令會傳回關於指定類型的資訊：

```
{
    "thingTypeProperties": {
        "searchableAttributes": [
```

```
            "model",
            "wattage"
        ],
        "thingTypeDescription": "light bulb type"
    },
    "thingTypeId": "df9c2d8c-894d-46a9-8192-9068d01b2886",
    "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",
    "thingTypeName": "LightBulb",
    "thingTypeMetadata": {
        "deprecated": false,
        "creationDate": 1544466338.399
    }
}
```

將物件類型與物件建立關聯

在建立物件時，您可以使用 CreateThing命令指定物件類型：

```
$ aws iot create-thing --thing-name "MyLightBulb" --thing-type-name "LightBulb" --
attribute-payload "{\"attributes\": {\"wattage\":\"75\", \"model\":\"123\"}}"
```

您可以隨時使用 UpdateThing命令變更與物件關聯的物件類型：

```
$ aws iot update-thing --thing-name "MyLightBulb"
                --thing-type-name "LightBulb" --attribute-payload  "{\"attributes\":
 {\"wattage\":\"75\", \"model\":\"123\"}}"
```

您也可以使用 UpdateThing命令取消物件與物件類型的關聯。

棄用物件類型

物件類型不可變。物件類型定義之後就無法變更。不過，您可以棄用物件類型，避免使用者將其與任何新物件建立關聯。所有與該物件類型關聯的現有物件都不會變更。

若要棄用物件類型，請使用 DeprecateThingType命令：

```
$ aws iot deprecate-thing-type --thing-type-name "myThingType"
```

您可以使用 DescribeThingType命令來查看結果：

```
$ aws iot describe-thing-type --thing-type-name "StopLight":
```

```
{
    "thingTypeName": "StopLight",
    "thingTypeProperties": {
        "searchableAttributes": [
            "wattage",
            "numOfLights",
            "model"
        ],
        "thingTypeDescription": "traffic light type",
    },
    "thingTypeMetadata": {
        "deprecated": true,
        "creationDate": 1468425854308,
        "deprecationDate": 1468446026349
    }
```

```
}
```

棄用物件類型是可逆的操作。您可以使用 `--undo-deprecate`旗標以及 DeprecateThingType CLI 命令來取消棄用：

```
$ aws iot deprecate-thing-type --thing-type-name "myThingType" --undo-deprecate
```

您可以使用 DescribeThingTypeCLI 命令查看結果：

```
$ aws iot describe-thing-type --thing-type-name "StopLight":
```

```
{
    "thingTypeName": "StopLight",
    "thingTypeArn": "arn:aws:iot:us-east-1:123456789012:thingtype/StopLight",
    "thingTypeId": "12345678abcdefgh12345678ijklmnop12345678"
    "thingTypeProperties": {
        "searchableAttributes": [
            "wattage",
            "numOfLights",
            "model"
        ],
        "thingTypeDescription": "traffic light type"
    },
    "thingTypeMetadata": {
        "deprecated": false,
        "creationDate": 1468425854308,
    }
}
```

刪除物件類型

物件類型只有在棄用後才能刪除。若要刪除物件類型，請使用 DeleteThingType命令：

```
$ aws iot delete-thing-type --thing-type-name "StopLight"
```

> **Note**
>
> 棄用物件類型後，您必須等待五分鐘才能夠將其刪除。

物件群組

實物群組可讓您將多個實物分類成群組以方便同時管理。群組還可以包含群組 —，這樣就能建立群組階層。您可以將政策連接至父群組，而此政策會被其子群組以及父、子群組內的所有物件繼承。這可讓您輕易控制大量物件的許可。

您可以對物件群組執行下列操作：

- 建立、描述或刪除群組。
- 新增物件至一個多個群組。
- 從群組移除物件
- 列出您建立的群組。
- 列出群組的所有子群組 (直系與旁系的後代)。
- 列出群組裡的物件，包括其子群組裡的所有物件。

- 列出群組的所有上階群組 (直系與旁系的父群組)。
- 新增、刪除或更新群組的屬性。(屬性是您可用於存放群組相關資訊的名稱/值對。)
- 讓政策與群組連接或分離。
- 列出連接至群組的政策。
- 列出物件所繼承的政策 (透過連接至其群組或其父群組的政策)。
- 設定群組中物件的記錄選項 (請參閱 設定 AWS IoT 日誌記錄 (p. 212))。
- 建立傳送至群組及其子群組中所有物件並執行的工作 (請參閱 任務 (p. 363))。

以下為物件群組的限制：

- 一個群組最多只能有一個直系的父群組。
- 如果某個群組要成為另一個群組的子群組，必須在建立時就先指定。
- 您無法在後來變更群組的父群組。(因此請規劃好您的群組階層，並且先建立父群組，再建立其包含的子群組。)
- 您無法讓一個物件加入超過 10 個群組。
- 在相同的階層內，您無法讓一個物件加入超過一個群組。(亦即，您無法讓物件加入父群組相同的兩個群組。)
- 您無法重新命名群組。
- 實物群組名稱無法包含國際字元，例如 û、é 和 ñ。

讓政策跟群組連接和分離，可以在許多重要方面強化您的 AWS IoT 操作安全。透過裝置將政策連接至憑證再連接到物件的方式相當耗時，而且在裝置機群中要迅速更新或變更政策也很困難。若政策連接至物件的群組，在輪換物件的憑證時就可以省卻許多步驟。此外，物件變更群組成員資格時，政策就會動態套用至物件，所以每當群組中有裝置變更成員資格，您就不必重新建立一組複雜的許可。

建立物件群組

您可以使用 CreateThingGroup命令來建立物件群組：

```
$ aws iot create-thing-group --thing-group-name LightBulbs
```

CreateThingGroup 命令傳回的回應會包含物件群組、其 ID 與 ARN：

```
{
    "thingGroupName": "LightBulbs",
    "thingGroupId": "abcdefgh12345678ijklmnop12345678qrstuvwx",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
}
```

> **Note**
>
> 我們不建議在實物群組名稱中使用個人識別資訊。

這個範例是在建立物件群組時，為其指定一個父群組：

```
$ aws iot create-thing-group --thing-group-name RedLights --parent-group-name LightBulbs
```

如先前所述，CreateThingGroup 命令傳回的回應會包含物件群組、其 ID 及 ARN：

```
{
    "thingGroupName": "RedLights",
```

```
    "thingGroupId": "abcdefgh12345678ijklmnop12345678qrstuvwx",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights",
}
```

Important

在建立群組階層時，請注意下列限制：

- 一個群組最多只能有一個直系的父群組。
- 一個群組最多可以有 100 個直系的子群組。
- 群組階層的最大深度為 7 層。
- 一個群組最多可具有 50 個屬性。(屬性是您可用於存放群組相關資訊的名稱/值對。) 每個屬性名稱最多可包含 128 個字元，而每個值最多可達 800 個字元。

描述物件群組

您可以使用 DescribeThingGroup命令獲得關於物件群組的資訊：

```
$ aws iot describe-thing-group --thing-group-name RedLights
```

DescribeThingGroup 命令會傳回關於指定群組的資訊：

```
{
    "thingGroupName": "RedLights",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights",
    "thingGroupId": "12345678abcdefgh12345678ijklmnop12345678",
    "version": 1,
    "thingGroupMetadata": {
        "creationDate": 1478299948.882
        "parentGroupName": "Lights",
        "rootToParentThingGroups": [
            {
                "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/ShinyObjects",
                "groupName": "ShinyObjects"
            },
            {
                "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs",
                "groupName": "LightBulbs"
            }
        ]
    },
    "thingGroupProperties": {
        "attributePayload": {
            "attributes": {
                "brightness": "3400_lumens"
            },
        },
        "thingGroupDescription": "string"
    },
}
```

將物件加入物件群組

您可以使用 AddThingToThingGroup命令將物件加入群組：

```
$ aws iot add-thing-to-thing-group --thing-name MyLightBulb --thing-group-name RedLights
```

AddThingToThingGroup命令不會產生任何輸出。

> **Important**
>
> 一個物件最多可加入 10 個群組。但是在相同的階層內，您無法讓一個物件加入超過一個群組。(亦即，您無法讓物件加入父群組相同的兩個群組。)
> 如果物件屬於 10 個物件群組，而其中有一或多個群組為動態物件群組，則您可以使用 overrideDynamicGroups 旗標讓靜態群組的優先順序高於動態群組。

從物件群組移除物件

您可以使用 RemoveThingFromThingGroup命令從群組移除物件：

```
$ aws iot remove-thing-from-thing-group --thing-name MyLightBulb --thing-group-name
 RedLights
```

RemoveThingFromThingGroup命令不會產生任何輸出。

列出物件群組中的物件

您可以使用 ListThingsInThingGroup 命令列出屬於群組的物件：

```
$ aws iot list-things-in-thing-group --thing-group-name LightBulbs
```

ListThingsInThingGroup命令會傳回該群組的物件清單：

```
{
    "things":[
        "TestThingA"
    ]
}
```

使用 `--recursive`參數，您就可以列出屬於某個群組及其任一子群組中的物件：

```
$ aws iot list-things-in-thing-group --thing-group-name LightBulbs --recursive
```

```
{
    "things":[
        "TestThingA",
        "MyLightBulb"
    ]
}
```

> **Note**
>
> 此操作為最終一致。換言之，可能不會立即反映物件群組的變更。

列出物件群組

您可以使用 ListThingGroups命令列出您所建立的群組：

```
$ aws iot list-thing-groups
```

ListThingGroups命令會傳回一份清單，列出在您 AWS 帳戶中所定義的群組：

```
{
    "thingGroups": [
        {
            "groupName": "LightBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
        },
        {
            "groupName": "RedLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights"
        },
        {
            "groupName": "RedLEDLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLEDLights"
        },
        {
            "groupName": "RedIncandescentLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
RedIncandescentLights"
        }
        {
            "groupName": "ReplaceableObjects",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/ReplaceableObjects"
        }
    ]
}
```

使用選用的篩選條件列出以特定群組為父群組的群組 (`--parent-group`) 或是名稱開頭為特定字首 (`--name-prefix-filter`) 的群組。 `--recursive` 參數可讓您列出所有子群組，不只是物件群組的直系子群組：

```
$ aws iot list-thing-groups --parent-group LightBulbs
```

在此情況下，ListThingGroups命令會傳回一份清潭，列出在您 AWS 帳戶中所定義物件群組的直系子群組：

```
{
    "childGroups":[
        {
            "groupName": "RedLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights"
        }
    ]
}
```

使用 `--recursive` 參數與 ListThingGroups 命令來列出物件群組的所有子群組，不只是直系子群組：

```
$ aws iot list-thing-groups --parent-group LightBulbs --recursive
```

ListThingGroups命令會傳回一份清單，列出物件群組的所有子群組：

```
{
    "childGroups":[
        {
            "groupName": "RedLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights"
        },
        {
            "groupName": "RedLEDLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLEDLights"
        },
        {
```

```
            "groupName": "RedIncandescentLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
RedIncandescentLights"
        }
    ]
}
```

Note

此操作為最終一致。換言之，可能不會立即反映物件群組的變更。

列出物件的群組

您可以使用 ListThingGroupsForThing命令列出物件所屬的群組，包括任何父群組：

```
$ aws iot list-thing-groups-for-thing --thing-name MyLightBulb
```

ListThingGroupsForThing命令會傳回一份清單，列出此物件所屬的群組，包括任何父群組：

```
{
    "thingGroups":[
        {
            "groupName": "LightBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
        },
        {
            "groupName": "RedLights",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RedLights"
        },
        {
            "groupName": "ReplaceableObjects",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/ReplaceableObjects"
        }
    ]
}
```

更新物件群組

您可以使用 UpdateThingGroup命令更新物件群組的屬性：

```
$ aws iot update-thing-group --thing-group-name "LightBulbs" --thing-group-properties
 "thingGroupDescription=\"this is a test group\", attributePayload=\"{\"attributes
\"={\"Owner\"=\"150\",\"modelNames\"=\"456\"}}"
```

UpdateThingGroup命令的回應會包含更新之後的群組版本編號：

```
{
    "version": 4
}
```

Note

一個群組最多可具有 50 個屬性。

刪除物件群組

若要刪除物件群組，請使用 DeleteThingGroup命令：

```
$ aws iot delete-thing-group --thing-group-name "RedLights"
```

DeleteThingGroup命令不會產生任何輸出。

> **Important**
>
> 如果您嘗試刪除具有子物件群組的物件群組，就會收到錯誤：
>
> ```
> A client error (InvalidRequestException) occurred when calling the
>  DeleteThingGroup
> operation: Cannot delete thing group : RedLights when there are still child groups
>  attached to it.
> ```
>
> 在刪除群組之前，您必須先刪除所有子群組。

您可以刪除具有子物件的群組，但由群組成員資格授予物件的任何許可將不再適用。在刪除已連接政策的群組之前，請確認移除這些許可不會使群組中的物件無法正常運作。此外請注意，雲端中的記錄正在更新時，顯示物件所屬群組的命令 (例如 ListGroupsForThing) 可能仍會顯示出該群組。

將政策連接至物件群組

您可以使用 AttachPolicy命令將政策連接至物件群組，進而連接至該群組及其子群組中的所有物件。

```
$ aws iot attach-policy \
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \
  --policy-name "myLightBulbPolicy"
```

AttachPolicy命令不會產生任何輸出。

> **Important**
>
> 您可以附加至群組的政策數上限為兩個。
>
> **Note**
>
> 我們不建議在政策名稱中使用個人識別資訊。

`--target` 參數可以是物件群組 ARN (如上所述)、憑證 ARN 或 Amazon Cognito Identity。如需關於政策、憑證與身分驗證的詳細資訊，請參閱身分驗證 (p. 130)。

從物件群組分離政策

您可以使用 DetachPolicy命令從物件群組分離政策，進而讓政策與該群組及其子群組中的所有物件分離。

```
$ aws iot detach-policy --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
 --policy-name "myLightBulbPolicy"
```

DetachPolicy命令不會產生任何輸出。

列出連接至物件群組的政策

您可以使用 ListAttachedPolicies命令列出連接至群組的政策：

```
$ aws iot list-attached-policies --target "arn:aws:iot:us-west-2:123456789012:thinggroup/
RedLights"
```

`--target` 參數可以是物件群組 ARN (如上所述)、憑證 ARN 或 Amazon Cognito Identity。

加入選用的 `--recursive`參數，納入與該群組之父群組連接的所有政策。

ListAttachedPolicies命令會傳回政策清單：

```
{
    "policies": [
        "MyLightBulbPolicy"
        ...
    ]
}
```

列出政策所在的群組

您可以使用 ListTargetsForPolicy命令列出政策連接的目標，包括任何群組：

```
$ aws iot list-targets-for-policy --policy-name "MyLightBulbPolicy"
```

新增選用的 `--page-size` *`number`*參數來指定每次查詢所回傳的結果數量上限，並在後續呼叫使用 `--marker` *`string`* 參數以擷取下一組結果。

ListTargetsForPolicy命令會傳回一份目標清單以及用於擷取更多結果的符記：

```
{
    "nextMarker": "string",
    "targets": [ "string" ... ]
}
```

為物件取得有效政策

您可以使用 GetEffectivePolicies命令列出物件的有效政策，包括連接至該物件所屬任何群組的政策 (無論此群組是直系的父群組或旁系的上階群組)：

```
$ aws iot get-effective-policies \
  --thing-name "MyLightBulb" \
  --principal "arn:aws:iot:us-east-1:123456789012:cert/
a0c01f5835079de0a7514643d68ef8414ab739a1e94ee4162977b02b12842847"
```

使用 `--principal`參數指定連接物件之憑證的 ARN。如果您使用 Amazon Cognito Identity 驗證，請以 `--cognito-identity-pool-id` 參數並加上選用的 `--principal` 參數來指定特定的 Cognito Identity。(如果您僅指定 `--cognito-identity-pool-id`，則會傳回該 identity pool 角色對於未授權使用者的相關政策。如果您兩者都使用，則會傳回該 identity pool 角色對於授權使用者的相關政策。

`--thing-name`參數為選用，可用於取代 `--principal` 參數。若使用此參數，則會傳回連接物件所屬群組的政策，以及連接這些群組之父群組 (最高到階層裡之根群組) 的政策。

GetEffectivePolicies命令會傳回政策清單：

```
{
    "effectivePolicies": [
        {
            "policyArn": "string",
            "policyDocument": "string",
            "policyName": "string"
        }
        ...
```

```
    ]
}
```

MQTT 動作的測試授權

您可以使用 TestAuthorization命令測試 MQTT 動作是否可用於物件：

```
aws iot test-authorization \
    --principal "arn:aws:iot:us-east-1:123456789012:cert/
a0c01f5835079de0a7514643d68ef8414ab739a1e94ee4162977b02b12842847" \
    --auth-infos "{\"actionType\": \"PUBLISH\", \"resources\": [ \"arn:aws:iot:us-
east-1:123456789012:topic/my/topic\"]}"
```

使用 `--principal`參數指定連接物件之憑證的 ARN。如果使用 Amazon Cognito Identity 驗證，請指定一個 Cognito Identity 做為 `--principal` 或使用 `--cognito-identity-pool-id` 參數，或兩者並用。(如果您僅指定 `--cognito-identity-pool-id`，則需考量該 identity pool 角色對於未授權使用者的相關政策。如果您兩者都使用，則需考量該 identity pool 角色對於授權使用者的相關政策。

若要指定一個或多個您想要測試的 MQTT 動作，請列出資源以及執行 `--auth-infos`參數後的動作類型。`actionType`(動作類型) 欄位應包含「PUBLISH」、「SUBSCRIBE」、「RECEIVE」或「CONNECT」。`resources`欄位應包含一份資源 ARN 的清單。如需詳細資訊，請參閱「AWS IoT 政策 (p. 147)」。

若要測試新增政策的效果，請為其指定 `--policy-names-to-add`參數。若要測試移除政策的效果，請為其指定 `--policy-names-to-skip`參數。

您可以使用選用的 `--client-id`參數進一步加強結果。

TestAuthorization 命令會傳回您指定的每筆 `--auth-infos` 查詢所允許或拒絕的動作詳細資訊：

```
{
    "authResults": [
        {
            "allowed": {
                "policies": [
                    {
                        "policyArn": "string",
                        "policyName": "string"
                    }
                ]
            },
            "authDecision": "string",
            "authInfo": {
                "actionType": "string",
                "resources": [ "string" ]
            },
            "denied": {
                "explicitDeny": {
                    "policies": [
                        {
                            "policyArn": "string",
                            "policyName": "string"
                        }
                    ]
                },
                "implicitDeny": {
                    "policies": [
                        {
                            "policyArn": "string",
                            "policyName": "string"
                        }
```

```
                    ]
                }
            },
            "missingContextValues": [ "string" ]
        }
    ]
}
```

動態物件群組

動態物件群組會透過搜尋查詢，更新群組成員資格。使用動態物件群組，您可以根據物件的連線狀態、登錄或影子資料，變更與物件互動的方式。

由於動態物件群組是連結到您的機群索引，因此您必須啟用機群索引服務才能使用這些群組。您可以先在動態物件群組中預覽物件，再使用機群索引搜尋查詢建立群組。如需詳細資訊，請參閱機群索引服務和機群索引服務：查詢語法。

您可以將動態物件群組指定為任務的目標。只有符合定義動態物件群組條件的物件才會執行任務。

例如，假設您想要更新裝置的韌體，但需要將中斷更新的機率降至最低，則請在裝置電量大於 80% 時才更新韌體。您可以建立只有報告裝置電量超過 80% 的動態物件群組，並使用該動態物件群組做為韌體更新任務的目標。只有滿足電量條件的裝置會進行韌體更新。當裝置到達 80% 電量條件，就會新增到動態實物群組並進行韌體更新。

如需有關將物件群組指定為任務目標的詳細資訊，請參閱使用 AWS IoT 任務 API。

動態物件群組與靜態物件群組的差異如下：

- 物件成員資格沒有明確定義。若要建立動態物件群組，您必須定義查詢字串，以定義群組成員資格。
- 動態物件群組無法成為階層的一部分。
- 您會使用不同的命令集來建立、更新和刪除動態物件群組。至於所有其他操作，則可以使用與靜態物件群組互動所用的相同命令，來與動態物件群組進行互動。
- 單一帳戶不能擁有超過 100 個已定義的動態物件群組。

如需靜態物件群組的詳細資訊，請參閱物件群組 (p. 112)。

舉例來說，假設我們建立動態群組，其中包含在溫度超過華氏 60 度的倉儲中的所有房間。當房間溫度為華氏 61 度或更高時，該房間就會新增到 RoomTooWarm 動態實物群組。在 RoomTooWarm 動態實物群組中的所有房間會開啟冷卻風扇。當一個房間的溫度降到 60 度或以下，該房間會從動態實物群組中移除，其風扇也會被關閉。

建立動態物件群組

使用 CreateDynamicThingGroup 命令建立動態實物群組。若要為房間溫度過高的案例建立動態實物群組，您應使用 create-dynamic-thing-group CLI 命令：

```
$ aws iot create-dynamic-thing-group --thing-group-name "RoomTooWarm" --query-string
 "attributes.temperature>60"
```

> **Note**
>
> 我們不建議在動態實物群組名稱中使用個人識別資訊。

CreateDynamicThingGroup 命令傳回的回應包含索引名稱、查詢字串、查詢版本、物件群組名稱、物件群組 ID，以及物件群組 ARN：

```
{
    "indexName": "AWS_Things",
    "queryVersion": "2017-09-30",
    "thingGroupName": "RoomTooWarm",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RoomTooWarm",
    "queryString": "attributes.temperature>60\n",
    "thingGroupId": "abcdefgh12345678ijklmnop12345678qrstuvwx"
}
```

動態物件群組不是即時建立。動態物件群組回填需要一些時間才能完成。正在建立動態物件群組時，群組的狀態會設定為 `BUILDING`。回填處理完成時，狀態會變更為 `ACTIVE`。若要查看動態物件群組的狀態，您可以使用 DescribeThingGroup 命令。

描述動態物件群組

使用 DescribeThingGroup 命令以取得動態物件群組的相關資訊：

```
$ aws iot describe-thing-group --thing-group-name "RoomTooWarm"
```

DescribeThingGroup 命令會傳回關於指定群組的資訊：

```
{
    "status": "ACTIVE",
    "indexName": "AWS_Things",
    "thingGroupName": "RoomTooWarm",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RoomTooWarm",
    "queryString": "attributes.temperature>60\n",
    "version": 1,
    "thingGroupMetadata": {
        "creationDate": 1548716921.289
    },
    "thingGroupProperties": {},
    "queryVersion": "2017-09-30",
    "thingGroupId": "84dd9b5b-2b98-4c65-84e4-be0e1ecf4fd8"
}
```

在動態物件群組上執行 DescribeThingGroup 會傳回動態物件群組的特有屬性，例如 queryString 和狀態。

動態物件群組的狀態可能會需要以下的值：

`ACTIVE`

動態物件群組已準備就緒。

`BUILDING`

正在建立動態群組，且正在處理物件成員資格。

`REBUILDING`

正在更新動態物件群組的成員資格，接著是調整該群組搜尋查詢。

> **Note**
>
> 您在建立動態物件群組後就可以使用該群組，無論其狀態為何。只有狀態為 `ACTIVE` 的動態物件群組，才會包含所有與該動態物件群組搜尋查詢相符的物件。狀態為 `BUILDING` 和 `REBUILDING` 的動態物件群組，則可能不會包含所有符合搜尋查詢的物件。

更新動態物件群組

使用 UpdateDynamicThingGroup 命令來更新動態實物群組的屬性，包括該群組的搜尋查詢。以下命令會將成員資格變更為 > 65 來更新實物群組描述和查詢字串：

```
$ aws iot update-dynamic-thing-group --thing-group-name "RoomTooWarm" --thing-group-
properties "thingGroupDescription=\"This thing group contains rooms warmer than 65F.\"" --
query-string "attributes.temperature>65"
```

UpdateDynamicThingGroup命令的回應會包含更新之後的群組版本編號：

```
{
    "version": 2
}
```

動態物件群組不是即時更新。動態物件群組回填需要一些時間才能完成。動態物件群組更新後，群組的狀態會在該群組更新其成員資格時變更為 `REBUILDING`。回填處理完成時，狀態會變更為 `ACTIVE`。若要查看動態物件群組的狀態，您可以使用 DescribeThingGroup 命令。

刪除動態物件群組

使用 DeleteDynamicThingGroup 命令來刪除動態物件群組：

```
$ aws iot delete-dynamic-thing-group --thing-group-name "RoomTooWarm"
```

DeleteDynamicThingGroup命令不會產生任何輸出。

雲端中的記錄正在更新時，顯示物件所屬群組的命令 (例如 ListGroupsForThing) 可能仍會顯示出該群組。

限制和衝突

動態物件群組與靜態物件群組有一些相同的限制：

- 單一物件群組最多可具備 50 個屬性。
- 單一實物最多可以屬於 10 個實物群組。
- 不能重新命名物件群組。
- 物件群組名稱不能包含國際字元，例如 û、é 和 ñ。

使用動態物件群組時，請謹記下列事項。

較舊的動態物件群組優先於較新的動態物件群組

在預設情況下，如果物件已屬於 10 個物件群組，則您無法將其新增到其他群組。如果在您建立或更新動態物件群組時，動態物件群組之間的成員資格發生衝突，較舊的動態物件群組會優先於較新的動態物件群組。

啟用 `overrideDynamicGroups` 時，靜態群組會優先於動態群組

在預設情況下，如果物件已屬於 10 個物件群組，則您無法將該物件新增到其他群組。如果您正在使用 AddThingToThingGroup 或 UpdateThingGroupsForThing 命令更新物件成員資格，您可以使用 `overrideDynamicGroups` 旗標讓靜態物件群組的優先順序高於動態物件群組。啟用 `overrideDynamicGroups` 時，如果物件屬於 10 個物件群組，而其中有一或多個群組為動態，則在將該物件新增至靜態物件群組時，會從最新的動態物件群組移除該物件。

例如，假設您建立名稱為 DynamicGroup1 的動態物件群組，接著再建立九個動態物件群組，而 DynamicGroup10 是您建立的最後一個群組。如果 Thing1 屬於全部 10 個動態物件群組，手動將 Thing1 新增至靜態群組並啟用 `OverrideDynamicGroups`，就會將該物件從 DynamicGroup10 移除。

將政策到套用動態實物群組的成員

最多可屬於 10 個動態群組的實物。如果搜尋查詢字串定義動態實物群組 (其中包含已經屬於 10 個動態群組的實物)，則政策不會套用到該實物。

動態物件群組成員資格為最終一致

只會對物件的最終狀態進行登錄評估。如果狀態快速更新，可能會略過中繼狀態。請避免將規則或任務與成員資格取決於中繼狀態的動態物件群組建立關聯。

您必須啟用機群索引服務

您必須啟用機群索引服務，而機群索引回填必須先完成，您才能建立和使用動態物件群組。請預期您在啟用機群索引服務後會出現延遲。完成回填可能會需要一些時間。您註冊的物件越多，回填的處理時間就越長。在您為動態物件群組啟用機群索引服務後，除非您刪除所有動態物件群組，否則無法停用服務。

> **Note**
>
> 如果您有查詢機群索引的許可，您就可以存取整個機群的物件資料。

為您的 AWS IoT 資源建立標籤

為了協助您管理和組織實物群組、實物類型、主題規則、任務、排程稽核及安全性描述檔，您可以選擇性地以標籤形式將您自己的中繼資料指派給其中每個資源。本節說明標籤並示範如何建立它們。

為了協助您管理與物件相關的成本，您可以建立包含物件的帳單群組 (p. 127)。您可以將包括中繼資料的標籤指派到每個帳單群組。本節也會討論可用來建立和管理的帳單群組和命令。

標籤基本概念

標籤可讓您以不同的方式分類您的 AWS IoT 資源，例如依據目的、擁有者或環境。當您有許多相同類型的資源時，這將會很有用，因為您可以依據先前指派的標籤，快速識別特定的資源。每個標籤皆包含由您定義的一個「索引鍵」與選擇性的「值」。例如，您可以為您的物件類型定義一組標籤，協助您以類型追蹤裝置。我們建議您為每種資源類型建立符合您需求的一組標籤金鑰。使用一致的標籤金鑰組可讓您更輕鬆的管理您的資源。

您可以根據新增或套用的標籤來搜尋與篩選資源。您也可以使用帳單群組標籤來分類和追蹤您的成本。您也可以使用標籤來控制對您資源的存取，如 搭配 IAM 政策使用標籤 (p. 126) 所述。

為了易用性，AWS 管理主控台中的標籤編輯器提供您集中、統一的方式來建立和管理標籤。如需詳細資訊，請參閱使用 AWS 管理主控台中的使用標籤編輯器。

您也可以利用 AWS CLI 及 AWS IoT API 來使用標籤。當您使用下列命令在「標籤」欄位中建立標籤時，您可以將標籤與實物群組、實物類型、主題規則、任務、安全性描述檔及帳單群組建立關聯：

- CreateBillingGroup (p. 722)
- CreateDynamicThingGroup (p. 726)
- CreateJob (p. 729)
- CreateOTAUpdate (p. 738)
- CreateScheduledAudit (p. 748)
- CreateSecurityProfile (p. 750)
- CreateStream (p. 753)
- CreateThingGroup (p. 758)
- CreateThingType (p. 761)
- CreateTopicRule (p. 763)

您可以使用下列命令新增、修改或刪除支援標記功能的現有資源標籤：

- TagResource (p. 1015)
- ListTagsForResource (p. 951)
- UntagResource (p. 1023)

您可以編輯標籤金鑰和值，並且可以隨時從資源移除標籤。您可以將標籤的值設為空白字串，但您無法將標籤的值設為 null。若您將與現有標籤具有相同鍵的標籤新增到該資源，則新值會覆寫舊值。如果您刪除資源，也會刪除與該資源相關聯的任何標籤。

如需其他資訊，請參閱 AWS 標記策略。

標籤的限制與上限

以下基本限制適用於標籤：

- 每個資源的標籤數上限 — 50
- 金鑰長度上限 — 127 個 UTF-8 Unicode 字元
- 值長度上限 — 255 個 UTF-8 Unicode 字元
- 標籤鍵與值皆區分大小寫。
- 標籤名稱或值不可使用 "aws:" 字首，因為它只保留給 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。
- 如果您的標籤建立結構描述跨多項服務和資源使用，請記得，其他服務可能有字元使用限制。通常，允許使用的字元為：可用 UTF-8 表示的英文字母、空格和數字，以及以下特殊字元：+ - = . _ : / @。

搭配 IAM 政策使用標籤

您可以在用於 AWS IoT API 動作的 IAM 政策中，套用以標籤為基礎的資源層級許可。這可讓您更有效地控制使用者可以建立、修改或使用哪些資源。您可以使用 `Condition` 元素 (也稱為 `Condition` 區塊)，以及 IAM 政策中的以下條件內容金鑰和值，來根據資源標籤控制使用者存取 (許可)：

- 使用 `aws:ResourceTag/`*`tag-key`*`:` *`tag-value`* 以允許或拒絕資源上具有特定標籤的使用者動作。
- 使用 `aws:RequestTag/`*`tag-key`*`:` *`tag-value`* 以在提出 API 請求時，要求使用 (或不使用) 特定標籤，以建立或修改允許標籤的資源。
- 使用 `aws:TagKeys: [`*`tag-key`*`, ...]` 以在提出 API 請求時，要求使用 (或不使用) 特定標籤金鑰集，以建立或修改允許標籤的資源。

> **Note**
>
> IAM 政策中的條件內容金鑰和值，只會套用到那些資源識別符可標記為必要參數的 AWS IoT 動作。例如，根據條件內容金鑰和值，使用 DescribeEndpoint (p. 823) 將不會受到允許或拒絕，因為在此請求中所參照的項目沒有可標記資源 (實物群組、實物類型、主題規則、任務或安全性描述檔)。

AWS Identity and Access Management 使用者指南中的使用標籤控制存取有更多標籤的使用資訊。該指南的 IAM JSON 政策參考章節有詳細的語法、說明，還有元素、變數範例，以及在 IAM 中的 JSON 政策評估邏輯。

以下範例政策會套用兩個以標籤為基礎的限制。受到此政策限制的 IAM 使用者：

- 無法給予資源 "env = prod" 標籤 (在範例中，請參閱此行 `"aws:RequestTag/env" : "prod"`
- 無法修改或存取包含現有標籤 "env = prod" 的資源 (在範例中，請參閱此行 `"aws:ResourceTag/env" : "prod"`)。

```
{
    "Version" : "2012-10-17",
    "Statement" : [
        {
          "Effect" : "Deny",
          "Action" : "iot:*",
          "Resource" : "*",
          "Condition" : {
            "StringEquals" : {
              "aws:RequestTag/env" : "prod"
            }
          }
```

```
        },
        {
          "Effect" : "Deny",
          "Action" : "iot:*",
          "Resource" : "*",
          "Condition" : {
            "StringEquals" : {
              "aws:ResourceTag/env" : "prod"
            }
          }
        },
        {
          "Effect": "Allow",
          "Action": [
            "iot:*"
          ],
          "Resource": "*"
        }
    ]
}
```

您也可以透過將其包含在清單中，為特定標籤金鑰指定多個標籤值，如下所示：

```
            "StringEquals" : {
              "aws:ResourceTag/env" : ["dev", "test"]
            }
```

> **Note**
>
> 如果您允許或拒絕使用者根據標籤存取資源，請務必考慮明確拒絕使用者將這些標籤新增至相同資源或從中移除的能力。否則，使用者可能透過修改標籤來避開您的限制，並取得資源的存取。

帳單群組

AWS IoT 不允許您將標籤直接套用至個別物件，但允許您將物件置於帳單群組中並將標籤套用到群組。針對 AWS IoT，根據標籤分配的成本和用量資料受限於帳單群組。

下列命令可供使用：

- AddThingToBillingGroup (p. 705) 將物件加入帳單群組。
- CreateBillingGroup (p. 722) 建立帳單群組。
- DeleteBillingGroup (p. 778) 刪除帳單群組。
- DescribeBillingGroup (p. 815) 傳回有關帳單群組的資訊。
- ListBillingGroups (p. 915) 列出您建立的帳單群組。
- ListThingsInBillingGroup (p. 966) 列出您已新增至特定帳單群組的物件。
- RemoveThingFromBillingGroup (p. 983) 從帳單群組移除特定物件。
- UpdateBillingGroup (p. 1027) 更新關於帳單群組的資訊。
- CreateThing (p. 756) 可讓您在建立物件時，為其指定帳單群組。
- DescribeThing (p. 848) 傳回物件的描述，包括該物件所屬的帳單群組 (若有)。

檢視成本分配和用量資料

您可以使用帳單群組標籤來分類和追蹤您的成本。當您將標籤套用至帳單群組 (也會套用到其所包含的物件)，AWS 會以逗號分隔值 (CSV) 檔案格式產生一份成本分配報告，其中包含按標籤彙總的用量與成本。您

可以套用代表業務類別 (例如成本中心、應用程式名稱或擁有者) 的標籤，來整理多個服務中的成本。如需有關使用標籤進行成本分配的詳細資訊，請參閱 AWS 帳單和成本管理使用者指南中的使用成本分配標籤。

> **Note**
>
> 若要準確地將用量和成本資料與您已置於帳單群組的物件建立關聯，每個裝置或應用程式皆必須：
>
> - 在 AWS IoT 已註冊為物件 (請參閱利用 AWS IoT 管理裝置 (p. 106))。
> - 只使用物件名稱做為用戶端 ID，透過 MQTT 連接到 AWS IoT 訊息中介裝置 (請參閱AWS IoT 的訊息中介裝置 (p. 242))。
> - 使用與物件關聯的用戶端憑證驗證。

帳單群組可使用以下定價維度 (根據與帳單群組關聯的物件活動)：

- 連線能力 (根據做為用戶端 ID 以用於連線的物件名稱)
- 傳訊 (根據物件傳入和傳出的訊息；僅限 MQTT)
- 影子操作 (根據其訊息觸發影子更新的物件)
- 觸發的規則 (根據其傳入訊息已觸發規則的實物；不適用於 MQTT 生命週期事件所觸發的那些規則)
- 物件索引更新 (根據新增到索引的物件)
- 遠端動作 (根據更新的物件)
- 偵測 (p. 598) 報告 (根據已報告的物件活動)

根據標籤的成本和用量資料 (以及帳單群組的報告) 不會反映以下活動：

- 裝置登錄操作 (包括實物、實物群組和實物類型的更新；請參閱利用 AWS IoT 管理裝置 (p. 106))
- 物件群組索引更新 (在新增物件群組時)
- 索引搜尋查詢
- 裝置佈建 (p. 470)
- 稽核 (p. 506) 報告

限制

- 物件只能屬於一個帳單群組。
- 與物件群組不同，帳單群組不能組織成階層。
- 為了將其用量註冊為標記或帳單用途，裝置必須：
  - 在 AWS IoT 已註冊為物件。
  - 只使用 MQTT 與 AWS IoT 通訊。
  - 只使用其物件名稱做為 clientID 與 AWS IoT 進行驗證。
  - 只使用 X.509 憑證或 Amazon Cognito 身分進行驗證。

  更多詳細資訊，請參閱利用 AWS IoT 管理裝置 (p. 106)、身分驗證 (p. 130)和裝置佈建 (p. 470)。API 命令 AttachThingPrincipal (p. 713) 可用來將憑證或其他登入資料附加到物件。
- 每個帳戶的帳單群組數上限為 20,000。

AWS IoT 中的安全性

雲端安全是 AWS 最重視的一環。身為 AWS 客戶的您，將能從資料中心和網路架構的建置中獲益，以滿足組織最為敏感的安全要求。

安全是 AWS 與您共同肩負的責任。共同的責任模型 將此描述為雲端 本身 的安全和雲端 內部 的安全：

- 雲端本身的安全 – AWS 負責保護執行 AWS 雲端內 AWS 服務的基礎設施。AWS 提供的服務，也可讓您安全使用。在 AWS 合規計劃中，第三方稽核員會定期測試並驗證我們安全性的功效。若要了解適用於 AWS IoT 的合規計劃，請參閱合規計劃範圍內的 AWS 服務。
- 雲端內部的安全 – 您的責任取決於所使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的要求和適用法律和法規。

本文件有助於您了解如何在使用 AWS IoT 時套用共同責任模型。下列主題將示範如何設定 AWS IoT 以達到您的安全和合規目標。您也將了解如何使用其他 AWS 服務，幫助您監控並保護 AWS IoT 資源。

主題

- AWS IoT 安全性 (p. 129)
- 身分驗證 (p. 130)
- 管理裝置憑證 (p. 145)
- 授權 (p. 145)
- AWS IoT Core 的資料保護 (p. 177)
- 適用於 AWS IoT 的 Identity and Access Management (p. 179)
- 使用 AWS IoT Core 進行記錄和監控 (p. 200)
- AWS IoT Core 的合規驗證 (p. 232)
- AWS IoT Core 中的彈性 (p. 232)
- AWS IoT 中的基礎設施安全 (p. 233)
- AWS IoT Core 中的漏洞分析和管理 (p. 233)
- AWS IoT Core 中的安全性最佳實務 (p. 233)

AWS IoT 安全性

每個連線的裝置或用戶端都必須具有認證才能與 AWS IoT 互動。所有往返 AWS IoT 的流量都會透過 Transport Layer Security (TLS) 安全地傳送。資料在 AWS IoT 與其他 AWS 服務間移動時，AWS 雲端的安全機制將保護其安全。

- 您需負責管理 AWS IoT 中的裝置登入資料 (X.509 憑證、AWS 登入資料、Amazon Cognito 身分、聯合身分或自訂身分驗證字符) 以及政策。如需詳細資訊，請參閱AWS IoT 中的金鑰管理 (p. 179)。您務必為每一裝置指派唯一身分，並管理每個裝置或裝置群組的許可。
- 您的裝置透過安全的 TLS 連線，使用 X.509 憑證或 Amazon Cognito 身分連線至 AWS IoT。在研究和開發期間，以及對於進行 API 呼叫或使用 WebSocket 的某些應用程式，您也可以使用 IAM 使用者和群組或自訂驗證字符來進行驗證。如需詳細資訊，請參閱IAM 使用者、群組和角色 (p. 137)。
- 使用 AWS IoT 驗證時，訊息中介裝置負責驗證您的裝置、安全地擷取裝置資料，以及使用 AWS IoT 政策授與或拒絕您為裝置指定的存取許可。
- 使用自訂驗證時，自訂授權方負責驗證您的裝置，以及授與或拒絕您使用 AWS IoT 或 IAM 政策為裝置指定的存取許可。
- AWS IoT 規則引擎會根據您定義的規則，將裝置資料轉送至其他裝置或其他 AWS 服務。它使用 AWS Identity and Access Management 將資料安全地傳輸到其最終目的地。如需詳細資訊，請參閱適用於 AWS IoT 的 Identity and Access Management (p. 179)。

身分驗證

身分驗證是一種機制，您可以用來驗證用戶端或伺服器的身分。伺服器身分驗證是裝置或其他用戶端用以確保它們是與實際 AWS IoT 端點通訊的程序。用戶端身分驗證是裝置或其他用戶端向 AWS IoT 驗證自己身分的程序 。

X.509 憑證概觀

X.509 憑證為數位憑證，其依據 X.509 公有金鑰基礎設施標準，將公有金鑰與憑證內含的身分建立關聯。X.509 憑證由稱為認證機構 (CA) 的可信任實體發行。CA 負責維護一個或多個稱為 CA 憑證的特殊憑證，用以發行 X.509 憑證。僅認證機構可存取 CA 憑證。X.509 憑證鏈同時用於用戶端的伺服器身分驗證，和伺服器的用戶端身分驗證。

伺服器驗證

當您的裝置或其他用戶端嘗試連線至 AWS IoT 時，AWS IoT 伺服器會傳送 X.509 憑證，讓您的裝置用來驗證伺服器。身分驗證是透過在 TLS 層驗證 X.509 憑證鏈進行。這與您造訪 HTTTPS URL 時瀏覽器所使用的方法相同。

當您的裝置或其他用戶端建立與 AWS IoT Core 端點的 TLS 連線時，AWS IoT Core 會顯示憑證鏈，讓裝置用來驗證它們正在與 AWS IoT Core 通訊，而非其他冒充 AWS IoT Core 的伺服器。所呈現的鏈，取決於裝置所連線端點的類型以及用戶端和 AWS IoT 在 TLS 交握期間所交涉密碼套件的組合。

端點類型

AWS IoT Core 支援兩種不同的資料端點類型：`iot:Data` 和 `iot:Data-ATS`。`iot:Data-ATS` 端點呈現由 VeriSign 類別 3 公開主要 G5 根 CA 憑證簽署的憑證。`iot:Data-ATS` 端點呈現由 Amazon Trust Services CA 簽署的伺服器憑證。

ATS 端點呈現的憑證由 Starfield 交叉簽署。某些 TLS 用戶端實作需要驗證信任根，並要求在用戶端的信任存放區中安裝 Starfield CA 憑證。

> **Warning**
>
> 不建議使用雜湊整個憑證 (包括發行者名稱等) 的憑證關聯方法，因為這會導致憑證驗證失敗，因為我們提供的 ATS 憑證是由 Starfield 交叉簽署，且具有不同的發行者名稱。

除非您的裝置需要 Symantec 或 Verisign CA 憑證，否則請使用 `iot:Data-ATS` 端點。Symantec 和 Verisign 憑證已被棄用，大多數 Web 瀏覽器不再支援。

使用 describe-endpoint 來建立 ATS 端點：

aws iot describe-endpoint --endpoint-type iot:Data-ATS

describe-endpoint 會傳回以下形式的端點：

`<account-specific-prefix>``.iot.`*`<your-region>`*`.amazonaws.com`

第一次呼叫 `describe-endpoint` 時，會建立一個端點。所有後續呼叫 `describe-endpoint` 會傳回相同的端點。

基於回溯相容性考量，AWS IoT 仍支援 Symantec 端點。如需詳細資訊，請參閱 AWS IoT Core 如何協助客戶應對即將到來的不信任 Symantec 憑證授權單位。在 ATS 端點上運作的裝置將能與同一帳戶中運作於 Symantec 端點上的裝置完全互通，而不需要進行任何重新註冊。

伺服器身分驗證的 CA 憑證

根據您使用的資料端點類型以及您已交涉的密碼套件，AWS IoT 伺服器身分驗證憑證是由下列其中一個根 CA 憑證簽署：

VeriSign 端點 (舊版)

- RSA 2048 位元金鑰：VeriSign 類別 3 公開主要 G5 根 CA 憑證

Amazon Trust Services 端點 (首選)

- RSA 2048 位元金鑰：Amazon 根 CA 1。
- RSA 4096 位元金鑰：Amazon Root CA 2。預留以供日後使用。
- ECC 256 位元金鑰：Amazon 根 CA 3。
- ECC 384 位元金鑰：Amazon Root CA 4。預留以供日後使用。

這些憑證都是由 Starfield 根 CA 憑證交叉簽署的。所有新的 AWS IoT Core 區域自 2018 年 5 月 9 日起，隨著 AWS IoT Core 於亞太 (孟買) 區域推出，將僅提供 ATS 憑證。

伺服器身分驗證準則

有許多變數會影響裝置驗證 AWS IoT Core 伺服器驗證憑證的能力。例如，裝置的記憶體可能受限，無法保存所有可能的根 CA 憑證，或者裝置可能實作非標準的憑證驗證方法。基於這些原因，我們建議您遵循以下準則：

- 我們建議您使用 ATS 端點並安裝所有支援的 Amazon 根 CA 憑證。

- 如果您無法在裝置上存放全部這些憑證，而且您的裝置未使用 ECC 式驗證，則可以略過 Amazon 根 CA 3 和 Amazon 根 CA 4 ECC 憑證。如果您的裝置未實作 RSA 式憑證驗證，則可以略過 Amazon 根 CA 1 和 Amazon 根 CA 2 RSA 憑證。
- 如果您在連線至 ATS 端點時遇到伺服器憑證驗證問題，請嘗試將相關的交叉簽署 Amazon Root CA 憑證新增至信任存放區。
  - 交叉簽署的 Amazon 根 CA 1
  - 交叉簽署的 Amazon 根 CA 2 - 預留以供日後使用。
  - 交叉簽署的 Amazon 根 CA 3
  - 交叉簽署的 Amazon 根 CA 4 - 預留以供日後使用。
- 如果您遇到伺服器憑證驗證問題，則您的裝置可能需要明確地信任根 CA。嘗試將 Starfield 根 CA 憑證新增至您的信任存放區。
- 如果您在執行上述步驟後仍然遇到問題，請聯絡 AWS 開發人員支援。

> **Note**
>
> CA 憑證在過期日期之後，即無法用於驗證伺服器的憑證。CA 憑證過期日期之前，可能需要進行替代。請確認您能夠於所有裝置更新根 CA 憑證，以維持連線能力並取得最新的安全最佳實務。

> **Note**
>
> 在裝置代碼中連接到 AWS IoT 時，請將憑證傳遞到您用於連接的 API。您使用的 API 會因 SDK 而異。如需詳細資訊，請參閱 AWS IoT 裝置 SDK。

用戶端身分驗證

AWS IoT 支援裝置或用戶端身分驗證的五種身分委託人類型：

- X.509 用戶端憑證 (p. 132)
- IAM 使用者、群組和角色 (p. 137)
- Amazon Cognito 身分 (p. 137)

這些身分可用於裝置、行動裝置、網頁或桌面應用程式。輸入 AWS IoT 命令列界面 (CLI) 命令的使用者也可以使用它們。一般而言，AWS IoT 裝置使用 X.509 憑證，而行動應用程式使用 Amazon Cognito 身分。Web 和桌面應用程式則使用 IAM 或聯合身分。CLI 命令使用 IAM。如需 IAM 身分的詳細資訊，請參閱適用於 AWS IoT 的 Identity and Access Management (p. 179)。

X.509 用戶端憑證

相較於其他識別和身分驗證機制，X.509 憑證具備多種好處。X.509 憑證可讓裝置使用非對稱金鑰。亦即您可將私有金鑰燒錄至某裝置安全的儲存空間。透過這種方式，敏感的加密資料永遠不會離開裝置。由於私有金鑰絕對不會離開裝置，因此 X.509 憑證提供更勝於其他機制 (例如使用者名稱和密碼或承載符記) 的強大用戶端身分驗證。

AWS IoT 使用 TLS 通訊協定的用戶端身分驗證模式來驗證用戶端憑證。TLS 支援多種程式設計語言與作業系統，且普遍用於加密資料。進行 TLS 用戶端身分驗證時，AWS IoT 會要求 X.509 用戶端憑證，並針對憑證的登錄檔，確認憑證狀態與 AWS 帳戶。之後，會針對憑證內公開金鑰所對應的私有金鑰，要求用戶端證明該私有金鑰的所有權。

AWS IoT 支援下列用戶端簽署憑證的演算法：

- SHA256WITHRSA
- SHA384WITHRSA
- SHA512WITHRSA
- DSA_WITH_SHA256

- ECDSA-WITH-SHA256
- ECDSA-WITH-SHA384
- ECDSA-WITH-SHA512

AWS IoT 可以使用 AWS IoT 產生的 X.509 憑證或您自己的 X.509 憑證來進行裝置身分驗證。AWS IoT 產生的 X.509 憑證時期很長 (但會於 2049-12-31T23:59:59Z，即 2049 年 12 月 31 日 GMT 午夜時到期。) 至於經 CA 憑證簽署的憑證，其過期日期與時間則於憑證建立時設定。

若要使用非由 AWS IoT 建立的 X.509 憑證，您必須註冊 CA 憑證。所有裝置憑證都必須由您註冊的 CA 憑證進行簽署。

> **Note**
>
> 我們建議為每一裝置提供唯一的憑證，藉此更精細地進行管理，包括憑證撤銷作業。裝置必須支援憑證的輪換和替代，以確保憑證過期時操作順暢。

您可以使用 AWS IoT 主控台或 CLI，執行下列憑證操作：

- 建立並註冊 AWS IoT 憑證。
- 註冊 CA 憑證。
- 註冊裝置憑證。
- 啟用或停用裝置憑證。
- 撤銷裝置憑證。
- 將裝置憑證轉移至另一個 AWS 帳戶。
- 列出所有於您 AWS 帳戶註冊的 CA 憑證。
- 列出所有於您 AWS 帳戶註冊的裝置憑證。

如需關於執行這些操作使用的 CLI 命令之詳細資訊，請參閱 AWS IoT CLI 參考。

如需使用 AWS IoT 主控台建立憑證的詳細資訊，請參閱建立並註冊 AWS IoT 裝置憑證 (p. 133)。

如需使用您自己的 X.509 憑證的詳細資訊，請參閱使用自有憑證 (p. 134)。

建立並註冊 AWS IoT 裝置憑證

您可以使用 AWS IoT 主控台或 AWS CLI 建立 AWS IoT 憑證。

若透過主控台建立憑證

1. 登入 AWS 管理主控台，並開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Security (安全)、Certificates (憑證)，然後選擇 Create (建立)。
3. 選擇 One-click certificate creation (按一下建立憑證) - Create certificate (建立憑證)。或者，若使用憑證簽署要求 (CSR) 產生憑證，請選擇 Create with CSR (透過 CSR 建立)。
4. 將公開金鑰、私有金鑰和憑證下載到安全的位置。
5. 選擇 Activate (啟用)。
6. 請選擇 Done (完成)。您也可以選擇 Attach a policy (連接政策)，將現有 AWS IoT 政策連接至憑證。

若透過 CLI 建立憑證

AWS CLI 提供兩個建立憑證的命令：

- create-keys-and-certificate

  這個命令會建立私有金鑰、公開金鑰和 X.509 憑證。

- create-certificate-from-csr

  此命令會建立給定 CSR 的憑證。

使用自有憑證

若要使用自己的 X.509 裝置憑證，您必須向 AWS IoT 註冊 CA 憑證。然後才可以使用該 CA 憑證簽署裝置憑證。您每個 AWS 區域的每個 AWS 帳戶，相同主體欄位可用於註冊至多 10 個 CA 憑證。如此一來，您能夠使用不只一個 CA 來簽署裝置憑證。

> **Note**
>
> 裝置憑證都必須由已註冊的 CA 憑證進行簽署。通常會使用一個 CA 憑證來建立中繼 CA 憑證。若您使用中繼憑證簽署裝置憑證，您必須註冊該中繼 CA 憑證。當您連接至 AWS IoT 時，即使您註冊自有根 CA 憑證，請使用 AWS IoT 的根 CA 憑證。裝置可使用 AWS IoT 的根 CA 憑證，以驗證 AWS IoT 的伺服器身分。

主題

- 註冊您的 CA 憑證 (p. 134)
- 使用您的 CA 憑證建立裝置憑證 (p. 135)
- 註冊裝置憑證 (p. 136)
- 手動註冊裝置憑證 (p. 136)
- 使用裝置憑證自動/即時註冊 (p. 136)
- 停用 CA 憑證 (p. 137)
- 撤銷裝置憑證 (p. 137)

若您沒有 CA 憑證，可以使用 OpenSSL 工具來建立。

若要建立 CA 憑證

1. 產生金鑰對。

   ```
   openssl genrsa -out rootCA.key 2048
   ```

2. 透過該金鑰對的私有金鑰來產生 CA 憑證。

   ```
   openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.pem
   ```

註冊您的 CA 憑證

> **Note**
>
> CA 憑證不可註冊至相同 AWS 區域內一個以上的帳戶。不過，若帳戶位於不同 AWS 區域，CA 憑證即可註冊到多個帳戶。

若要註冊 CA 憑證

1. 自 AWS IoT 取得註冊碼。此代碼將做為私有金鑰驗證憑證的 `Common Name`：

   ```
   aws iot get-registration-code
   ```

2. 產生私有金鑰驗證憑證的金鑰對：

   ```
   openssl genrsa -out verificationCert.key 2048
   ```

3. 建立私有金鑰驗證憑證的 CSR。將憑證的 `Common Name` 欄位設定為您的註冊碼。

   ```
   openssl req -new -key verificationCert.key -out verificationCert.csr
   ```

會顯示提示，要求您輸入一些此憑證的資訊，包括 `Common Name`。

```
Country Name (2 letter code) [AU]:
    State or Province Name (full name) []:
    Locality Name (for example, city) []:
    Organization Name (for example, company) []:
    Organizational Unit Name (for example, section) []:
    Common Name (e.g. server FQDN or YOUR name)
 []:XXXXXXXXXXXXMYREGISTRATIONCODEXXXXXX
    Email Address []:
```

4. 使用 CSR 建立私有金鑰驗證憑證：

   openssl x509 -req -in verificationCert.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out verificationCert.pem -days 500 -sha256

5. 向 AWS IoT 註冊該 CA 憑證。將該 CA 憑證和私有金鑰驗證憑證傳入 register-ca-certificate CLI 命令：

   aws iot register-ca-certificate --ca-certificate file://rootCA.pem --verification-cert file://verificationCert.pem

6. 使用 update-certificate CLI 命令來啟用 CA 憑證：

   aws iot update-ca-certificate --certificate-id xxxxxxxxxxx --new-status ACTIVE

使用您的 CA 憑證建立裝置憑證

您可以使用向 AWS IoT 註冊的 CA 憑證，建立裝置憑證。裝置憑證必須向 AWS IoT 註冊後方能使用。

若要建立裝置憑證

1. 產生金鑰對。

   openssl genrsa -out deviceCert.key 2048

2. 建立裝置憑證的 CSR。

   openssl req -new -key deviceCert.key -out deviceCert.csr

   將出現提示，要求您輸入一些資訊，如下所示：

```
Country Name (2 letter code) [AU]:
    State or Province Name (full name) []:
    Locality Name (for example, city) []:
    Organization Name (for example, company) []:
    Organizational Unit Name (for example, section) []:
    Common Name (e.g. server FQDN or YOUR name) []:
    Email Address []:
```

3. 透過 CSR 建立裝置憑證。

   openssl x509 -req -in deviceCert.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out deviceCert.pem -days 500 -sha256

   > **Note**
   >
   > 您必須使用向 AWS IoT 註冊的 CA 憑證來簽署裝置憑證。若您的 AWS 帳戶註冊多個 CA 憑證 (主體欄位和公開金鑰均相同)，在您註冊裝置憑證時，必須指定用於建立該憑證的 CA 憑證。

4. 註冊裝置憑證：

   aws iot register-certificate --certificate-pem file://deviceCert.pem --ca-certificate-pem file://rootCA.pem

5. 使用 update-certificate CLI 命令啟用裝置憑證：

   aws iot update-certificate --certificate-id xxxxxxxxxxx --new-status ACTIVE

註冊裝置憑證

您必須使用向 AWS IoT 註冊的 CA 憑證來簽署裝置憑證。若您的 AWS 帳戶註冊多個 CA 憑證 (主體欄位和公開金鑰均相同)，在您註冊裝置憑證時，必須指定用於簽署該憑證的 CA 憑證。您可以手動註冊每一裝置憑證，或者使用自動註冊，在裝置首次連接至 AWS IoT，即註冊其憑證。

手動註冊裝置憑證

使用 register-certificate CLI 命令來註冊裝置憑證：

aws iot register-certificate --certificate-pem file://deviceCert.crt --ca-certificate-pem file://caCert.crt

使用裝置憑證自動/即時註冊

為了在裝置首次連接至 AWS IoT 就自動註冊裝置憑證，您必須啟用 CA 憑證的自動註冊。如此一來，您 CA 憑證所簽署的裝置憑證就會在連接至 AWS IoT 時註冊。

啟用自動註冊

使用 update-ca-certificate CLI 命令將 CA 憑證的 `auto-registration-status` 設定為 `ENABLE`：

aws iot update-ca-certificate --certificate-id ***caCertificateId*** --new-auto-registration-status ENABLE

若您使用 `auto-registration-status`API 註冊 CA 憑證，也可以將 `ENABLE` 設定為 register-ca-certificate：

aws iot register-ca-certificate --ca-certificate file://rootCA.pem --verification-cert file://privateKeyVerificationCert.crt --allow-auto-registration

當裝置在 TLS 交握期間第一次嘗試連線至 AWS IoT，裝置必須顯示一個檔案，其中同時包含已註冊的 CA 憑證以及由 CA 憑證所簽署的裝置憑證。您可以使用類似下面的命令來組合這兩個檔案：

cat deviceCert.crt sampleCACertificate.pem > deviceCertAndCACert.crt

當您連線到 AWS IoT 時，請使用 `deviceCertAndCACert.crt` 檔案做為憑證檔案。AWS IoT 會將 CA 憑證辨識為已註冊的 CA 憑證、註冊裝置憑證，並將其狀態設定為 `PENDING_ACTIVATION`。亦即會自動註冊該裝置憑證，並準備啟用。憑證狀態必須為 `ACTIVE` 狀態，之後才可用於連接至 AWS IoT。當 AWS IoT 自動註冊憑證，或當裝置呈現狀態為 `PENDING_ACTIVATION` 的憑證時，AWS IoT 會發佈訊息至下列 MQTT 主題：

`$aws/events/certificates/registered/`*`caCertificateID`*

`caCertificateID`為發行該裝置憑證的 CA 憑證 ID。

發佈至此主題的訊息具有以下結構：

```
{
        "certificateId": "certificateID",
        "caCertificateId": "caCertificateId",
        "timestamp": timestamp,
        "certificateStatus": "PENDING_ACTIVATION",
        "awsAccountId": "awsAccountId",
        "certificateRegistrationTimestamp": "certificateRegistrationTimestamp"
    }
```

您可以建立規則以監聽此主題，並執行部分動作。我們建議您建立 Lambda 規則，此規則可驗證裝置憑證未列於憑證撤銷清單 (CRL) 中，同時啟用憑證，並建立和連接政策至憑證。政策會決定裝置能夠存取哪些資

源。如需關於如何建立會監聽 `$aws/events/certificates/registered/caCertificateID` 主題並執行這些動作的 Lambda 規則之詳細資訊，請參閱即時註冊。

若裝置憑證自動註冊發生錯誤或例外，AWS IoT 會傳送事件或訊息至您的 CloudWatch Logs 日誌。如需關於您帳戶日誌設定的詳細資訊，請參閱 Amazon CloudWatch 文件相關文章。

停用 CA 憑證

當您註冊裝置憑證時，AWS IoT 會檢查用來簽署裝置憑證的 CA 憑證是否為 `ACTIVE`。若該 CA 憑證為 `INACTIVE`，AWS IoT 不允許該裝置憑證進行註冊。藉由將 CA 憑證標記為 `INACTIVE`，您可以防止由 CA 發行的任何新裝置憑證註冊至您的帳戶。您可以使用 update-ca-certificate CLI 命令來停用 CA 憑證：

aws iot update-ca-certificate --certificate-id ***certificateId*** --new-status INACTIVE

> **Note**
>
> 由遭入侵 CA 憑證簽署的任何已註冊裝置憑證，將持續運作，直到您確實撤銷這些憑證。

使用 `ListCertificatesByCA` API 或 list-certificates-by-ca CLI 命令，可取得由指定 CA 簽署的所有已註冊裝置憑證清單。針對由指定 CA 憑證所簽署的每一裝置憑證，請使用 `UpdateCertificate` API 來撤銷該裝置憑證，以避免受到使用。

撤銷裝置憑證

若您已註冊的裝置憑證偵測到可疑活動，您可以使用 update-certificate CLI 命令來將其撤銷：

aws iot update-certificate --certificate-id ***certificateId*** --new-status REVOKED

IAM 使用者、群組和角色

IAM 使用者、群組和角色為管理 AWS 身分及身分驗證的標準機制，這些功能可使用 AWS SDK 和 CLI，以連接至 AWS IoT HTTP 介面。

IAM 角色亦允許 AWS IoT 代您存取您帳戶的其他 AWS 資源。例如，若您希望將裝置的狀態發佈至 DynamoDB 表格，IAM 角色允許 AWS IoT 與 Amazon DynamoDB 互動。如需詳細資訊，請參閱 IAM 角色。

若訊息中介裝置透過 HTTP 進行連線，AWS IoT 會透過 Signature 第 4 版簽署流程，驗證 IAM; 使用者、群組和角色。如需更多詳細資訊，請參閱簽署 AWS API 請求相關文章。

當您於 AWS IoT 使用 AWS Signature 第 4 版，用戶端的 TLS 實作必須支援下列項目：

- TLS 1.2、TLS 1.1、TLS 1.0
- SHA-256 RSA 憑證簽章驗證
- TLS 密碼套件支援部分的一種密碼套件

如需相關資訊，請參閱「適用於 AWS IoT 的 Identity and Access Management (p. 179)」。

Amazon Cognito 身分

Amazon Cognito Identity 可讓您建立暫時、有限權限的 AWS 登入資料，以便在行動和 Web 應用程式中使用。使用 Amazon Cognito Identity 時，您可以建立身分集區來為使用者建立唯一身分，並使用身分提供者 (例如使用 Amazon、Facebook 和 Google 登入) 進行身分驗證。您也可以將 Amazon Cognito 身分識別與自己的開發人員已驗證身分搭配使用。如需更多詳細資訊，請參閱 Amazon Cognito 身分。

若要使用 Amazon Cognito Identity，您可以定義與 IAM 角色相關聯的 Amazon Cognito 身分集區。IAM 角色與 IAM 政策相關聯，可授予身分集區中的身分存取 AWS 資源的許可，例如呼叫 AWS 服務。

Amazon Cognito Identity 會建立未驗證和已驗證的身分。未驗證的身分適用於行動裝置或 Web 應用程式中想使用應用程式但不想登入的訪客使用者。未驗證的使用者只會獲得與身分集區關聯之 IAM 政策中所指定的許可。

當您使用已驗證的身分，除了連接到身分集區的 IAM 政策，您還可以使用 AttachPrincipalPolicy API 將 AWS IoT 政策連接到 Amazon Cognito Identity，並提供精細的許可給 AWS IoT 應用程式的個別使用者。如此一來，您即可將許可指派給特定客戶及其裝置。如需為 Amazon Cognito 身分建立政策的詳細資訊，請參閱發佈/訂閱政策範例 (p. 156)。

自訂身分驗證

AWS IoT 可讓您定義自訂授權方，以便可以管理自己的裝置身分驗證和授權。若要這樣做，您可以使用 AWS Lambda 函式，將裝置登入資料傳送至身分驗證服務。

建立 HTTP 連線 (或選擇升級為 WebSocket 連線) 且缺少 Signature Version 4 標頭時，AWS IoT 裝置閘道會檢查端點是否已設定自訂授權方。若是如此，AWS IoT 裝置閘道會使用自訂授權方來驗證連線並授權裝置。自訂授權方可以實作多種身分驗證策略 (例如 JWT 驗證、OAuth 提供者呼叫等)，也必須傳回由裝置閘道用於授權 MQTT 操作的政策文件。

實作自訂身分驗證的方式有兩種：

- 自訂身分驗證 – 使用 HTTP 發佈操作或 MQTT over WSS 連線，在 HTTP 標頭內接收承載字符 (例如 JSON Web 字符 (JWT) 或 OAuth 字符) 中的裝置登入資料。
- 增強型自訂身分驗證 – 延伸自訂身分驗證，以啟用透過 MQTT CONNECT 訊息 (在 `username` 和 `password` 欄位中) 或做為 HTTP 發佈或升級請求中的查詢參數 (若為 MQTT over WSS) 傳遞裝置登入資料。此功能也可讓您移除自訂授權方的字符簽署需求。此功能為 Beta 版。

> **Note**
>
> 您最多可以定義 10 個自訂授權方。

主題

- 自訂授權方 (p. 138)
- 設定自訂授權方 (p. 140)
- 自訂授權方工作流程 (p. 141)
- 增強型自訂身分驗證 (Beta 版) (p. 142)

自訂授權方

自訂授權方包含：

名稱

可辨識授權方的任意獨特字串。

Lambda 函數 ARN

實作驗證邏輯並回傳授權政策的 Lambda 函數 ARN。

Public key

此為金鑰對的公開金鑰，可防止向授權方的 Lambda 函數進行未經授權的呼叫。

使用下列命令來產生金鑰對：

openssl genrsa -out myKeyPair.pem 2048

使用下列命令來擷取金鑰對的公開金鑰：

openssl rsa -in myKeyPair.pem -pubout > mykey.pub

符記金鑰名稱

用於自 WebSocket 連線標頭擷取符記的金鑰名稱。

執行驗證的邏輯於 Lambda 函數進行實作。

Note

費用依 Lambda 函數的請求數量和程式碼執行持續時間計算。如需 AWS Lambda 的詳細資訊，請參閱 AWS Lambda 定價和AWS Lambda Developer Guide。

此函數需要代表裝置的符記，而且會驗證裝置，並回傳下列資訊：

isAuthenticated

確認符記是否經過驗證的布林值。若此處為 `false`，請忽略其他回應欄位。

principalId

獲授予 `policyDocuments` 中所述許可之委託人的 ID。

policyDocuments

遵循與 AWS IoT 政策相同慣例的 JSON 格式政策文件的清單。清單包含最多 10 個政策文件，每個文件包含最多 2,048 個字元。

DisconnectAfterInSecs

對 AWS IoT 閘道連線的最大持續時間 (以秒為單位)，超過此時間便中斷連線。最小值為 300 秒，最大值為 86,400 秒。

RefreshAfterInSecs

政策重新整理之間的期間。經過此期間時，AWS 政策引擎會重新評估政策文件，而 AWS IoT Device Gateway 會再次叫用 Lambda 函數以允許政策變更。最小值為 300 秒，最大值為 86,400 秒。

Context

驗證在 規則引擎 SQL 陳述式AWS IoT和 IAM/AWS IoT 政策變數提供之符記後衍生的資訊。

您必須授予許可給 AWS IoT 服務委託人，以呼叫可實作自訂身分驗證/授權邏輯的 Lambda 函數。可採用下列 CLI 命令：

aws lambda add-permission --function-name ***<lambda_function_name>*** --statement-id ***<unique_identifier_string>*** --action 'lambda:InvokeFunction' --principal iot.amazonaws.com --source-arn arn:aws:iot:***<your-aws-region>***:***<account_id>***: authorizer/***<authorizer-name>***

add-permission CLI 命令會使用下列參數：

`function-name`

您要授予叫用許可的 Lambda 函數名稱。

`statement-id`

陳述式識別符。

`action`

要授予執行許可的 Lambda 動作。

`principal`

要授予許可的委託人。

`source-arn`

自訂授權方 ARN。指定此值可確保您的 Lambda 函數僅能由預期的自訂授權方進行呼叫。

如需更多關於授予呼叫 Lambda 函數的許可，請參閱AWS Lambda 許可 。

若連線要求未納入授權方資訊，您可以預設授權方：

aws iot set-default-authorizer --authorizer-name ***<my-authorizer>***

設定自訂授權方

1. 建立 Lambda 函數來實作您的身分驗證/授權邏輯 (例如，下列步驟中的 `MyAuthorizerFunction`)。以下範例說明自訂授權 Lambda 函數可能傳回的項目：

```
{
        "isAuthenticated": true,
        "principalId": "xxxxxxxx",
        "disconnectAfterInSeconds": 86400,
        "refreshAfterInSeconds": 300,
        "policyDocuments": [
            "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Action\":\"...\",\"Effect\":
\"Allow|Deny\",\"Resource\":\"...\"}]}"
        ],
        "context": {
            "username": "johnDoe123",
            "city": "Seattle",
            "country": "USA"
        }
    }
```

   Lambda 函數的傳回值應該類似此回應。它可以是 JSON 序列化或非序列化物件。

2. 使用 `create-authorizer` API 向 AWS IoT 註冊自訂授權方。

```
aws iot create-authorizer --authorizer-name MyAuthorizer
        --authorizer-function-arn arn:aws:lambda:us-
west-2:<account_id>:function:MyAuthorizerFunction   // Lambda ARN
        --token-key-name MyAuthorizerToken                              // Key use to
 extract token from headers
        --token-signing-public-keys FIRST_KEY=                          // Public key
 used to verify token signature
    "-----BEGIN PUBLIC KEY-----
    [...insert your public key here...]
    -----END PUBLIC KEY-----"
        --status ACTIVE                                                 // Authorizer
 status - must be ACTIVE
```

```
        --region us-west-2                                          // AWS region
        --endpoint https://us-west-2.iot.amazonaws.com              // IoT
 endpoint
```

您可使用 `test-invoke-authorizer` API 測試自訂授權方 Lambda 函數是否設定正確，如下所示：

```
aws iot test-invoke-authorizer --authorizer-name <NAME_OF_AUTHORIZER> --token
 <TOKEN_VALUE> --token-signature <TOKEN_SIGNATURE>
```

> **Note**
>
> *<TOKEN_SIGNATURE>* 必須以公有/私有金鑰對中的私有金鑰進行簽署，且該金鑰對已上傳至 `create-authorizer` 呼叫所使用的 AWS IoT。從類似 UNIX 命令列在本機建立 *<TOKEN_SIGNATURE>* 的一項方法如下：
>
> ```
> echo -n <TOKEN_VALUE> | openssl dgst -sha256 -sign <PRIVATE_KEY> | openssl
> base64
> ```
>
> 您必須從此命令的結果中截去所有換行符號，才能將 *<TOKEN_SIGNATURE>* 值傳遞至 `test-invoke-authorizer` API。

自訂授權方工作流程

若裝置欲使用自訂授權方向 AWS IoT 裝置閘道進行身分驗證，將需要 AWS 用於驗證符記的符記和簽章，方能呼叫授權方。

當裝置嘗試連接至 AWS IoT，將傳送下列資訊至 HTTP 標頭：

- 您身分驗證服務產生的符記。
- 您身分驗證服務產生的簽章。
- 用於驗證符記的授權方。若省略，則會使用預設的授權方。

以下為 HTTP 要求透過 WebSocket 通訊協定連接至 AWS IoT 的範例。

```
GET /mqtt HTTP/1.1
    Host: <your-iot-endpoint>
    Upgrade: WebSocket
    Connection: Upgrade
    x-amz-customauthorizer-name: <authorizer-name>
    x-amz-customauthorizer-signature: <token-signature>
    <token-key-name>: <some-token>
    sec-WebSocket-Key: <any random base64 value>
    sec-websocket-protocol: mqtt
    sec-WebSocket-Version: <websocket version>
```

在此範例中，`x-amz-customauthorizer-name` 標頭指定欲使用的自訂授權方、`x-amz-customauthorizer-signature` 標頭包含用於身分驗證符記的數位簽章，而`token-key-name` 是由傳送至 `create-authorizer` API 的 `--token-key-name` 所指定的符記金鑰名稱。

> **Note**
>
> 有些 web 瀏覽器可能不支援自訂 HTTP 標頭。

AWS IoT 裝置閘道會驗證數位簽章，若有效，將呼叫指定的授權方。以下為 AWS IoT 將承載傳送至自訂驗證方的 Lambda 函數之範例。

```
{
"token": "some-token"
}
```

授權方驗證字符，並回傳委託人 ID、其相關 AWS IoT/IAM 政策、連線的存活期 (TTL) 資訊。

以下為自訂授權方的回應範例。

```
{
     "isAuthenticated":true,
     "principalId": "xxxxxxxx",
     "disconnectAfterInSeconds": 86400,
     "refreshAfterInSeconds", 300,
     "policyDocuments": [
      "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Action\": \"...\", \"Effect\":
 \"Allow|Deny\", \"Resource\": \"...\" } ] }"
     ]
    }
```

Lambda 函數的傳回值應類似於此回應，且可能為 JSON 序列化或非序列化物件。

AWS IoT裝置閘道接著會建立 WebSocket 連線。AWS IoT 會針對委託人相關的政策進行快取，因此後續呼叫無須重新驗證裝置即可獲得授權。自訂身分驗證若發生任何故障，將導致身分驗證失敗並終止連線。

如需此工作流程的端對端範例，請參閱如何使用 Identity and Access Management 系統來控制對 AWS IoT 資源的存取。

增強型自訂身分驗證 (Beta 版)

此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

增強型自訂身分驗證可在自訂授權方由可設定端點實作時，自訂授權方可以使用的身分驗證方法中提供更大的彈性。此預覽功能集可讓您除了標頭之外，還使用 HTTP 查詢字串傳遞字符給您的授權方。它也可讓您選擇是否簽署字符。這些功能也可透過在 MQTT CONNECT 訊息的 `username` 和 `password` 欄位中傳遞登入資料來啟用身分驗證。本主題描述如何使用這些增強功能，以及如何將自訂授權方新增至您的可設定端點。如需詳細資訊，請參閱可設定端點 (Beta 版)。

增強型自訂身分驗證會使用下列其中一種方法，將裝置登入資料傳送至您的自訂授權方：

- HTTP 發佈 (標頭欄位)
- HTTP 發佈 (查詢字串)
- MQTT 連線 (使用者名稱和密碼欄位)
- HTTP 升級至 MQTT (HTTP 標頭或查詢字串)
- HTTP 升級至 MQTT (MQTT 使用者名稱和密碼)

> **Note**
>
> 若要傳遞未簽署的字符或使用 MQTT 使用者名稱/密碼身分驗證，裝置必須傳送伺服器名稱指示 (SNI) TLS 延伸，並搭配對應於網域組態中所指定之名稱的完整網域名稱。若要測試此服務，請使用 GitHub 中每個 AWS IoT 裝置 SDK 的 v2 版本。

主題

- 為增強型自訂身分驗證建立自訂授權方 (p. 143)

- 叫用具有增強型自訂身分驗證的自訂授權方 (p. 143)

為增強型自訂身分驗證建立自訂授權方

此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

您可以使用您用來建立自訂授權方 API 以進行自訂身分驗證的相同 CreateAuthorizer API，建立自訂授權方 API 以進行增強型自訂身分驗證。如需建立自訂授權方的程序，請參閱設定自訂授權方。

您可以使用 `CreateAuthorizer` API 的 `signingDisabled` 參數。這可讓您 AWS IoT 針對 HTTP 請求中提供的字符選擇退出簽章驗證。強烈建議您只在測試環境中使用此功能。您無法更新自訂授權方的 `signingDisabled` 值。若要變更此行為，您必須使用不同的 `signingDisabled` 值建立自訂授權方。`tokenKeyName` 和 `tokenSigningPublicKeys` 的值是選用的，但是當您指定 `tokenSigningPublicKeys` 的值時，`tokenKeyName` 是必要的。

將自訂授權方新增至可設定端點

在建立您的自訂授權方之後，您可以使用 UpdateDomainConfiguration API，將它附加到現有的網域組態做為預設授權方。您也可以使用 CreateDomainConfiguration API，並在 `defaultAuthorizerName` 參數中指定自訂授權方的名稱來建立組態。

叫用具有增強型自訂身分驗證的自訂授權方

此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

本節描述如何使用增強型自訂身分驗證，使用 HTTP 標頭和查詢字串或 MQTT 使用者名稱和密碼，將裝置登入資料傳遞給您的自訂授權方。

使用增強型自訂身分驗證的裝置必須傳送伺服器名稱指示 (SNI) TLS 延伸，其值符合適當網域組態的網域。如果您的網域組態中的 `allowAuthorizerOverride` 值設為 `true`，您可以指定不是預設授權方的自訂授權方。

若要在 MQTT 連線中使用增強型自訂身分驗證，裝置也須傳送應用程式層通訊協定交涉 (ALPN) TLS 延伸 (其值為 `mqtt`)，並在連接埠 443 上進行連線。

> **Note**
>
> 您可以在 V2 AWS IoT 裝置 SDK 中找到這些 TLS 延伸的組態。如需詳細資訊，請參閱 AWSLabs GitHub 儲存庫。

HTTPS 請求

自訂授權方的請求可以使用下列其中一項來傳遞字符：HTTP 發佈請求或 HTTP 升級請求中的 HTTP 標頭或查詢字串，用於建立 MQTT over WSS 工作階段。下列範例會使用 HTTP 標頭將升級請求傳送至 AWS IoT 裝置閘道。

```
GET /mqtt HTTP/1.1
Host: your-endpoint
Upgrade: WebSocket
Connection: Upgrades
x-amz-customauthorizer-signature: token-signature
token-key-name: some-token
sec-WebSocket-Key: any random base64 value
sec-websocket-protocol: mqtt
sec-WebSocket-Version: websocket version
```

請求不包含 x-amz-customauthorizer-name 的值，因此 AWS IoT 會使用在網域組態中指定的預設授權方。如果在授權方上停用簽署，x-amz-customauthorizer-signature 的值是選用的。

以下範例顯示如何使用查詢字串參數進行相同的請求。

```
GET /mqtt?x-amz-customauthorizer-signature=${sign}&token-name=${token-value} HTTP/1.1
Host: your-endpoint
Connection: Upgrade
Upgrade: websocket
sec-WebSocket-Key: any random base64 value
sec-websocket-protocol: mqtt
sec-WebSocket-Version: websocket version
```

MQTT 使用者名稱及密碼

您可以使用 MQTT 訊息的 username 和 password 欄位，將使用者名稱和密碼傳遞給自訂授權方。密碼資料為 base64 編碼形式，以支援任意二進位值。username 值可以選擇性地包含一個查詢字串，將其他值 (包括字符、簽章和授權方名稱) 傳遞給您的授權方。

以下範例包含一個 username 字串，其中其有額外參數來指定字符和簽章。您可以使用此方法，藉由使用承載字符來驗證 MQTT 連線。

```
username?x-amz-customauthorizer-name=${name}&x-amz-customauthorizer-signature=
${sign}&token-name=${token-value}
```

傳送給自訂授權方的資料

AWS IoT 傳送至自訂授權方 Lambda 函式的資料取決於連線中存在的通訊協定和參數。對於 HTTP 請求 (發佈和 WSS 升級)，AWS IoT 會傳送所有標頭和查詢參數 (最多 8 KB 的資料)。如需在 MQTT 連線時進行 WSS 升級，AWS IoT 也會傳送所有 MQTT 資料。

如果已啟用簽署，並且請求包含字符和簽章，則 AWS IoT 不會觸發您自訂授權方的 Lambda 函式，除非字符簽章可以解密，以符合請求中提供的字符。您可以使用此方法，讓其他連線到您網域的人員不會過度觸發您的 Lambda 函式。

> **Important**
>
> 強烈建議您使用 AWS IoT 提供的簽章驗證功能。

以下是 AWS IoT 傳送至自訂驗證方 Lambda 函式的範例承載。AWS IoT 只會填入請求包含的欄位。

```
{
    "token" :"aToken",
    "signatureVerified": boolean, // Indicates whether the device gateway has validated the
 signature.
    "protocols": ["tls", "http", "mqtt"], // Indicates which protocols to expect.
    "protocolData": {
        "tls" : {
            "serverName": "serverName" // The SNI string.
        },
        "http": {
            "headers": {
                "#{name}": "#{value}"
            },
            "queryString": "?#{name}=#{value}"
        },
        "mqtt": {
```

```
            "username": "myUserName",
            "password": "myPassword", // base64 encoded.
            "clientId": "myClientId" // Provided only when the device sends it.
        }
    },
    "connectionMetadata": {
        "id": UUID // The connection ID. You can use this for logging.
    },
}
```

> **Note**
>
> 所有密碼資料都是 base64 編碼形式，因此你的 Lambda 函式需要將其解碼。

與一般自訂身分驗證一樣，自訂授權方必須傳回回應，指出是否已授權 AWS IoT 的請求。以下為自訂授權方的成功回應範例。

```
{
     "isAuthenticated":true,
     "principalId": "xxxxxxxx",
     "disconnectAfterInSeconds": 86400,
     "refreshAfterInSeconds", 300,
     "policyDocuments": [
      "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Action\": \"...\", \"Effect\":
 \"Allow|Deny\", \"Resource\": \"...\" } ] }"
     ]
    }
```

在 `refreshAfterInSeconds` 的值中指定的持續時間內，AWS IoT 會快取與委託人相關聯的政策。在此持續時間內，後續呼叫會被獲授權，而不必重新驗證裝置。自訂身分驗證期間發生的失敗會導致身分驗證失敗，因而停止連線。

管理裝置憑證

您的裝置可以使用 X.509 憑證向 AWS IoT Core 進行身分驗證。

伺服器驗證

AWS IoT 根 CA 憑證可讓您的裝置驗證它們是否正在與 AWS IoT Core 通訊，而不是與另一部模擬 AWS IoT Core 的伺服器通訊。如需詳細資訊，請參閱用於服務驗證的 CA 憑證。

授權

授權是授與許可給已驗證身分的程序。在 AWS IoT 中可以使用 AWS IoT 和 IAM 政策來授與許可。本主題涵蓋 AWS IoT 政策。如需 IAM 政策的詳細資訊，請參閱適用於 AWS IoT 的 Identity and Access Management (p. 179)和IAM 政策 (p. 183)。

AWS IoT 政策會判定已驗證的身分可執行的動作。裝置、行動應用程式、Web 應用程式和桌面應用程式，都會使用未驗證的身分，未驗證身分的使用者甚至能夠輸入 AWS IoT CLI 命令。身分擁有授予執行 AWS IoT 操作之許可的政策時，才能執行這些操作。

AWS IoT 會使用 IAM 政策與 AWS IoT 政策，以控制某身分 (亦稱委託人) 可執行的操作。您使用的政策類型，取決於您向 AWS IoT 驗證所使用的身分類型。

AWS IoT 的運作分成兩組：

- 控制平面 API 可讓您執行管理任務，例如建立或更新憑證、物件、規則等。
- 資料平面 API 可讓您將資料傳送至 AWS IoT 並從此接收資料。

您使用的政策類型，取決於您正使用控制平面或資料平面 API。

下表說明身分類型、其使用的通訊協定以及可用於授權的政策類型。

AWS IoT 資料平面 API 和政策類型

通訊協定和身分驗證機制	SDK	身分類型	政策類型		
透過交互身分驗證的 MQTT (連接埠 8883 或 443[†] (p. 242))	AWS IoT裝置 SDK	X.509 憑證	AWS IoT 政策		
經 WebSocket 的 MQTT (連接埠 443)	AWS Mobile SDK	Amazon Cognito、IAM 或聯合身分	已驗證 Amazon Cognito 身分的 AWS IoT 政策 適用其他身分的 IAM 政策		
經伺服器驗證的 HTTP (連接埠 443)	AWS CLI	Amazon Cognito、IAM 或聯合身分	已驗證 Amazon Cognito 身分的 AWS IoT 政策 適用其他身分的 IAM 政策		
經交互身分驗證的 HTTP (連接埠 8443)	無 SDK 支援項目	X.509 憑證	AWS IoT 政策		

AWS IoT 控制平面 API 和政策類型

通訊協定和身分驗證機制	SDK	身分類型	政策類型		
經伺服器驗證的 HTTP (連接埠 443)	AWS CLI	Amazon Cognito、IAM 或聯合身分	已驗證 Amazon Cognito 身分的 AWS IoT 政策 適用其他身分的 IAM 政策		

AWS IoT 政策連接到 X.509 憑證或 Amazon Cognito 身分。IAM 政策連接到 IAM 使用者、群組或角色。若您透過 AWS IoT 主控台或 AWS IoT CLI 來連接政策 (至憑證或 Amazon Cognito Identity)，請使用 AWS IoT 政策。否則，請使用 IAM 政策。

政策型授權是一項強大工具。能夠讓您完全控制裝置、使用者或應用程式可在 AWS IoT 執行的動作。例如，假設一個裝置透過憑證連接至 AWS IoT。您可以允許該裝置存取所有 MQTT 主題，或者限制其存取單一主題。以另一個範例而言，假設一名使用者於命令列輸入 CLI 命令。您可以透過政策，允許或拒絕該使用者存取命令或 AWS IoT 資源。您也可以控制應用程式存取 AWS IoT 資源。

AWS IoT 政策

AWS IoT 政策為 JSON 文件。他們遵循相同的慣例作為 IAM 政策。AWS IoT 支援具名政策，因此許多身分可以參考相同的政策文件。具名政策會經過版本控制，因此可輕鬆還原。

AWS IoT 政策可讓您控制 AWS IoT 資料平面的存取。AWS IoT 資料層包含您可進行的操作，像是連線至 AWS IoT 訊息中介裝置、傳送及接收 MQTT 訊息，以及取得或更新裝置的影子。

AWS IoT 政策為包含一個或多個政策陳述式的 JSON 文件。每個陳述式都包含：

- `Effect`，指定是否允許或拒絕該動作。
- `Action`，指定政策允許或拒絕的動作。
- `Resource`，指定資源或動作可用或不可用的資源。

AWS IoT 政策動作

以下為 AWS IoT 定義的政策動作：

MQTT 政策動作

iot:Connect

代表連接至 AWS IoT 訊息中介裝置的許可。中介裝置每次接收 `iot:Connect` 要求，就會檢查 `CONNECT` 許可。訊息中介裝置不允許兩個相同 ID 的用戶端同時在線。第二個用戶端連線之後，中介裝置會關閉現有的連線。可使用 `iot:Connect` 許可，以確保僅已授權的用戶端可使用特定用戶端 ID 來連線。

iot:Publish

代表於 MQTT 主題的發佈許可。中介裝置每次接收 PUBLISH 要求，就會檢查本許可。這可用來允許用戶端發佈至特定主題模式。

> **Note**
>
> 您必須也授予 `iot:Connect` 的許可，方能授予 `iot:Publish` 的許可。

iot:Receive

代表自 AWS IoT 接收訊息的許可。訊息每次傳送至用戶端，就會檢查 `iot:Receive`許可。由於每此交付都會檢查本許可，因此可用來撤銷目前訂閱某主題的用戶端的許可。

iot:Subscribe

代表主題篩選條件的訂閱許可。中介裝置每次接收 SUBSCRIBE 要求，就會檢查本許可。這可用來允許用戶端訂閱符合特定主題模式的主題。

> **Note**
>
> 您必須也授予 `iot:Connect` 的許可，方能授予 `iot:Subscribe` 的許可。

影子政策動作

iot:DeleteThingShadow

代表可刪除裝置影子的許可。每次發出刪除影子內容的要求時，就會檢查一次 `iot:DeleteThingShadow`許可。

iot:GetThingShadow

代表可擷取裝置影子的許可。每次發出擷取影子內容的要求時，就會檢查一次 `iot:GetThingShadow`許可。

iot:UpdateThingShadow

代表可更新裝置影子的許可。每次發出更新影子內容的要求時，就會檢查一次 `iot:UpdateThingShadow`許可。

> **Note**
>
> 工作執行政策動作僅適用於 HTTP TLS 端點。如果您使用 MQTT 端點，必須使用此主題上述所定義的 MQTT 政策動作。

工作執行 AWS IoT 政策動作

iot:DescribeJobExecution

代表可擷取指定物件之工作執行的許可。每次發出取得工作執行的要求時，就會檢查一次 `iot:DescribeJobExecution` 許可。

iot:GetPendingJobExecutions

代表可擷取工作 (對物件來說並非結束狀態) 之清單的許可。每次發出擷取清單的要求時，就會檢查一次 `iot:GetPendingJobExecutions`許可。

iot:UpdateJobExecution

代表可更新工作執行的許可。每次發出更新工作執行狀態的要求時，就會檢查一次 `iot:UpdateJobExecution`許可。

iot:StartNextPendingJobExecution

代表可取得並啟動物件之下一個待定工作執行的許可。(也就是將狀態為 QUEUED 的工作執行，更新為 IN_PROGRESS。) 每次發出啟動下一個待定工作執行的要求時，就會檢查一次 `iot:StartNextPendingJobExecution`許可。

AWS IoT 動作資源

若要指定 AWS IoT 政策動作的資源，您必須使用資源的 ARN。所有資源 ARN 的格式如下：

```
arn:aws:iot:region:AWS account ID:resource type/resource name
```

下表說明每一動作類型須指定的資源：

Action	資源
`iot:DeleteThingShadow`	物件 ARN：`arn:aws:iot:us-east-1:123456789012:thing/thingOne`
`iot:Connect`	用戶端 ID ARN：`arn:aws:iot:us-east1:123456789012:client/myClientId`
`iot:Publish`	ARN 主題：`arn:aws:iot:us-east-1:123456789012:topic/myTopicName`
`iot:Subscribe`	主題篩選 ARN：`arn:aws:iot:us-east-1:123456789012:topicfilter/myTopicFilter`

Action	資源
iot:Receive	ARN 主題：arn:aws:iot:us-east-1:123456789012:topic/myTopicName
iot:UpdateThingShadow	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne
iot:GetThingShadow	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne
iot:DescribeJobExecution	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne
iot:GetPendingJobExecutions	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne
iot:UpdateJobExecution	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne
iot:StartNextPendingJobExecution	物件 ARN：arn:aws:iot:us-east-1:123456789012:thing/thingOne

AWS IoT 政策變數

AWS IoT 定義可用於 AWS IoT 政策中 `Resource` 或 `Condition` 區塊的政策變數。當政策受到評估時，實際值就會替代原則變數。例如，若用戶端 ID 為「100-234-3456」的裝置連接至 AWS IoT 訊息中介裝置，`iot:ClientId` 政策變數將於政策文件內替代為「100-234-3456」。 如需關於政策變數的詳細資訊，請參閱 IAM 政策變數和多值條件相關文章。

基本 AWS IoT 政策變數

AWS IoT 定義的基本政策變數如下：

- `iot:ClientId`：用於連接至 AWS IoT 訊息中介裝置的用戶端 ID。
- `aws:SourceIp`：用戶端連接至 AWS IoT 訊息中介裝置的 IP 地址。

下列 AWS IoT 政策顯示使用政策變數的政策：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Connect"],
        "Resource": [
          "arn:aws:iot:us-east-1:123451234510:client/${iot:ClientId}"
        ]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Publish"],
        "Resource": [
          "arn:aws:iot:us-east-1:123451234510:topic/my/topic/${iot:ClientId}"
        ]
      }
    ]
}
```

在這些範例中，當政策受到評估時，`${iot:ClientId}` 會由用戶端連接至 AWS IoT 訊息中介裝置的 ID 所替代。當您使用諸如 `${iot:ClientId}`的原則變數，您可能會意外開放部分主題的存取。例如，若您的政策使用 `${iot:ClientId}`來指定主題篩選條件：

```
{
    "Effect": "Allow",
    "Action": ["iot:Subscribe"],
    "Resource": [
      "arn:aws:iot:us-east-1:123456789012:topicfilter/my/${iot:ClientId}/topic"
    ]
}
```

用戶端可使用 `+`做為用戶端 ID 來連接，使用者即可訂閱符合主題篩選條件 `my/+/topic` 的任何主題。若要避免這類安全漏洞，請使用 `iot:Connect` 政策動作來控制可連接的用戶端 ID。例如，此政策僅允許用戶端 ID 為 `clientid1` 的用戶端連接：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Connect"],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/clientid1"
        ]
      }
    ]
}
```

X.509 憑證 AWS IoT 政策變數

X.509 憑證原則變數讓您能夠依據 X.509 憑證屬性，撰寫許可授予的 AWS IoT 政策。以下部分說明如何使用這些憑證原則變數。

物件原則變數

物件政策變數讓您能夠依據物件屬性 (如物件名稱、物件類型、物件屬性值)，撰寫用於授予或拒絕許可的 AWS IoT 政策。當物件連接至 `Connect` 時，將自 MQTT AWS IoT訊息內的用戶端 ID 取得物件名稱。當物件透過 MQTT 使用 TLS 交互身分驗證連接至 AWS IoT，或經 WebSocket 通訊協定的 MQTT 使用已驗證的 Amazon Cognito 身分，物件政策變數會被取代。當憑證或已驗證的 Amazon Cognito Identity 連接至物件，物件政策變數亦將受到取代。您可以使用 AttachThingPrincipal API，將憑證和已驗證的 Amazon Cognito Identity 連接至物件。

現已提供下列物件原則變數：

- `iot:Connection.Thing.ThingName`

  這可以解析為 AWS IoT 登錄檔中要評估政策的實物名稱。AWS IoT 會使用裝置在驗證時提供的憑證，判斷哪些實物要用於驗證連線。本原則變數僅適用於裝置透過 MQTT 或 MQTT over WebSocket 通訊協定進行連線時。
- `iot:Connection.Thing.ThingTypeName`

  當政策受到評估時，這可以解析為所屬物件相關的物件類型。物件名稱設定為 MQTT/WebSocket 連線的用戶端 ID。藉由呼叫 `DescribeThing`API，可取得物件類型名稱。本原則變數僅供透過 MQTT 或是經 WebSocket 通訊協定的 MQTT 進行之連線所使用。
- `iot:Connection.Thing.Attributes[`*`attributeName`*`]`

  當政策受到評估時，這可以解析為所屬物件相關的特定屬性值。單一物件至多可具備 50 個屬性。每一屬性均可做為原則變數：`iot:Connection.Thing.Attributes[`*`attributeName`*`]`，其中

attributeName 為屬性名稱。物件名稱設定為 MQTT/WebSocket 連線的用戶端 ID。本原則變數僅供透過 MQTT 或是經 WebSocket 通訊協定的 MQTT 進行之連線所使用。

- `iot:Connection.Thing.IsAttached`

  若要評估政策的憑證或 Amazon Cognito 身分已連接到 IoT 事物，這可以解析為 `true`。如果裝置出示的憑證並未連接到 AWS IoT 登錄中的 IoT 實物，您可以使用此變數，防止裝置連接到 AWS IoT。

發行者屬性

下列 AWS IoT 政策變數可讓您依據憑證發行者所設定的憑證屬性，允許或拒絕許可。

- `iot:Certificate.Issuer.DistinguishedNameQualifier`
- `iot:Certificate.Issuer.Country`
- `iot:Certificate.Issuer.Organization`
- `iot:Certificate.Issuer.OrganizationalUnit`
- `iot:Certificate.Issuer.State`
- `iot:Certificate.Issuer.CommonName`
- `iot:Certificate.Issuer.SerialNumber`
- `iot:Certificate.Issuer.Title`
- `iot:Certificate.Issuer.Surname`
- `iot:Certificate.Issuer.GivenName`
- `iot:Certificate.Issuer.Initials`
- `iot:Certificate.Issuer.Pseudonym`
- `iot:Certificate.Issuer.GenerationQualifier`

主體屬性

下列 AWS IoT 政策變數可讓您依據憑證發行者所設定的憑證主體屬性，授予或拒絕許可。

- `iot:Certificate.Subject.DistinguishedNameQualifier`
- `iot:Certificate.Subject.Country`
- `iot:Certificate.Subject.Organization`
- `iot:Certificate.Subject.OrganizationalUnit`
- `iot:Certificate.Subject.State`
- `iot:Certificate.Subject.CommonName`
- `iot:Certificate.Subject.SerialNumber`
- `iot:Certificate.Subject.Title`
- `iot:Certificate.Subject.Surname`
- `iot:Certificate.Subject.GivenName`
- `iot:Certificate.Subject.Initials`
- `iot:Certificate.Subject.Pseudonym`
- `iot:Certificate.Subject.GenerationQualifier`

X.509 憑證的這些屬性可包含一個或多個值。根據預設，每個多值屬性的原則變數會回傳第一個值。例如，`Certificate.Subject.Country` 屬性可能包含國家名稱的清單，但在政策中評估時，會將 `iot:Certificate.Subject.Country` 取代為第一個國家的名稱。您可以使用開頭為零的索引，要求第

一個國家名稱以外的特定屬性值。例如，在 `iot:Certificate.Subject.Country.1`屬性中，第二個國家/地區名稱將取代 `Certificate.Subject.Country`。若您指定的索引值不存在 (例如，僅對該屬性指派兩個值但您要求第三個值)，將不會進行替換，而授權會失敗。您可以在原則變數名稱添加 `.List` 尾碼，指定屬性全部的值。

Registered Devices (2)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策允許具在 AWS IoT 登錄檔中登錄之實物名稱的用戶端進行連接，但將權力限制為將特定主題的實物名稱發佈到含以下憑證的那些用戶端：其 `Certificate.Subject.Organization` 屬性設為 `"Example Corp"` 或 `"AnyCompany"`。使用 `"Condition"` 欄位，此限制即會伴隨而來：該欄位會指定必須符合的條件才能允許先前的動作。此例的條件就是與憑證關聯的 `Certificate.Subject.Organization` 屬性必須包含其中一個所列的值：

```
{
    "Version":"2012-10-17",
    "Statement":[
      {
        "Effect":"Allow",
        "Action":[
          "iot:Connect"
        ],
        "Resource":[
          "arn:aws:iot:us-east-1:123456789012:client/${iot:Connection.Thing.ThingName}"
        ]
      },
      {
        "Effect":"Allow",
        "Action":[
          "iot:Publish"
        ],
        "Resource":[
          "arn:aws:iot:us-east-1:123456789012:topic/my/topic/
${iot:Connection.Thing.ThingName}"
        ],
        "Condition":{
          "ForAllValues:StringEquals":{
            "iot:Certificate.Subject.Organization.List":[
              "Example Corp",
              "AnyCompany"
            ]
          }
        }
      }
    ]
}
```

Unregistered Devices (2)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2` 和 `client3` 連接至 AWS IoT，但將權力限制為將 client-id 特定主題發佈到含以下憑證的那些用戶端：其 `Certificate.Subject.Organization` 屬性設為 `"Example Corp"` 或 `"AnyCompany"`。使用 `"Condition"` 欄位，此限制即會伴隨而來：該欄位會指定必須符合的條件才能允許先前的動作。此例的條件就是與憑證關聯的 `Certificate.Subject.Organization` 屬性必須包含其中一個所列的值：

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

```
          "iot:Connect"
        ],
        "Resource":[
          "arn:aws:iot:us-east-1:123456789012:client/client1",
          "arn:aws:iot:us-east-1:123456789012:client/client2",
          "arn:aws:iot:us-east-1:123456789012:client/client3"
        ]
      },
      {
        "Effect":"Allow",
        "Action":[
          "iot:Publish"
        ],
        "Resource":[
          "arn:aws:iot:us-east-1:123456789012:topic/my/topic/${iot:ClientId}"
        ],
        "Condition":{
          "ForAllValues:StringEquals":{
            "iot:Certificate.Subject.Organization.List":[
              "Example Corp",
              "AnyCompany"
            ]
          }
        }
      }
    ]
}
```

發行者別名屬性

下列 AWS IoT 政策變數可讓您依據憑證發行者所設定的發行者別名屬性，授予或拒絕許可。

- `iot:Certificate.Issuer.AlternativeName.RFC822Name`
- `iot:Certificate.Issuer.AlternativeName.DNSName`
- `iot:Certificate.Issuer.AlternativeName.DirectoryName`
- `iot:Certificate.Issuer.AlternativeName.UniformResourceIdentifier`
- `iot:Certificate.Issuer.AlternativeName.IPAddress`

主體別名屬性

下列 AWS IoT 政策變數可讓您依據憑證發行者所設定的主體別名屬性，授予或拒絕許可。

- `iot:Certificate.Subject.AlternativeName.RFC822Name`
- `iot:Certificate.Subject.AlternativeName.DNSName`
- `iot:Certificate.Subject.AlternativeName.DirectoryName`
- `iot:Certificate.Subject.AlternativeName.UniformResourceIdentifier`
- `iot:Certificate.Subject.AlternativeName.IPAddress`

其他屬性

您可以使用 `iot:Certificate.SerialNumber`，依據憑證的序號允許或拒絕 AWS IoT 資源的存取。`iot:Certificate.AvailableKeys`原則變數包含的所有憑證原則變數名稱都具備值。

X.509 憑證原則變數限制

以下限制適用於 X.509 憑證原則變數：

萬用字元

若憑證屬性出現萬用字元，憑證屬性值不會取代原則變數，政策文件會保留 `${policy-variable}`文字。如此可能導致授權失敗。

陣列欄位

具備陣列的憑證屬性僅限五個項目，其他項目會遭到忽略。

字串長度

所有字串值上限為 1024 個字元。若憑證屬性的字串超過 1024 個字元，憑證屬性值不會取代原則變數，政策文件會保留 `${policy-variable}`。如此可能導致授權失敗。

AWS IoT 政策範例

JSON 文件會指定 AWS IoT 政策。AWS IoT 政策由下列項目組成：

版本

必須設定為 `"2012-10-17"`。

效果

必須設定為 `"Allow"`或 `"Deny"`。

動作

必須設定為 "iot:*operation-name*"，其中 *operation-name* 是下列之一：

`"iot:Connect"`：連線到 AWS IoT。

`"iot:Receive"`：自 AWS IoT 接收訊息。

`"iot:Publish"`：MQTT 發佈。

`"iot:Subscribe"`：MQTT 訂閱。

`"iot:UpdateThingShadow"`：更新裝置的影子。

`"iot:GetThingShadow"`：擷取裝置的影子。

`"iot:DeleteThingShadow"`：刪除裝置的影子。

資源

必須設定為下列之一：

用戶端：`arn:aws:iot:`*region*`:`*account-id*`:client/`*client-id*

主題 ARN：`arn:aws:iot:`*region*`:`*account-id*`:topic/`*topic-name*

主題篩選條件 ARN：`arn:aws:iot:`*region*`:`*account-id*`:topicfilter/`*topic-filter*

連接政策範例

下列政策會授予許可，以使用用戶端 ID `client1` 連接到 AWS IoT：

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

```
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/client1"
        ]
      }
    ]
}
```

以下政策拒絕使用用戶端 ID `client2` 和 AWS IoT 連接到 `client1` 的許可，同時允許裝置使用與在 AWS IoT 登錄檔中登錄之實物名稱相符的用戶端 ID 來進行連接：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Deny",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/client1",
          "arn:aws:iot:us-east-1:123456789012:client/client2"
        ]
      },
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/${iot:Connection.Thing.ThingName}"
        ]
      }
    ]
}
```

Registered Devices (3)

以下政策會授予裝置使用其物件名稱做為用戶端 ID 進行連線的權限，並訂閱主題篩選條件 `my/topic/filter`。裝置必須向 AWS IoT 註冊。當裝置連線至 AWS IoT 時，裝置必須提供與 AWS IoT 登錄中的 IoT 物件所關聯的憑證：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/${iot:Connection.Thing.ThingName}"
        ]
      },
      {
        "Effect": "Allow",
        "Action": [
          "iot:Subscribe"
        ],
```

```
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic/filter"
        ]
      }
    ]
}
```

Unregistered Devices (3)

若裝置未在 AWS IoT 登錄中註冊為物件，則以下政策會使用用戶端 ID `client1` 以授與連線權限，並訂閱主題篩選條件 `my/topic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic"
            ]
        }
    ]
}
```

發佈/訂閱政策範例

您使用的政策取決於您連接至 AWS IoT 的方式。您可以使用 MQTT 用戶端、HTTP 或 WebSocket 連接至 AWS IoT。當您使用 MQTT 用戶端連接，必須使用 X.509 憑證進行驗證。當您透過 HTTP 或 WebSocket 通訊協定連接，必須使用 Signature 第 4 版和 Amazon Cognito 進行驗證。

適用 MQTT 用戶端的政策

當您指定 MQTT 用戶端的 AWS IoT 政策主題篩選條件，MQTT 萬用字元「+」和「#」視為字面字元使用。因此可能會造成意外行為。

Registered Devices (4)

若裝置已在 AWS IoT 登錄中註冊為物件，則以下政策會使用符合物件名稱的用戶端 ID 以授與連線至 AWS IoT 的權限，並僅訂閱主題篩選條件 `some/+/topic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
```

```
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/some/+/topic"
            ]
        }
    ]
}
```

Unregistered Devices (4)

若裝置未在 AWS IoT 登錄中註冊為物件，則以下政策會使用用戶端 ID `client1` 以授與連線至 AWS IoT 的權限，並僅訂閱主題篩選條件 `some/+/topic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/client1"
        ]
      },
      {
        "Effect": "Allow",
        "Action": [
          "iot:Subscribe"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:topicfilter/some/+/topic"
        ]
      }
    ]
}
```

Note

在政策中，MQTT 的萬用字元 + 將被視為文字，而非萬用字元。訂閱符合 `some/+/topic` 模式的主題篩選條件之嘗試會失敗，並導致用戶端連線中斷。

政策的資源屬性可以使用 * 做為萬用字元。例如，如果帳戶中每個裝置必須在為其單獨保留的唯一主題上發佈，請使用以下政策：

Registered Devices (5)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用與實物名稱相符的用戶端 ID 來連接至 AWS IoT，並發佈至字首為實物名稱的任何主題：

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

```
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}",
            ]
        }
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/
${iot:Connection.Thing.ThingName}/*"
            ]
        }
    ]
}
```

Unregistered Devices (5)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2` 或 `client3` 來連接至 AWS IoT，並發佈至字首為用戶端 ID 的任何主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
        }
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:ClientId}/*"
            ]
        }
    ]
}
```

主題篩選條件的結尾亦可使用萬用字元 *。使用萬用字元，可能會導致非預期權限的授予，因此您必須在小心考量後才使用他們。在以下狀況他們可能會很有用：當裝置必須訂閱包含許多不同主題的訊息時 (例如，如果裝置必須訂閱來自多個位置之溫度感應器的報告)。

Registered Devices (6)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用裝置實物名稱做為用戶端 ID 來連接至 AWS IoT，並訂閱字首為實物名稱、後接 `room`，再接任何字串的主題。(預期這些主題應是 `thing1/room1`、`thing1/room2` 等等)：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/
${iot:Connection.Thing.ThingName}/room*"
            ]
        }
    ]
}
```

Unregistered Devices (6)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2`、`client3` 來連接至 AWS IoT，並訂閱字首為用戶端 ID、後接 `room`，再接任何字串的主題。(預期這些主題應是 `client1/room1`、`client1/room2` 等等)：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/${iot:ClientId}/room*"
            ]
        }
    ]
}
```

Registered Devices (7)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用裝置的實物名稱做為用戶端 ID 來連接至 AWS IoT，和訂閱主題 `my/topic` 和 `my/othertopic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic",
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/othertopic"
            ]
        }
    ]
}
```

Unregistered Devices (7)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1`來連接至 AWS IoT，和訂閱主題 `my/topic` 和 `my/othertopic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic",
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/othertopic"
            ]
        }
    ]
}
```

Registered Devices (8)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用裝置的實物名稱做為用戶端 ID 來連接至 AWS IoT，和訂閱對該實物名稱/用戶端 ID 是唯一的主題：

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/my/topic/
${iot:Thing.ThingName}"
            ]
        }
    ]
}
```

Unregistered Devices (8)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1` 來連接至 AWS IoT，和發佈至對該用戶端 ID 是唯一的主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/my/topic/${iot:ClientId}"
            ]
        }
    ]
}
```

Registered Devices (9)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用裝置實物名稱做為用戶端 ID 來連接至 AWS IoT，並發佈至字首為該實物名稱或用戶端的任何主題 (除了結尾為 `bar` 的主題)：

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

```
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:Thing.ThingName}/*"
            ]
        },
        {
            "Effect": "Deny",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:Thing.ThingName}/bar"
            ]
        }
    ]
}
```

Unregistered Devices (9)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1` 和 `client1` 來連接至 AWS IoT，並發佈至字首為連接所用之用戶端 ID 的任何主題 (除了結尾為 `bar` 的主題)：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:ClientId}/*"
            ]
        },
        {
            "Effect": "Deny",
            "Action": [
                "iot:Publish"
            ],
```

```
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:ClientId}/bar"
            ]
        }
    ]
}
```

Registered Devices (10)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用裝置的實物名稱做為用戶端 ID 來連接至 AWS IoT。裝置可以訂閱主題 `my/topic`，但不能發佈到 *<thing-name>* `/bar`，其中 *<thing-name>* 是連接到 AWS IoT 的 IoT 實物名稱：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic"
            ]
        },
        {
            "Effect": "Deny",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/${iot:Thing.ThingName}/bar"
            ]
        }
    ]
}
```

Unregistered Devices (10)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1` 來連接至 AWS IoT，和訂閱主題 `my/topic`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic"
            ]
        }
    ]
}
```

當憑證或已驗證的 Amazon Cognito Identity 連接至物件，物件政策變數亦將受到取代。下列政策會授予許可，以使用用戶端 ID `client1` 連接到 AWS IoT，和訂閱並接收主題 `iotmonitor/provisioning/987654321098`。它也可讓憑證持有人訂閱這個主題。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish",
                "iot:Receive"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/iotmonitor/
provisioning/987654321098"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/iotmonitor/
provisioning/987654321098"
            ]
        }
    ]
}
```

適用 HTTP 和 WebSocket 用戶端的政策

對以下列操作而言，AWS IoT 使用連接至 Amazon Cognito Identity 的 AWS IoT 政策 (透過 `AttachPolicy` API)，以未驗證身分縮小連接至 Amazon Cognito Identity 集區的許可範圍。這表示 Amazon Cognito Identity 必須自連接至該集區的 IAM 角色政策取得許可，以及自連接至 Amazon Cognito Identity 的 AWS IoT 政策，透過 AWS IoT `AttachPolicy` 取得許可。

- `iot:Connect`

- `iot:Publish`
- `iot:Subscribe`
- `iot:Receive`
- `iot:GetThingShadow`
- `iot:UpdateThingShadow`
- `iot:DeleteThingShadow`

> **Note**
>
> 以其他 AWS IoT 操作或未驗證的身分而言，AWS IoT 不會針對連接至 Amazon Cognito Identity Pool 角色的許可，縮小範圍。對於已驗證和未驗證兩者的身分而言，這是我們針對連接至 Amazon Cognito 集區角色的政策，所提供的最寬容建議。

HTTP

若要允許未驗證 Amazon Cognito 身分透過 HTTP 對 Amazon Cognito Identity 特定的主題發佈訊息，請將下列政策連接至 Amazon Cognito Identity 集區角色：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish",
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${cognito-
identity.amazonaws.com:sub}"]
        }
    ]
}
```

若要允許已驗證的使用者，請使用 AWS IoT AttachPrincipalPolicy API，將先前政策連接到 Amazon Cognito Identity 集區角色和 Amazon Cognito Identity。

> **Note**
>
> 授權 Amazon Cognito 身分時，AWS IoT 將會同時考慮這些政策和授予指定的最低權限。只有在兩個政策都允許要求的動作時，才會允許執行動作。如果兩個政策都禁止動作，則該動作為未授權。

MQTT

若要允許未驗證 Amazon Cognito 身分透過 WebSockets 對您帳戶中 Amazon Cognito Identity 特定的主題發佈 MQTT 訊息，請將下列政策連接至 Amazon Cognito Identity 集區角色：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish",
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${cognito-
identity.amazonaws.com:sub}"]
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect",
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${cognito-
identity.amazonaws.com:sub}"]
        }
    ]
}
```

若要允許已驗證的使用者，請使用 AWS IoT AttachPrincipalPolicy API，將先前政策連接到 Amazon Cognito Identity 集區角色和 Amazon Cognito Identity。

> **Note**
>
> 授權 Amazon Cognito 身分時，AWS IoT 將會同時考慮這些政策和授予指定的最低權限。只有在兩個政策都允許要求的動作時，才會允許執行動作。如果兩個政策都禁止動作，則該動作為未授權。

接收政策範例

Registered Devices (11)

對於登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以連接至其用戶端 ID 與實物名稱相符的 AWS IoT，和訂閱並接收 `my/topic` 主題的訊息：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Receive"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/my/topic"
            ]
        }
    ]
}
```

Unregistered Devices (11)

對於未登錄為 AWS IoT 登錄檔中實物的裝置，以下政策會授予許可，以使用用戶端 ID `client1` 連接至 AWS IoT，和訂閱並接收某主題的訊息：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/client1"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/my/topic"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Receive"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/my/topic"
            ]
        }
    ]
}
```

連線和發佈政策範例

對於登錄為 AWS IoT 登錄檔中實物的裝置，下列政策會授予許可來連接至其用戶端 ID 與實物名稱相符的 AWS IoT，並將裝置限制為在用戶端 ID 或物件名稱特定的 MQTT 主題上發佈。若要讓連線成功，您必須使用實物中連接的身分或委託人，在 AWS IoT 登錄檔中登錄實物名稱並進行驗證：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action":["iot:Publish"],
        "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/
${iot:Connection.Thing.ThingName}"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Connect"],
        "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
      }
    ]
}
```

對於未登錄為 AWS IoT 登錄檔中實物的裝置，下列政策會授予許可以使用用戶端 ID `client1` 連接至 AWS IoT，並將裝置限制為在用戶端 ID 特定的 MQTT 主題上發佈：

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

```
      {
        "Effect": "Allow",
        "Action":["iot:Publish"],
        "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${iot:clientId}"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Connect"],
        "Resource": ["arn:aws:iot:us-east-1:123456789012:client/client1"]
      }
    ]
}
```

憑證政策範例

對於在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以連接至其用戶端 ID 與實物名稱相符的 AWS IoT，並發佈至其名稱同等於驗證裝置本身所用的憑證 `certificateId`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${iot:CertificateId}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        }
    ]
}
```

對於未在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2` 和 `client3` 連接至 AWS IoT，並發佈至其名稱同等於驗證裝置本身所用的憑證 `certificateId` 之主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/${iot:CertificateId}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
```

```
        }
    ]
}
```

對於在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以連接至其用戶端 ID 與實物名稱相符的 AWS IoT，並發佈至其名稱同等於驗證裝置本身所用的憑證主體 `CommonName` 欄位：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/
${iot:Certificate.Subject.CommonName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        }
    ]
}
```

> **Note**
>
> 在這個範例中，憑證的主體通用名稱欄位會用作為主題識別符，並假設主體通用名稱對每個登錄憑證是唯一的。如果憑證在多個裝置間共用，所有共用此憑證之裝置的主體通用名稱都是相同的，因此允許從多個裝置對相同主題的發佈權限 (不建議)。

對於未在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2` 和 `client3` 連接至 AWS IoT，並發佈至其名稱同等於驗證裝置本身所用的憑證主體 `CommonName` 欄位之主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/
${iot:Certificate.Subject.CommonName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
        }
```

```
    ]
}
```

Note

在這個範例中，憑證的主體通用名稱欄位會用作為主題識別符，並假設主體通用名稱對每個登錄憑證是唯一的。如果憑證在多個裝置間共用，所有共用此憑證之裝置的主體通用名稱都是相同的，因此允許從多個裝置對相同主題的發佈權限 (不建議)。

對於在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以連接至其用戶端 ID 與實物名稱相符的 AWS IoT，並在驗證裝置所用的憑證將其 `Subject.CommonName.2` 欄位設為 `Administrator` 時，發佈至其名稱字首為 `admin/` 的主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/admin/*"],
            "Condition": {
                "StringEquals": {
                    "iot:Certificate.Subject.CommonName.2": "Administrator"
                }
            }
        }
    ]
}
```

對於未在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2`和 `client3` 連接至 AWS IoT，並在驗證裝置所用的憑證將其 `Subject.CommonName.2` 欄位設為 `Administrator` 時，發佈至其名稱字首為 `admin/` 的主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
```

```
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/admin/*"],
            "Condition": {
                "StringEquals": {
                    "iot:Certificate.Subject.CommonName.2": "Administrator"
                }
            }
        }
    ]
}
```

對於在 AWS IoT 登錄檔中登錄的裝置，以下政策可讓裝置在驗證裝置所用的憑證將其 `Subject.CommonName` 欄位設為 `Administrator` 時，使用其實物名稱在包含 `admin/` 並後接 `ThingName` 的特定主題上發佈：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/admin/
${iot:Connection.Thing.ThingName}"],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "iot:Certificate.Subject.CommonName.List": "Administrator"
                }
            }
        }
    ]
}
```

對於未在 AWS IoT 登錄檔登錄的裝置，以下政策會授予許可，以使用用戶端 ID `client1`、`client2` 和 `client3` 連接至 AWS IoT，並在驗證裝置所用的憑證將其中一個 `Subject.CommonName` 欄位設為 `Administrator` 時，發佈至主題 `admin`：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1",
                "arn:aws:iot:us-east-1:123456789012:client/client2",
                "arn:aws:iot:us-east-1:123456789012:client/client3"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/admin"],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "iot:Certificate.Subject.CommonName.List": "Administrator"
                }
            }
        }
    ]
}
```

物件政策範例

若向 AWS IoT 驗證的憑證連接至某物件且其政策正受評估，則下列政策允許裝置進行連線。

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":["iot:Connect"],
            "Resource":[ "*" ],
            "Condition": {
                "Bool": {
                    "iot:Connection.Thing.IsAttached": ["true"]
                }
            }
        }
    ]
}
```

使用 Amazon Cognito 身分的授權

有兩種類型的 Amazon Cognito 身分：未驗證和已驗證。當您的應用程式支援未驗證 Amazon Cognito 身分時，不會執行驗證，因此您不知道使用者是誰。對於未驗證的使用者，您可以將 IAM 角色連接至未驗證的身分集區來授與許可。您應該只授與您想要提供給未知使用者之資源的存取權。

當您的應用程式支援已驗證 Amazon Cognito 身分時，您可以在兩個位置指定政策。您可以將 IAM 政策連接到已驗證 Amazon Cognito Identity 集區，並將 AWS IoT 政策連接到 Amazon Cognito Identity。您可以在建立 Amazon Cognito Identity 集區時，使用 Amazon Cognito Identity 主控台將 IAM 政策連接到 Amazon Cognito Identity 集區。若要將 AWS IoT 政策連接至 Amazon Cognito Identity，您必須定義呼叫 `AttachPolicy` 的 Lambda 函數。

授權直接呼叫 AWS 服務

裝置會透過 TLS 交互身分驗證通訊協定，使用 X.509 憑證連接至 AWS IoT。其他 AWS 服務不支援憑證型身分驗證，但可透過 AWS 登入資料 (AWS Signature Version 4 格式) 進行呼叫。簽章版本 4 演算法通常會要求呼叫者提供存取金鑰 ID 和私密存取金鑰。AWS IoT 有登入資料供應商，可讓您使用內建的 X.509 憑證做為唯一的裝置身分以驗證 AWS 請求。這樣就不需要在裝置上儲存存取金鑰 ID 與私密存取金鑰。

登入資料提供者會使用 X.509 憑證來對發起人進行身分驗證並發出有限權限的暫時性安全字符。字符可用於簽署並對任何 AWS 請求進行身分驗證。這個對 AWS 請求進行身分驗證的方法需要您建立並設定 AWS Identity and Access Management (IAM) 角色並附加適當的 IAM 政策到角色，讓登入資料提供者代表您擔任該角色。如需 AWS IoT 和 IAM 的詳細資訊，請參閱「適用於 AWS IoT 的 Identity and Access Management (p. 179)」。

下圖說明登入資料提供者的工作流程。

1. AWS IoT 裝置對登入資料提供者發出 HTTPS 請求，以取得安全性字符。請求包含用於身分驗證的裝置 X.509 憑證。
2. 登入資料提供者將請求轉發給 AWS IoT 身分驗證與授權模組來驗證憑證有效性，並確認裝置具有請求安全性字符的許可。
3. 如果憑證有效且具有請求安全性字符的權限，AWS IoT 身分驗證與授權模組將傳回成功。否則，它會傳送例外到裝置。
4. 在成功驗證憑證後，登入資料提供者會叫用 AWS Security Token Service (AWS STS) 來擔任您為該憑證建立的 IAM 角色。
5. AWS STS 會傳回暫時性且僅有限權限的安全性字符給登入資料提供者。
6. 登入資料提供者傳回安全性字符到裝置。
7. 裝置使用安全性字符來簽署含有 AWS Signature Version 4 的 AWS 請求。
8. 請求的服務會叫用 IAM 來驗證簽章有效性並根據附加到您為登入資料提供者建立的 IAM 角色的存取政策來授權請求。
9. 如果 IAM 成功驗證簽章有效性並授權請求，表示請求成功。否則，IAM 會傳送例外狀況。

下列章節說明如何使用憑證來取得安全性字符。該流程假設您已經註冊裝置且已建立並啟用自己的憑證。

如何使用憑證來取得安全性字符

1. 設定登入資料提供者代表您的裝置所擔任的 IAM 角色。將下列信任政策連接至該角色。

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

```
        "Effect": "Allow",
        "Principal": {"Service": "credentials.iot.amazonaws.com"},
        "Action": "sts:AssumeRole"
    }
}
```

對於每個您想要呼叫的 AWS 服務，皆連接存取政策到該角色。登入資料提供者支援以下政策變數：

- `credentials-iot:ThingName`
- `credentials-iot:ThingTypeName`
- `credentials-iot:AwsCertificateId`

當裝置提供其對 AWS 服務的請求中的物件名稱時，登入資料提供者新增 `credentials-iot:ThingName` 與 `credentials-iot:ThingTypeName` 做為上下文變數到安全性字符中。登入資料提供者會提供 `credentials-iot:AwsCertificateId` 做為上下文變數，即使裝置在請求中未提供物件名稱。您以 `x-amzn-iot-thingname` HTTP 請求標頭的值來傳遞物件名稱。

這三個變數僅適用於 IAM 政策，不適用於 AWS IoT 政策。

2. 請確認執行下一步驟 (建立角色別名) 的使用者擁有傳遞這個新建角色到 AWS IoT 的權限。下列政策會提供 `iam:GetRole` 與 `iam:PassRole` 權限給 AWS 使用者。`iam:GetRole` 許可讓使用者取得有關您剛建立的角色資訊。`iam:PassRole` 許可可讓使用者傳遞角色到另一個 AWS 服務。

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "iam:GetRole",
            "iam:PassRole"
        ],
        "Resource": "arn:aws:iam::your aws account id:role/your role name"
    }
}
```

3. 建立 AWS IoT 角色別名。將直接呼叫 AWS 服務的裝置，在連接至 AWS IoT 時，必須知道該使用何角色 ARN。硬編碼的角色 ARN 不是好的解決方案，因為您需要隨角色 ARN 變更而更新裝置。更好的解決方案是使用 `CreateRoleAlias` API 來建立指向該角色 ARN 的角色別名。如果角色 ARN 變更時，您只需更新角色別名。無需在裝置上進行變更。此 API 會使用下列參數：

   `roleAlias`

   強制性。辨識角色別名的任意字串。可做為在角色別名資料模型中的主要金鑰功能。它包含 1-128 字元且必須僅包含英數字元和 =、@ 和 - 符號。可使用大寫和小寫字母字元。

   `roleArn`

   強制性。角色別名所指的角色之 ARN。

   `credentialDurationInSeconds`

   選用。登入資料有效期間 (秒)。最低值為 900 秒 (15 分鐘)。最高值為 3,600 秒 (60 分鐘)。預設值為 3,600 秒。

   如需此 API 的詳細資訊，請參閱 CreateRoleAlias。

4. 將政策連接至裝置憑證。連接至本裝置憑證的政策，必須授予裝置承擔該角色的許可。您需要授予角色別名 `iot:AssumeRoleWithCertificate` 動作的許可，如下列範例。

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

```
    "Statement": {
        "Effect": "Allow",
        "Action": "iot:AssumeRoleWithCertificate",
        "Resource": "arn:aws:iot:your region:your_aws_account_id:rolealias/your role
 alias"
    }
}
```

5. 對登入資料提供者發出 HTTPS 請求，以取得安全性字符。請提供下列資訊：

   - 憑證：因為這是一個對 TLS 共同身分驗證的 HTTP 請求，您提出請求時需提供憑證與私有金鑰給用戶端。使用您在向 AWS IoT 註冊憑證時所使用之相同憑證與私有金鑰。

     為確保您的裝置是與 AWS IoT (而非冒充的服務) 通訊，請參閱伺服器驗證，點選所列連結下載相應的 CA 憑證，然後將其複製到您的裝置。
   - RoleAlias：您為登入資料提供者所建立的角色別名的名稱。
   - ThingName：您在註冊 AWS IoT 物件時建立的物件名稱。這會以 `x-amzn-iot-thingname` HTTP 標頭的值來傳遞。只有在您於 AWS IoT 或 IAM 政策使用物件屬性做為政策變數時才需要此值。

   在 AWS CLI 中執行下列命令來為您的 AWS 帳戶取得登入資料提供者端點。如需此 API 行為的詳細資訊，請參閱 DescribeEndpoint。

   aws iot describe-endpoint --endpoint-type iot:CredentialProvider

   以下 JSON 物件為 describe-endpoint 指令的範例輸出。它包含用來要求安全字符的 `endpointAddress`。

```
{
    "endpointAddress": "your_aws_account_specific_prefix.credentials.iot.your
 region.amazonaws.com"
}
```

   使用端點對登入資料提供者發出 HTTPS 請求，以傳回安全性字符。下列範例命令使用 `curl`，但您可以使用任何 HTTP 用戶端。

   curl --cert ***your certficate*** --key ***your device certificate key pair*** -H "x-amzn-iot-thingname: ***your thing name***" --cacert AmazonRootCA1.pem https://***your endpoint*** /role-aliases/***your role alias***/credentials

   此命令會傳回一個包含 `accessKeyId`、`secretAccessKey`、`sessionToken` 以及一個過期情況的安全性字符物件。以下 JSON 物件為 `curl` 指令的範例輸出。

```
    {"credentials":{"accessKeyId":"access key","secretAccessKey":"secret access
 key","sessionToken":"session token","expiration":"2018-01-18T09:18:06Z"}}
```

   然後，您可以使用 `accessKeyId`、`secretAccessKey`、和 `sessionToken` 值來簽署請求到 AWS 服務。如需端對端示範，請參閱 AWS 安全部落格上的如何使用 AWS IoT 登入資料提供者來免除裝置中的硬編碼 AWS 登入資料部落格文章。

跨帳戶存取

AWS IoT 可讓您使用委託人來發佈或訂閱主題，且該主題必須由非委託人所有的 AWS 帳戶來定義。您可以建立 IAM 政策和 IAM 角色，藉此設定跨帳戶存取，並將政策連接至該角色。

首先，如建立 IAM 政策中所述建立客戶受管的 IAM 政策，如同為 AWS 帳戶中的其他使用者和憑證所做的一樣。

對於在 AWS IoT 登錄檔中登錄的裝置，以下政策會授予許可，讓裝置使用與裝置實物名稱相符的用戶端 ID 來連接至 AWS IoT，並發佈至 `my/topic/<thing-name>` ，其中 `<thing-name>` 為裝置的實物名稱：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/my/topic/
${iot:Connection.Thing.ThingName}"],
        }
    ]
}
```

對於未在 AWS IoT 登錄檔中登錄的裝置，以下政策會為裝置授予許可，以使用帳戶 (123456789012) AWS IoT 登錄檔中登錄的實物名稱 `client1` 來連接到 AWS IoT，和發佈其名稱字首為 `my/topic/` 的用戶端 ID 特定主題：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/my/topic/
${iot:ClientId}"]
        }
    ]
}
```

對於未在 AWS IoT 登錄檔中登錄的裝置，以下政策會為裝置授予許可，以使用帳戶 (123456789012) AWS IoT 登錄檔中登錄的實物名稱 `client1` 來連接到 AWS IoT，和發佈其名稱字首為 `my/topic/` 的用戶端 ID 特定主題：

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

```
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:client/client1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/my/topic/
${iot:ClientId}"]
        }
    ]
}
```

接著，依照建立角色以將許可委派給 IAM 使用者中的步驟。輸入您欲共用存取的 AWS 帳戶之帳戶 ID。最後一個步驟是，將政策連接至您剛建立的角色。若稍後您需要修改授予存取的 AWS 帳戶 ID，您可以使用下列信任政策的格式：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": { "AWS": "arn:aws:iam:us-east-1us-
east-1:567890123456:user:MyUser" },
            "Action": "sts:AssumeRole",
        }
    ]
}
```

AWS IoT Core 的資料保護

AWS IoT 會遵循 AWS 共同責任模型，此模型包含資料保護的法規和指導。AWS 會負責保護執行所有 AWS 服務的全球基礎設施。AWS 會維持對此基礎設施上託管資料的控制，包含處理客戶內容和個人資料的安全性組態控制。身為資料控制者或資料處理者的 AWS 客戶和 APN 合作夥伴都需負責保護在 AWS 雲端中放置的任何個人資料。

基於資料保護目的，我們建議您保護 AWS 帳戶登入資料，並使用 AWS Identity and Access Management (IAM) 設定個別使用者帳戶，以便每個使用者都只獲得完成其任務所需的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶都使用多重驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源通訊。

- 使用 AWS CloudTrail 設定 API 和使用者活動記錄。
- 使用 AWS 加密解決方案，以及 AWS 服務中的所有預設安全控制。
- 使用進階的受管安全服務，例如 Amazon Macie，其可協助探索和保護存放在 Amazon S3 的個人資料。

我們強烈建議您絕對不要將客戶帳戶號碼等敏感的識別資訊，放在自由格式的欄位中，如 Name (名稱) 欄位。這包括當您使用 AWS IoT 或使用主控台、API、AWS CLI 或 AWS 開發套件的其他 AWS 服務。您輸入

AWS IoT 或其他服務的任何資料都可能被選入診斷日誌中。當您提供外部伺服器的 URL 時，請勿在 URL 中包含登入資料資訊，以驗證您對該伺服器的請求。

如需資料保護的詳細資訊，請參閱「AWS 安全部落格」上的 AWS 共同責任模型和 GDPR 部落格文章。

AWS IoT 裝置會收集資料、對該資料執行一些操作，然後將該資料傳送至另一個 Web 服務。您可以選擇在裝置上短時間存放一些資料。您有責任對靜態資料提供資料保護。當您的裝置傳送資料至 AWS IoT，它會透過如本節稍後所述的 TLS 連線來進行。AWS IoT 裝置可以將資料傳送至任何 AWS 服務。如需有關每個服務之資料安全性的詳細資訊，請參閱該服務的文件。AWS IoT 可設定為將日誌寫入 CloudWatch Logs 以及將 AWS IoT API 呼叫記錄至 AWS CloudTrail。如需這些服務之資料安全的詳細資訊，請參閱 Amazon CloudWatch 的身分驗證與存取控制以及使用 AWS KMS 受管金鑰加密 CloudTrail 日誌檔案。

AWS IoT 中的傳輸安全性

AWS IoT 訊息中介裝置與 Device Shadow 服務均會使用 TLS 1.2 版加密所有通訊。TLS 可針對 AWS IoT 支援的應用程式通訊協定 (MQTT、HTTP)，確保其機密性。TLS 支援多種程式設計語言與作業系統。

以 MQTT 而言，TLS 會加密裝置和中介裝置的連線，而 AWS IoT 會使用 TLS 用戶端身分驗證來辨識裝置。以 HTTP 而言，TLS 會加密裝置和中介裝置的連線，而身分驗證則委託 AWS Signature 第 4 版完成。

TLS 密碼套件支援

AWS IoT 支援下列密碼套件：

- ECDHE-ECDSA-AES128-GCM-SHA256 (建議)
- ECDHE-RSA-AES128-GCM-SHA256 (建議)
- ECDHE-ECDSA-AES128-SHA256
- ECDHE-RSA-AES128-SHA256
- ECDHE-ECDSA-AES128-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA
- ECDHE-ECDSA-AES256-SHA
- AES128-GCM-SHA256
- AES128-SHA256
- AES128-SHA
- AES256-GCM-SHA384
- AES256-SHA256
- AES256-SHA

AWS IoT 的資料加密

資料保護是指保護傳輸中資料 (當它往返於 AWS IoT 時) 以及靜態資料 (當它存放在裝置或其他 AWS 服務中)。所有傳送到 AWS IoT 的資料都會使用 MQTT 或 HTTPS 通訊協定透過 TLS 連線傳送，因此在傳輸過程中預設是安全的。AWS IoT 裝置會收集資料，然後將資料傳送至其他 AWS 服務以便進一步處理。如需有關其他 AWS 服務上資料加密的詳細資訊，請參閱該服務的安全性文件。

Amazon FreeRTOS 提供 PKCS#11 程式庫，可抽象化金鑰儲存、存取密碼編譯物件和管理工作階段。您有責任使用此程式庫來加密裝置上的靜態資料。如需詳細資訊，請參閱 https://docs.aws.amazon.com/freertos/latest/userguide/security-pkcs.htmlAmazon FreeRTOS 公開金鑰加密標準 (PKCS) #11 程式庫

AWS IoT 中的金鑰管理

所有 AWS IoT 的連線都是使用 TLS 進行，因此初始 TLS 連線不需要用戶端加密金鑰。裝置必須使用 X.509 憑證或 Amazon Cognito Identity 來進行身分驗證。您可以讓 AWS IoT 為您產生憑證，在這種情況下，它會產生公有/私有金鑰對。如果您使用 AWS IoT 主控台，系統會提示您下載憑證和金鑰。如果您使用 `create-keys-and-certificate` CLI 命令，CLI 命令會傳回憑證和金鑰。您有責任將憑證和私有金鑰複製到您的裝置上，並保護它的安全。

適用於 AWS IoT 的 Identity and Access Management

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be authenticated (signed in) and authorized (have permissions) to use AWS IoT resources. IAM is an AWS service that you can use with no additional charge.

主題

- 對象 (p. 179)
- 使用 IAM 身分來驗證 (p. 179)
- 使用政策管理存取權 (p. 181)
- AWS IoT 如何搭配 IAM 運作 (p. 183)
- AWS IoT 身分類型政策範例 (p. 196)
- 針對 AWS IoT Identity and Access 進行故障診斷 (p. 198)

對象

How you use AWS Identity and Access Management (IAM) differs, depending on the work you do in AWS IoT.

Service user – If you use the AWS IoT service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more AWS IoT features to do your work, you might need additional permissions. Understanding how access is managed can help you request the right permissions from your administrator. If you cannot access a feature in AWS IoT, see 針對 AWS IoT Identity and Access 進行故障診斷 (p. 198).

Service administrator – If you're in charge of AWS IoT resources at your company, you probably have full access to AWS IoT. It's your job to determine which AWS IoT features and resources your employees should access. You must then submit requests to your IAM administrator to change the permissions of your service users. Review the information on this page to understand the basic concepts of IAM. To learn more about how your company can use IAM with AWS IoT, see AWS IoT 如何搭配 IAM 運作 (p. 183).

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to AWS IoT. To view example AWS IoT identity-based policies that you can use in IAM, see AWS IoT 身分類型政策範例 (p. 196).

使用 IAM 身分來驗證

在 AWS IoT 中，身分可以是裝置 (X.509) 憑證、Amazon Cognito 身分，或 IAM 使用者或群組。本主題僅討論 IAM 身分。如需 AWS IoT 支援之其他身分的詳細資訊，請參閱用戶端身分驗證 (p. 132)。

Authentication is how you sign in to AWS using your identity credentials. For more information about signing in using the AWS Management Console, see The IAM Console and Sign-in Page in the IAM User Guide.

You must be authenticated (signed in to AWS) as the AWS account root user, an IAM user, or by assuming an IAM role. You can also use your company's single sign-on authentication, or even sign in using Google or Facebook. In these cases, your administrator previously set up identity federation using IAM roles. When you access AWS using credentials from another company, you are assuming a role indirectly.

To sign in directly to the AWS Management Console, use your password with your root user email or your IAM user name. You can access AWS programmatically using your root user or IAM user access keys. AWS provides SDK and command line tools to cryptographically sign your request using your credentials. If you don’t use AWS tools, you must sign the request yourself. Do this using Signature Version 4, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 Signing Process in the AWS General Reference.

Regardless of the authentication method that you use, you might also be required to provide additional security information. For example, AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account. To learn more, see Using Multi-Factor Authentication (MFA) in AWS in the IAM User Guide.

AWS Account Root User

When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account root user and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks.

IAM 使用者和群組

An IAM user is an identity within your AWS account that has specific permissions for a single person or application. An IAM user can have long-term credentials such as a user name and password or a set of access keys. To learn how to generate access keys, see Managing Access Keys for IAM Users in the IAM User Guide. When you generate access keys for an IAM user, make sure you view and securely save the key pair. You cannot recover the secret access key in the future. Instead, you must generate a new access key pair.

An IAM group is an identity that specifies a collection of IAM users. You can't sign in as a group. You can use groups to specify permissions for multiple users at a time. Groups make permissions easier to manage for large sets of users. For example, you could have a group named IAMAdmins and give that group permissions to administer IAM resources.

Users are different from roles. A user is uniquely associated with one person or application, but a role is intended to be assumable by anyone who needs it. Users have permanent long-term credentials, but roles provide temporary credentials. To learn more, see When to Create an IAM User (Instead of a Role) in the IAM User Guide.

IAM 角色

An IAM role is an identity within your AWS account that has specific permissions. It is similar to an IAM user, but is not associated with a specific person. You can temporarily assume an IAM role in the AWS Management Console by switching roles. You can assume a role by calling an AWS CLI or AWS API operation or by using a custom URL. For more information about methods for using roles, see Using IAM Roles in the IAM User Guide.

IAM roles with temporary credentials are useful in the following situations:

- Temporary IAM user permissions – An IAM user can assume an IAM role to temporarily take on different permissions for a specific task.
- Federated user access – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as federated users. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated Users and Roles in the IAM User Guide.
- Cross-account access – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see How IAM Roles Differ from Resource-based Policies in the IAM User Guide.
- AWS service access – A service role is an IAM role that a service assumes to perform actions in your account on your behalf. When you set up some AWS service environments, you must define a role for the service to assume. This service role must include all the permissions that are required for the service to access the AWS resources that it needs. Service roles vary from service to service, but many allow you to choose your permissions as long as you meet the documented requirements for that service. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. You can create, modify, and delete a service role from within IAM. For example, you can create a role that allows Amazon Redshift to access an Amazon S3 bucket on your behalf and then load data from that bucket into an Amazon Redshift cluster. For more information, see Creating a Role to Delegate Permissions to an AWS Service in the IAM User Guide.
- Applications running on Amazon EC2 – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM Role to Grant Permissions to Applications Running on Amazon EC2 Instances in the IAM User Guide.

To learn whether to use IAM roles, see When to Create an IAM Role (Instead of a User) in the IAM User Guide.

使用政策管理存取權

You control access in AWS by creating policies and attaching them to IAM identities or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when an entity (root user, IAM user, or IAM role) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON Policies in the IAM User Guide.

An IAM administrator can use policies to specify who has access to AWS resources, and what actions they can perform on those resources. Every IAM entity (user or role) starts with no permissions. In other words, by default, users can do nothing, not even change their own password. To give a user permission to do something, an administrator must attach a permissions policy to a user. Or the administrator can add the user to a group that has the intended permissions. When an administrator gives permissions to a group, all users in that group are granted those permissions.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, suppose that you have a policy that allows the `iam:GetRole` action. A user with that policy can get role information from the AWS Management Console, the AWS CLI, or the AWS API.

身分型政策

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, role, or group. These policies control what actions that identity can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Creating IAM Policies in the IAM User Guide.

Identity-based policies can be further categorized as inline policies or managed policies. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choosing Between Managed Policies and Inline Policies in the IAM User Guide.

資源型政策

Resource-based policies are JSON policy documents that you attach to a resource such as an Amazon S3 bucket. Service administrators can use these policies to define what actions a specified principal (account member, user, or role) can perform on that resource and under what conditions. Resource-based policies are inline policies. There are no managed resource-based policies.

存取控制清單 (ACL)

Access control policies (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they are the only policy type that does not use the JSON policy document format. Amazon S3, AWS WAF, and Amazon VPC are examples of services that support ACLs. To learn more about ACLs, see Access Control List (ACL) Overview in the Amazon Simple Storage Service Developer Guide.

其他政策類型

AWS supports additional, less-common policy types. These policy types can set the maximum permissions granted to you by the more common policy types.

- Permissions boundaries – A permissions boundary is an advanced feature in which you set the maximum permissions that an identity-based policy can grant to an IAM entity (IAM user or role). You can set a permissions boundary for an entity. The resulting permissions are the intersection of entity's identity-based policies and its permissions boundaries. Resource-based policies that specify the user or role in the `Principal` field are not limited by the permissions boundary. An explicit deny in any of these policies overrides the allow. For more information about permissions boundaries, see Permissions Boundaries for IAM Entities in the IAM User Guide.
- Service control policies (SCPs) – SCPs are JSON policies that specify the maximum permissions for an organization or organizational unit (OU) in AWS Organizations. AWS Organizations is a service for grouping and centrally managing multiple AWS accounts that your business owns. If you enable all features in an organization, then you can apply service control policies (SCPs) to any or all of your accounts. The SCP limits permissions for entities in member accounts, including each AWS account root user. For more information about Organizations and SCPs, see How SCPs Work in the AWS Organizations User Guide.
- Session policies – Session policies are advanced policies that you pass as a parameter when you programmatically create a temporary session for a role or federated user. The resulting session's permissions are the intersection of the user or role's identity-based policies and the session policies. Permissions can also come from a resource-based policy. An explicit deny in any of these policies overrides the allow. For more information, see Session Policies in the IAM User Guide.

多種政策類型

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see Policy Evaluation Logic in the IAM User Guide.

AWS IoT 如何搭配 IAM 運作

在您使用 IAM 管理 AWS IoT 的存取權前，建議您先了解可與 AWS IoT 搭配使用的 IAM 功能有哪些。若要取得 AWS IoT 和其他 AWS 服務與 IAM 搭配運作的高層級檢視，請參閱《IAM User Guide》中的與 IAM 搭配運作的 AWS 服務。

主題

- IAM 政策 (p. 183)
- AWS IoT 身分型政策 (p. 184)
- AWS IoT 資源型政策 (p. 195)
- 根據 AWS IoT 標籤進行授權 (p. 195)
- AWS IoT IAM 角色 (p. 196)

IAM 政策

AWS IoT 適用於 AWS IoT 和 IAM 政策。本主題僅討論 IAM 政策。如需詳細資訊，請參閱AWS IoT 政策 (p. 147)。AWS Identity and Access Management 定義每個 AWS IoT 定義之操作的政策動作，包括控制平面和資料平面 API。

IAM 受管政策

AWS IoT 提供一組 IAM 受管政策，您可以依原狀使用或做為建立自訂 IAM 政策的起點。這些政策允許存取組態和資料操作。組態操作讓您能夠建立物件、憑證、政策和規則。資料操作則可透過 MQTT 或 HTTP 通訊協定傳送資料。這些範本於下列資料表中說明。

政策範本	描述
AWSIoTConfigAccess	允許相關身分存取所有 AWS IoT 組態操作。此政策會影響資料處理和儲存。
AWSIoTConfigReadOnlyAccess	允許相關身分存取唯讀組態操作。
AWSIoTDataAccess	允許相關身分完整存取所有 AWS IoT 資料操作。資料操作則可透過 MQTT 或 HTTP 通訊協定傳送資料。
AWSIoTEventsFullAccess	允許相關身分完整存取 AWS IoT 事件。
AWSIoTEventsReadOnlyAccess	允許相關的身分唯讀存取 AWS IoT 事件。
AWSIoTFullAccess	允許相關身分完整存取所有 AWS IoT 組態和簡訊操作。
AWSIotLogging	允許相關的身分建立 Amazon CloudWatch Logs 群組和為群組建立 strean 日誌。此政策會連接至您的 CloudWatch 記錄角色。
AWSIoTOTAUpdate	允許相關的身分建立 AWS IoT 任務、AWS IoT 程式碼簽署任務，以及描述 AWS 程式碼簽署者任務。

政策範本	描述
AWSIoTRuleActions	允許相關身分存取所有受 AWS IoT 規則動作支援的 AWS 服務。
AWSIoTThingsRegistration	允許相關的身分使用 StartThingRegistrationTask API 大量登錄物件。此政策會影響資料處理和儲存。

AWS IoT 身分型政策

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下會允許或拒絕動作。AWS IoT 支援特定動作、資源及條件金鑰。若要了解您在 JSON 政策中使用的所有元素，請參閱 IAM User Guide中的 IAM JSON 政策元素參考。

動作

The `Action` element of an IAM identity-based policy describes the specific action or actions that will be allowed or denied by the policy. Policy actions usually have the same name as the associated AWS API operation. The action is used in a policy to grant permissions to perform the associated operation.

下表列出 IAM IoT 動作、相關的 AWS IoT API，以及動作所操控的資源。

政策動作	AWS IoT API	資源
iot:AcceptCertificateTransfer	AcceptCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id* Note ARN 指定的 AWS 帳戶必須是憑證將傳送至的帳戶。
iot:AddThingToThingGroup	AddThingToThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:AssociateTargetsWithJob	AssociateTargetsWithJob	無
iot:AttachPolicy	AttachPolicy	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 或 arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:AttachPrincipalPolicy	AttachPrincipalPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:AttachThingPrincipal	AttachThingPrincipal	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:CancelCertificateTransfer	CancelCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id* Note ARN 指定的 AWS 帳戶必須是憑證將傳送至的帳戶。
iot:CancelJob	CancelJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:CancelJobExecution	CancelJobExecution	arn:aws:iot:*region*:*account-id*:job/*job-id*

政策動作	AWS IoT API	資源
		arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ClearDefaultAuthorizer	ClearDefaultAuthorizer	無
iot:CreateAuthorizer	CreateAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name*
iot:CreateCertificateFromCsr	CreateCertificateFromCsr	*
iot:CreateJob	CreateJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:CreateKeysAndCertificate	CreateKeysAndCertificate	*
iot:CreatePolicy	CreatePolicy	*
CreatePolicyVersion	iot:CreatePolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name* Note 這必須是 AWS IoT 政策，而非 IAM 政策。
iot:CreateRoleAlias	CreateRoleAlias	(參數：roleAlias) arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:CreateThing	CreateThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:CreateThingGroup	CreateThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 適用正在建立的群組和父群組 (若使用)
iot:CreateThingType	CreateThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:CreateTopicRule	CreateTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:DeleteAuthorizer	DeleteAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-name*
iot:DeleteCACertificate	DeleteCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:DeleteCertificate	DeleteCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DeleteJob	DeleteJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:DeleteJobExecution	DeleteJobExecution	arn:aws:iot:*region*:*account-id*:job/*job-id* arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:DeletePolicy	DeletePolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:DeletePolicyVersion	DeletePolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:DeleteRegistrationCode	DeleteRegistrationCode	*
iot:DeleteRoleAlias	DeleteRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:DeleteThing	DeleteThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*

政策動作	AWS IoT API	資源
iot:DeleteThingGroup	DeleteThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DeleteThingType	DeleteThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DeleteTopicRule	DeleteTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:DeleteV2LoggingLevel	DeleteV2LoggingLevel	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DeprecateThingType	DeprecateThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DescribeAuthorizer	DescribeAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name* (參數：authorizerName) 無
iot:DescribeCACertificate	DescribeCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:DescribeCertificate	DescribeCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DescribeDefaultAuthorizer	DescribeDefaultAuthorizer	無
iot:DescribeEndpoint	DescribeEndpoint	*
iot:DescribeEventConfigurations	DescribeEventConfigurations	無
iot:DescribeIndex	DescribeIndex	arn:aws:iot:*region*:*account-id*:index/*index-name*
iot:DescribeJob	DescribeJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:DescribeJobExecution	DescribeJobExecution	無
iot:DescribeRoleAlias	DescribeRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:DescribeThing	DescribeThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:DescribeThingGroup	DescribeThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DescribeThingRegistrationTask	DescribeThingRegistrationTask	無
iot:DescribeThingType	DescribeThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DetachPolicy	DetachPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id* 或 arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DetachPrincipalPolicy	DetachPrincipalPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DetachThingPrincipal	DetachThingPrincipal	arn:aws:iot:*region*:*account-id*:cert/*cert-id*

政策動作	AWS IoT API	資源
iot:DisableTopicRule	DisableTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:EnableTopicRule	EnableTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:GetEffectivePolicies	GetEffectivePolicies	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:GetIndexingConfiguration	GetIndexingConfiguration	無
iot:GetJobDocument	GetJobDocument	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:GetLoggingOptions	GetLoggingOptions	*
iot:GetPolicy	GetPolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:GetPolicyVersion	GetPolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:GetRegistrationCode	GetRegistrationCode	*
iot:GetTopicRule	GetTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:ListAttachedPolicies	ListAttachedPolicies	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 或 arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListAuthorizers	ListAuthorizers	無
iot:ListCACertificates	ListCACertificates	*
iot:ListCertificates	ListCertificates	*
iot:ListCertificatesByCA	ListCertificatesByCA	*
iot:ListIndices	ListIndices	無
iot:ListJobExecutionsForJob	ListJobExecutionsForJob	無
iot:ListJobExecutionsForThing	ListJobExecutionsForThing	無
iot:ListJobs	ListJobs	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 若使用 thingGroupName 參數
iot:ListOutgoingCertificates	ListOutgoingCertificates	*
iot:ListPolicies	ListPolicies	*
iot:ListPolicyPrincipals	ListPolicyPrincipals	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:ListPolicyVersions	ListPolicyVersions	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:ListPrincipalPolicies	ListPrincipalPolicies	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListPrincipalThings	ListPrincipalThings	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListRoleAliases	ListRoleAliases	無
iot:ListTargetsForPolicy	ListTargetsForPolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*

政策動作	AWS IoT API	資源
iot:ListThingGroups	ListThingGroups	無
iot:ListThingGroupsForThing	ListThingGroupsForThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ListThingPrincipals	ListThingPrincipals	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ListThingRegistrationTaskReports	ListThingRegistrationTaskReports	無
iot:ListThingRegistrationTasks	ListThingRegistrationTasks	無
iot:ListThingTypes	ListThingTypes	*
iot:ListThings	ListThings	*
iot:ListThingsInThingGroup	ListThingsInThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:ListTopicRules	ListTopicRules	*
iot:ListV2LoggingLevels	ListV2LoggingLevels	無
iot:RegisterCACertificate	RegisterCACertificate	*
iot:RegisterCertificate	RegisterCertificate	*
iot:RegisterThing	RegisterThing	無
iot:RejectCertificateTransfer	RejectCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:RemoveThingFromThingGroup	RemoveThingFromThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ReplaceTopicRule	ReplaceTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:SearchIndex	SearchIndex	arn:aws:iot:*region*:*account-id*:index/*index-id*
iot:SetDefaultAuthorizer	SetDefaultAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name*
iot:SetDefaultPolicyVersion	SetDefaultPolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:SetLoggingOptions	SetLoggingOptions	arn:aws:iot:*region*:*account-id*:role/*role-name*
iot:SetV2LoggingLevel	SetV2LoggingLevel	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:SetV2LoggingOptions	SetV2LoggingOptions	arn:aws:iot:*region*:*account-id*:role/*role-name*
iot:StartThingRegistrationTask	StartThingRegistrationTask	無
iot:StopThingRegistrationTask	StopThingRegistrationTask	無
iot:TestAuthorization	TestAuthorization	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:TestInvokeAuthorizer	TestInvokeAuthorizer	無
iot:TransferCertificate	TransferCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*

政策動作	AWS IoT API	資源
iot:UpdateAuthorizer	UpdateAuthorizer	arn:aws:iot:*region*:*account-id*:authorizerfunction/*authorizer-function-name*
iot:UpdateCACertificate	UpdateCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:UpdateCertificate	UpdateCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:UpdateEventConfigurations	UpdateEventConfigurations	無
iot:UpdateIndexingConfiguration	UpdateIndexingConfiguration	無
iot:UpdateRoleAlias	UpdateRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:UpdateThing	UpdateThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:UpdateThingGroup	UpdateThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:UpdateThingGroupsForThing	UpdateThingGroupsForThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*

AWS IoT 中的政策動作會在動作前使用以下前綴：`iot:`。例如，若要授與某人使用 `ListThings` API 列出 AWS 帳戶中註冊之所有 IoT 物件的許可，請將 `iot:ListThings` 動作包含在其政策中。政策陳述式必須包含 `Action` 或 `NotAction` 元素。AWS IoT 會定義一組自己的動作，來描述您可以使用此服務執行的任務。

若要在單一陳述式中指定多個動作，請用逗號分隔，如下所示：

```
"Action": [
      "ec2:action1",
      "ec2:action2"
```

您也可以使用萬用字元 (*) 來指定多個動作。例如，若要指定開頭是 `Describe` 文字的所有動作，請包含以下動作：

```
"Action": "iot:Describe*"
```

若要查看 AWS IoT 動作的清單，請參閱 IAM User Guide中的 Actions Defined by AWS IoT。

資源

The `Resource` element specifies the object or objects to which the action applies. Statements must include either a `Resource` or a `NotResource` element. You specify a resource using an ARN or using the wildcard (*) to indicate that the statement applies to all resources.

AWS IoT 資源

政策動作	AWS IoT API	資源
iot:AcceptCertificateTransfer	AcceptCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id* Note ARN 指定的 AWS 帳戶必須是憑證將傳送至的帳戶。
iot:AddThingToThingGroup	AddThingToThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*

政策動作	AWS IoT API	資源
		arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:AssociateTargetsWithJob	AssociateTargetsWithJob	無
iot:AttachPolicy	AttachPolicy	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 或 arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:AttachPrincipalPolicy	AttachPrincipalPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:AttachThingPrincipal	AttachThingPrincipal	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:CancelCertificateTransfer	CancelCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id* Note ARN 指定的 AWS 帳戶必須是憑證將傳送至的帳戶。
iot:CancelJob	CancelJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:CancelJobExecution	CancelJobExecution	arn:aws:iot:*region*:*account-id*:job/*job-id* arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ClearDefaultAuthorizer	ClearDefaultAuthorizer	無
iot:CreateAuthorizer	CreateAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name*
iot:CreateCertificateFromCsr	CreateCertificateFromCsr	*
iot:CreateJob	CreateJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:CreateKeysAndCertificate	CreateKeysAndCertificate	*
iot:CreatePolicy	CreatePolicy	*
CreatePolicyVersion	iot:CreatePolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name* Note 這必須是 AWS IoT 政策，而非 IAM 政策。
iot:CreateRoleAlias	CreateRoleAlias	(參數：roleAlias) arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:CreateThing	CreateThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:CreateThingGroup	CreateThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 適用正在建立的群組和父群組 (若使用)
iot:CreateThingType	CreateThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*

政策動作	AWS IoT API	資源
iot:CreateTopicRule	CreateTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:DeleteAuthorizer	DeleteAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-name*
iot:DeleteCACertificate	DeleteCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:DeleteCertificate	DeleteCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DeleteJob	DeleteJob	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:DeleteJobExecution	DeleteJobExecution	arn:aws:iot:*region*:*account-id*:job/*job-id* arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:DeletePolicy	DeletePolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:DeletePolicyVersion	DeletePolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:DeleteRegistrationCode	DeleteRegistrationCode	*
iot:DeleteRoleAlias	DeleteRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:DeleteThing	DeleteThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:DeleteThingGroup	DeleteThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DeleteThingType	DeleteThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DeleteTopicRule	DeleteTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:DeleteV2LoggingLevel	DeleteV2LoggingLevel	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DeprecateThingType	DeprecateThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DescribeAuthorizer	DescribeAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name* (參數：authorizerName) 無
iot:DescribeCACertificate	DescribeCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:DescribeCertificate	DescribeCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DescribeDefaultAuthorizer	DescribeDefaultAuthorizer	無
iot:DescribeEndpoint	DescribeEndpoint	*
iot:DescribeEventConfigurations	DescribeEventConfigurations	無
iot:DescribeIndex	DescribeIndex	arn:aws:iot:*region*:*account-id*:index/*index-name*
iot:DescribeJob	DescribeJob	arn:aws:iot:*region*:*account-id*:job/*job-id*

政策動作	AWS IoT API	資源
iot:DescribeJobExecution	DescribeJobExecution	無
iot:DescribeRoleAlias	DescribeRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:DescribeThing	DescribeThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:DescribeThingGroup	DescribeThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DescribeThingRegistrationTask	DescribeThingRegistrationTask	無
iot:DescribeThingType	DescribeThingType	arn:aws:iot:*region*:*account-id*:thingtype/*thing-type-name*
iot:DetachPolicy	DetachPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id* 或 arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:DetachPrincipalPolicy	DetachPrincipalPolicy	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DetachThingPrincipal	DetachThingPrincipal	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:DisableTopicRule	DisableTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:EnableTopicRule	EnableTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:GetEffectivePolicies	GetEffectivePolicies	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:GetIndexingConfiguration	GetIndexingConfiguration	無
iot:GetJobDocument	GetJobDocument	arn:aws:iot:*region*:*account-id*:job/*job-id*
iot:GetLoggingOptions	GetLoggingOptions	*
iot:GetPolicy	GetPolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:GetPolicyVersion	GetPolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:GetRegistrationCode	GetRegistrationCode	*
iot:GetTopicRule	GetTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:ListAttachedPolicies	ListAttachedPolicies	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 或 arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListAuthorizers	ListAuthorizers	無
iot:ListCACertificates	ListCACertificates	*
iot:ListCertificates	ListCertificates	*
iot:ListCertificatesByCA	ListCertificatesByCA	*

政策動作	AWS IoT API	資源
iot:ListIndices	ListIndices	無
iot:ListJobExecutionsForJob	ListJobExecutionsForJob	無
iot:ListJobExecutionsForThing	ListJobExecutionsForThing	無
iot:ListJobs	ListJobs	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* 若使用 thingGroupName 參數
iot:ListOutgoingCertificates	ListOutgoingCertificates	*
iot:ListPolicies	ListPolicies	*
iot:ListPolicyPrincipals	ListPolicyPrincipals	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:ListPolicyVersions	ListPolicyVersions	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:ListPrincipalPolicies	ListPrincipalPolicies	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListPrincipalThings	ListPrincipalThings	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:ListRoleAliases	ListRoleAliases	無
iot:ListTargetsForPolicy	ListTargetsForPolicy	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:ListThingGroups	ListThingGroups	無
iot:ListThingGroupsForThing	ListThingGroupsForThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ListThingPrincipals	ListThingPrincipals	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:ListThingRegistrationTaskReports	ListThingRegistrationTaskReports	無
iot:ListThingRegistrationTasks	ListThingRegistrationTasks	無
iot:ListThingTypes	ListThingTypes	*
iot:ListThings	ListThings	*
iot:ListThingsInThingGroup	ListThingsInThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:ListTopicRules	ListTopicRules	*
iot:ListV2LoggingLevels	ListV2LoggingLevels	無
iot:RegisterCACertificate	RegisterCACertificate	*
iot:RegisterCertificate	RegisterCertificate	*
iot:RegisterThing	RegisterThing	無
iot:RejectCertificateTransfer	RejectCertificateTransfer	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:RemoveThingFromThingGroup	RemoveThingFromThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name* arn:aws:iot:*region*:*account-id*:thing/*thing-name*

政策動作	AWS IoT API	資源
iot:ReplaceTopicRule	ReplaceTopicRule	arn:aws:iot:*region*:*account-id*:rule/*rule-name*
iot:SearchIndex	SearchIndex	arn:aws:iot:*region*:*account-id*:index/*index-id*
iot:SetDefaultAuthorizer	SetDefaultAuthorizer	arn:aws:iot:*region*:*account-id*:authorizer/*authorizer-function-name*
iot:SetDefaultPolicyVersion	SetDefaultPolicyVersion	arn:aws:iot:*region*:*account-id*:policy/*policy-name*
iot:SetLoggingOptions	SetLoggingOptions	arn:aws:iot:*region*:*account-id*:role/*role-name*
iot:SetV2LoggingLevel	SetV2LoggingLevel	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:SetV2LoggingOptions	SetV2LoggingOptions	arn:aws:iot:*region*:*account-id*:role/*role-name*
iot:StartThingRegistrationTask	StartThingRegistrationTask	無
iot:StopThingRegistrationTask	StopThingRegistrationTask	無
iot:TestAuthorization	TestAuthorization	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:TestInvokeAuthorizer	TestInvokeAuthorizer	無
iot:TransferCertificate	TransferCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:UpdateAuthorizer	UpdateAuthorizer	arn:aws:iot:*region*:*account-id*:authorizerfunction/*authorizer-function-name*
iot:UpdateCACertificate	UpdateCACertificate	arn:aws:iot:*region*:*account-id*:cacert/*cert-id*
iot:UpdateCertificate	UpdateCertificate	arn:aws:iot:*region*:*account-id*:cert/*cert-id*
iot:UpdateEventConfigurations	UpdateEventConfigurations	無
iot:UpdateIndexingConfiguration	UpdateIndexingConfiguration	無
iot:UpdateRoleAlias	UpdateRoleAlias	arn:aws:iot:*region*:*account-id*:rolealias/*role-alias-name*
iot:UpdateThing	UpdateThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*
iot:UpdateThingGroup	UpdateThingGroup	arn:aws:iot:*region*:*account-id*:thinggroup/*thing-group-name*
iot:UpdateThingGroupsForThing	UpdateThingGroupsForThing	arn:aws:iot:*region*:*account-id*:thing/*thing-name*

如需 ARN 格式的詳細資訊，請參閱 Amazon Resource Name (ARN) 和 AWS 服務命名空間。

有些 AWS IoT 動作 (例如用來建立資源的動作) 無法在特定資源上執行。在這些情況下，您必須使用萬用字元 (*)。

```
"Resource": "*"
```

若要查看 AWS IoT 資源類型的清單及其 ARN，請參閱《IAM User Guide》中的 Resources Defined by AWS IoT。您若要了解您可以使用哪些動作指定每項資源的 ARN，請參閱 Actions Defined by AWS IoT。

條件索引鍵

The `Condition` element (or `Condition` block) lets you specify conditions in which a statement is in effect. The `Condition` element is optional. You can build conditional expressions that use condition operators, such as equals or less than, to match the condition in the policy with values in the request.

If you specify multiple `Condition` elements in a statement, or multiple keys in a single `Condition` element, AWS evaluates them using a logical `AND` operation. If you specify multiple values for a single condition key, AWS evaluates the condition using a logical `OR` operation. All of the conditions must be met before the statement's permissions are granted.

You can also use placeholder variables when you specify conditions. For example, you can grant an IAM user permission to access a resource only if it is tagged with their IAM user name. For more information, see IAM Policy Elements: Variables and Tags in the IAM User Guide.

AWS IoT 會定義自己的一組條件金鑰，也支援使用一些全球條件金鑰。若要查看所有 AWS 全域條件金鑰，請參閱 IAM User Guide中的 AWS 全域條件內容金鑰。

AWS IoT 條件金鑰

AWS IoT 條件金鑰	描述	類型
`aws:RequestTag/${<tag-key>}`	在使用者對 AWS IoT 提出的請求中存在的標籤金鑰。	字串
`aws:ResourceTag/${<tag-key>}`	連接到 AWS IoT 資源的標籤的標籤金鑰元件。	字串
`aws:TagKeys`	與請求中資源相關聯的所有標籤鍵名稱清單。	字串

若要查看 AWS IoT 條件金鑰的清單，請參閱《IAM User Guide》中的 Condition Keys for AWS IoT。若要了解您可以針對何種動作及資源使用條件金鑰，請參閱 Actions Defined by AWS IoT。

範例

若要檢視 AWS IoT 身分型政策範例，請參閱AWS IoT 身分類型政策範例 (p. 196)。

AWS IoT 資源型政策

資源型政策是 JSON 政策文件，這些文件會指定指定的委託人可對 AWS IoT 資源以及在怎樣的條件下執行哪些動作。

AWS IoT 不支援 IAM 資源型政策。不過，它支援 AWS IoT 資源型政策。

根據 AWS IoT 標籤進行授權

您可以將標籤連接到 AWS IoT 資源，或是在請求中將標籤傳遞給 AWS IoT。若要根據標籤控制存取權，請使用 `iot:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件金鑰，在政策的條件元素中提供標籤資訊。如需詳細資訊，請參閱搭配 IAM 政策使用標籤 (p. 126)。如需標記 AWS IoT 資源的詳細資訊，請參閱為您的 AWS IoT 資源建立標籤 (p. 125)。

若要檢視身分型政策範例，了解如何根據資源上的標籤來限制該資源的存取權，請參閱 根據標籤檢視 AWS IoT 資源 (p. 198)。

AWS IoT IAM 角色

IAM 角色是您 AWS 帳戶中的一種實體，具備特定許可。

搭配 AWS IoT 使用暫時登入資料

您可以搭配聯合使用暫時登入資料登入、取得 IAM 角色，或是取得跨帳戶角色。您取得暫時安全登入資料的方式是透過呼叫 AWS STS API 操作 (例如，AssumeRole 或 GetFederationToken)。

AWS IoT 支援使用暫時登入資料。

服務連結角色

服務連結角色可讓 AWS 服務存取其他服務中的資源，以代您完成動作。服務連結角色會顯示在您的 IAM 帳戶中，並由該服務所擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

AWS IoT 不支援服務連結角色。

服務角色

此功能可讓服務代表您擔任服務角色。此角色可讓服務存取其他服務中的資源，以代表您完成動作。服務角色會出現在您的 IAM 帳戶，且由該帳戶所擁有。這表示 IAM 管理員可以變更此角色的許可。不過，這樣可能會破壞此服務的功能。

AWS IoT 不支援服務角色。

AWS IoT 身分類型政策範例

根據預設，IAM 使用者和角色不具備建立或修改 AWS IoT 資源的許可。他們也無法使用 AWS Management Console、AWS CLI 或 AWS API 執行任務。IAM 管理員必須建立 IAM 政策，授予使用者和角色在他們所需指定資源上執行特定 API 操作的許可。管理員接著必須將這些政策連接至需要這些許可的 IAM 使用者或群組。

若要了解如何使用這些範例 JSON 政策文件建立 IAM 身分類型政策，請參閱 IAM User Guide中的在 JSON 標籤上建立政策。

主題

- 政策最佳實務 (p. 196)
- 使用 AWS IoT 主控台 (p. 197)
- 允許使用者檢視他們自己的許可 (p. 197)
- 根據標籤檢視 AWS IoT 資源 (p. 198)

政策最佳實務

Identity-based policies are very powerful. They determine whether someone can create, access, or delete AWS IoT resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- Get Started Using AWS Managed Policies – To start using AWS IoT quickly, use AWS managed policies to give your employees the permissions they need. These policies are already available in your account and are maintained and updated by AWS. For more information, see Get Started Using Permissions With AWS Managed Policies in the IAM User Guide.
- Grant Least Privilege – When you create custom policies, grant only the permissions required to perform a task. Start with a minimum set of permissions and grant additional permissions as necessary. Doing so

is more secure than starting with permissions that are too lenient and then trying to tighten them later. For more information, see Grant Least Privilege in the IAM User Guide.

- Enable MFA for Sensitive Operations – For extra security, require IAM users to use multi-factor authentication (MFA) to access sensitive resources or API operations. For more information, see Using Multi-Factor Authentication (MFA) in AWS in the IAM User Guide.
- Use Policy Conditions for Extra Security – To the extent that it's practical, define the conditions under which your identity-based policies allow access to a resource. For example, you can write conditions to specify a range of allowable IP addresses that a request must come from. You can also write conditions to allow requests only within a specified date or time range, or to require the use of SSL or MFA. For more information, see IAM JSON Policy Elements: Condition in the IAM User Guide.

使用 AWS IoT 主控台

若要存取 AWS IoT 主控台，您必須擁有最低的一組許可。這些許可必須允許您列出和檢視您 AWS 帳戶中 AWS IoT 資源的詳細資訊。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (IAM 使用者或角色) 而言，主控台就無法如預期運作。

為確保那些實體仍可使用 AWS IoT 主控台，請也將以下 AWS 受管政策連接到實體：`AWSIoTFullAccess`如需詳細資訊，請參閱《IAM User Guide》中的新增許可到使用者。

您不需要針對只呼叫 AWS CLI 或 AWS API 的使用者允許最基本的主控台許可。反之，只需允許存取符合您嘗試執行之 API 操作的動作就可以了。

允許使用者檢視他們自己的許可

This example shows how you might create a policy that allows IAM users to view the inline and managed policies that are attached to their user identity. This policy includes permissions to complete this action on the console or programmatically using the AWS CLI or AWS API.

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "ViewOwnUserInfo",
                "Effect": "Allow",
                "Action": [
                    "iam:GetUserPolicy",
                    "iam:ListGroupsForUser",
                    "iam:ListAttachedUserPolicies",
                    "iam:ListUserPolicies",
                    "iam:GetUser"
                ],
                "Resource": [
                    "arn:aws:iam::*:user/${aws:username}"
                ]
            },
            {
                "Sid": "NavigateInConsole",
                "Effect": "Allow",
                "Action": [
                    "iam:GetGroupPolicy",
                    "iam:GetPolicyVersion",
                    "iam:GetPolicy",
                    "iam:ListAttachedGroupPolicies",
                    "iam:ListGroupPolicies",
                    "iam:ListPolicyVersions",
                    "iam:ListPolicies",
                    "iam:ListUsers"
                ],
```

```
                "Resource": "*"
            }
        ]
    }
```

根據標籤檢視 AWS IoT 資源

您可以在身分類型政策中使用條件，來根據標籤控制存取 AWS IoT 資源。此範例會示範如何建立政策，允許檢視物件。但是，只有在物件標籤 `Owner` 的值是該使用者的使用者名稱時，才會授予該許可。此政策也會授予在主控台完成此動作的必要許可。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ListThingsInConsole",
            "Effect": "Allow",
            "Action": "iot:ListThings",
            "Resource": "*"
        },
        {
            "Sid": "ViewThingsIfOwner",
            "Effect": "Allow",
            "Action": "iot:DescribeThing",
            "Resource": "arn:aws:iot:*:*:thing/*",
            "Condition": {
                "StringEquals": {"iot:ResourceTag/Owner": "${aws:username}"}
            }
        }
    ]
}
```

您可以將此政策連接到您帳戶中的 IAM 使用者。若名為 `richard-roe` 的使用者嘗試檢視 AWS IoT 物件，該物件必須標記為 `Owner=richard-roe` 或 `owner=richard-roe`。否則他便會被拒絕存取。條件標籤鍵 `Owner` 符合 `Owner` 和 `owner`，因為條件金鑰名稱不區分大小寫。如需詳細資訊，請參閱《IAM User Guide》中的 IAM JSON 政策元素：條件。

針對 AWS IoT Identity and Access 進行故障診斷

請使用以下資訊來協助您診斷和修復使用 AWS IoT 和 IAM 時發生的常見問題。

主題

- 我未獲得在 AWS IoT 中執行動作的授權 (p. 198)
- 我未獲得執行 iam: PassRole 的授權 (p. 199)
- 我想要檢視我的存取金鑰 (p. 199)
- 我是管理員，我希望允許其他人存取 AWS IoT (p. 199)
- 我希望允許我 AWS 帳戶之外的人存取我的 AWS IoT 資源 (p. 200)

我未獲得在 AWS IoT 中執行動作的授權

若 AWS Management Console 告知您並未獲得執行動作的授權，您必須聯絡您的管理員以取得協助。您的管理員是提供您使用者名稱和密碼的人員。

以下範例錯誤會在 `mateojackson` IAM 使用者嘗試使用主控台檢視物件的詳細資訊，但卻沒有 `iot:`*`DescribeThing`* 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 iot:DescribeThing
            on resource: MyIoTThing
```

在此情況下，Mateo 會請求管理員更新他的政策，允許他使用 `iot:DescribeThing` 動作存取 *`MyIoTThing`* 資源。

我未獲得執行 iam: PassRole 的授權

If you receive an error that you're not authorized to perform the `iam:PassRole` action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password. Ask that person to update your policies to allow you to pass a role to AWS IoT.

Some AWS services allow you to pass an existing role to that service, instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in AWS IoT. However, the action requires the service to have permissions granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform: iam:PassRole
```

In this case, Mary asks her administrator to update her policies to allow her to perform the `iam:PassRole` action.

我想要檢視我的存取金鑰

After you create your IAM user access keys, you can view your access key ID at any time. However, you can't view your secret access key again. If you lose your secret key, you must create a new access key pair.

Access keys consist of two parts: an access key ID (for example, `AKIAIOSFODNN7EXAMPLE`) and a secret access key (for example, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Like a user name and password, you must use both the access key ID and secret access key together to authenticate your requests. Manage your access keys as securely as you do your user name and password.

> **Important**
>
> Do not provide your access keys to a third party, even to help find your canonical user ID. By doing this, you might give someone permanent access to your account.

When you create an access key pair, you are prompted to save the access key ID and secret access key in a secure location. The secret access key is available only at the time you create it. If you lose your secret access key, you must add new access keys to your IAM user. You can have a maximum of two access keys. If you already have two, you must delete one key pair before creating a new one. To view instructions, see Managing Access Keys in the IAM User Guide.

我是管理員，我希望允許其他人存取 AWS IoT

To allow others to access AWS IoT, you must create an IAM entity (user or role) for the person or application that needs access. They will use the credentials for that entity to access AWS. You must then attach a policy to the entity that grants them the correct permissions in AWS IoT.

To get started right away, see Creating Your First IAM Delegated User and Group in the IAM User Guide.

我希望允許我 AWS 帳戶之外的人存取我的 AWS IoT 資源

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether AWS IoT supports these features, see AWS IoT 如何搭配 IAM 運作 (p. 183).
- To learn how to provide access to your resources across AWS accounts that you own, see Providing Access to an IAM User in Another AWS Account That You Own in the IAM User Guide.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing Access to AWS Accounts Owned by Third Parties in the IAM User Guide.
- To learn how to provide access through identity federation, see Providing Access to Externally Authenticated Users (Identity Federation) in the IAM User Guide.
- To learn the difference between using roles and resource-based policies for cross-account access, see How IAM Roles Differ from Resource-based Policies in the IAM User Guide.

使用 AWS IoT Core 進行記錄和監控

監控是維護 AWS IoT 及您 AWS 解決方案之可靠性、可用性和效能的重要部分。您應該收集 AWS 解決方案各方面的監控資料，以便在發生多點故障的情況下，更輕鬆地偵錯。在開始監控 AWS IoT 之前，應先建立監控計畫，為下列問題提供解答：

- 監控目標是什麼？
- 監控哪些資源？
- 監控這些資源的頻率為何？
- 將使用哪些監控工具？
- 誰將執行監控任務？
- 發生問題時應該通知誰？

下一步是在各個時間點和不同的負載條件下測量效能，以在您的環境中確立 AWS IoT 正常效能的基準。當您監控 AWS IoT 時，請存放歷史記錄監控資料，如此才能與目前的效能資料做比較、辨識正常效能模式和效能異常狀況、規劃問題處理方式。

舉例而言，若您使用的是 Amazon EC2，您可以監控您的執行個體的 CPU 使用率、磁碟 I/O、網路運作。若效能不符合您所建立的基準，您可能需要重新設定或將執行個體最佳化，以降低 CPU 使用率、改善磁碟 I/O、降低網路流量。

若要建立基準，您至少必須監控下列指標：

- `PublishIn.Success`
- `PublishOut.Success`
- `Subscribe.Success`
- `Ping.Success`
- `Connect.Success`
- `GetThingShadow.Accepted`
- `UpdateThingShadow.Accepted`
- `DeleteThingShadow.Accepted`

- `RulesExecuted`

主題

- 監控工具 (p. 201)
- 使用 Amazon CloudWatch 進行監控 (p. 202)
- 使用 CloudWatch Logs 進行監控 (p. 210)
- 使用 AWS CloudTrail 記錄 AWS IoT API 呼叫 (p. 230)

監控工具

AWS 提供可讓您用來監控 AWS IoT 的工具。您可設定部分這些工具以為您執行監控工作。部分工具將需要手動操作。建議您盡可能自動化監控任務。

自動化監控工具

您可以使用下列自動化監控工具來監看 AWS IoT，並在發生錯誤時進行回報：

- Amazon CloudWatch 警示 – 監看指定時段內的單一指標，並根據與多個時段內給定之閾值相對的指標值來執行一或多個動作。該動作為傳送至 Amazon Simple Notification Service (Amazon SNS) 主題或 Amazon EC2 Auto Scaling 政策的通知。CloudWatch 警示並不會僅因處於特定狀態而叫用動作。狀態必須已變更，且在指定的期間數內維持此狀態。如需詳細資訊，請參閱 使用 Amazon CloudWatch 進行監控 (p. 202)。
- Amazon CloudWatch Logs – 監控、存放及存取來自 AWS CloudTrail 或其他來源的日誌檔案。如需詳細資訊，請參閱 Amazon CloudWatch User Guide 中的監控日誌檔案。
- Amazon CloudWatch Events – 比對事件並將其路由至一或多個目標函數或串流，以進行變更、擷取狀態資訊，以及採取修正動作。如需更多資訊，請參閱 Amazon CloudWatch User Guide 中的何謂 Amazon CloudWatch Events。
- AWS CloudTrail 日誌監控 – 在帳戶之間共享日誌檔、將 CloudTrail 日誌檔傳送至 CloudWatch Logs, 以對其進行即時監控、以 Java 撰寫日誌處理應用程式，以及驗證日誌檔在由 CloudTrail 交付之後沒有發生變更。如需詳細資訊，請參閱 AWS CloudTrail User Guide中的使用 CloudTrail 日誌檔。

手動監控工具

監控 AWS IoT 的另一個重要部分包含手動監控 CloudWatch 警示未涵蓋的項目。AWS IoT、CloudWatch 和其他 AWS 服務主控台儀表板可提供您 AWS 環境狀態的快速概覽。建議您也查看 AWS IoT 上的日誌檔。

- AWS IoT 儀表板顯示：
  - CA 憑證
  - 憑證
  - 政策
  - 規則
  - 實物
- CloudWatch 首頁顯示：
  - 目前警示與狀態。
  - 警示與資源的圖形。
  - 服務運作狀態。

  您可以使用 CloudWatch 來執行以下操作：
  - 建立自定儀表板來監控您注重的服務
  - 用於疑難排解問題以及探索驅勢的圖形指標資料。

- 搜尋與瀏覽所有 AWS 資源指標。
- 建立與編輯要通知發生問題的警示。

使用 Amazon CloudWatch 進行監控

您可以使用 AWS IoT 來監控 CloudWatch，該服務會收集並處理來自 AWS IoT 的原始資料，進而將這些資料轉換為便於讀取且幾近即時的指標。這些統計資料會保存兩週的期間，以便您存取歷史資訊，並更清楚 web 應用程式或服務的執行方式。依預設，AWS IoT 指標資料會以一分鐘的間隔自動傳送至 CloudWatch。如需詳細資訊，請參閱 Amazon CloudWatch User Guide中的 Amazon CloudWatch、Amazon CloudWatch Events 和 Amazon CloudWatch Logs 為何？。

主題

- AWS IoT 指標與維度 (p. 202)
- 如何使用 AWS IoT 指標？ (p. 208)
- 建立 CloudWatch 警示來監控 AWS IoT (p. 208)

AWS IoT 指標與維度

當您與 AWS IoT 互動時，該服務會將以下指標與維度以每分鐘為週期傳送到 CloudWatch。您可以使用下列程序來檢視 AWS IoT 的指標。

查看指標 (CloudWatch 主控台)

指標會先依服務命名空間分組，再依各命名空間內不同的維度組合分類。

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. 在導覽窗格中，選擇指標。
3. 在 CloudWatch Metrics by Category (依類別劃分 CloudWatch 指標) 窗格中的 AWS IoT 指標類別下，選擇一個指標類別，然後在上方窗格中向下捲動以檢視完整的指標清單。

檢視指標 (CLI)

- 在命令提示中，使用下列命令：

```
aws cloudwatch list-metrics --namespace "AWS/IoT"
```

CloudWatch 顯示 AWS IoT 的下列指標：

AWS IoT 指標

`AWS/IoT` 命名空間包含下列指標。AWS IoT 每次收到請求時，都會傳送下列指標給 CloudWatch 一次。如需更多有關 CloudWatch 指標的資訊，請參閱 Amazon CloudWatch 指標。

AWS IoT 指標

指標	描述
`RulesExecuted`	AWS IoT 規則執行次數。
`NumLogEventsFailedToPublishThrottled`	批次中因調節錯誤而無法發佈的日誌事件數。

指標	描述
NumLogBatchesFailedToPublishThrottled	因調節錯誤而無法發佈的單批日誌事件。

規則指標

指標	描述
TopicMatch	對規則所接聽主題發佈的傳入訊息數目。RuleName 維度會包含規則的名稱。
ParseError	JSON 剖析錯誤數目，發生於對規則所接聽主題發佈的訊息中。RuleName 維度會包含規則的名稱。
RuleNotFound	找不到欲觸發的規則。RuleName 維度會包含規則的名稱。
RuleMessageThrottled	由於惡意行為或訊息數量超過規則引擎調節限制，而被規則引擎調節的訊息數量。RuleName 維度包含欲觸發規則的名稱。

規則動作指標

指標	描述
Success	成功規則動作呼叫次數。RuleName 維度包含可指定動作之規則的名稱。ActionType 維度包含已呼叫動作的類型。
Failure	失敗規則動作呼叫次數。RuleName 維度包含可指定動作之規則的名稱。RuleName 維度包含可指定動作之規則的名稱。ActionType 維度包含已呼叫動作的類型。

HTTP 動作特定指標

指標	描述
HttpInvalidUrl	在取代替換範本之後，如果端點 URL 未以 https:// 開頭，則會產生此指標。
HttpCode_5XX	如果來自下游 Web 服務/應用程式的回應狀態碼介於 500 到 599 之間，則會產生此指標。
HttpCode_4XX	如果來自下游 Web 服務/應用程式的回應狀態碼介於 400 到 499 之間，則會產生此指標。
HttpCode_Other	如果來自下游 Web 服務/應用程式的回應狀態碼不是 2xx、4xx 或 5xx，則會產生此指標。
HttpRequestTimeout	如果下游 Web 服務/應用程式未在請求逾時限制內傳回回應，則會產生此指標。如需更多詳細資訊，請參閱 AWS IoT 規則引擎 HTTP 動作限制 (p. 695)。
HttpUnknownHost	如果 URL 有效，但服務不存在或無法連線，則會產生此指標。

訊息中介裝置指標

指標	描述
Connect.AuthError	訊息中介裝置無法授權的連線要求數目。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Connect.ClientError	因 MQTT 訊息不符合 AWS IoT 限制中所定義的需求，而拒絕的連線要求數。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Connect.ClientIDThrottle	由於用戶端超過特定用戶端 ID 所允許之連線要求率而調節的連線要求數目。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Connect.ServerError	因發生內部錯誤而失敗的連線要求數目。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Connect.Success	訊息中介裝置成功的連線次數。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Connect.Throttle	因用戶端超過允許連線要求率而調節的連線要求數目。這可能是來自同一用戶端 ID 的帳戶層級連線速率或連線數量。Protocol 維度包含用來傳送 CONNECT 訊息的協定。
Ping.Success	訊息中介裝置收到的 ping 訊息數目。Protocol 維度包含用來傳送 ping 訊息的協定。
PublishIn.AuthError	訊息中介裝置無法授權的發佈要求數目。Protocol 維度包含用來發佈訊息的協定。
PublishIn.ClientError	因訊息不符合 AWS IoT 限制中所定義的需求，而遭訊息中介裝置拒絕的發佈要求數。Protocol 維度包含用來發佈訊息的協定。
PublishIn.ServerError	訊息中介裝置因發生內部錯誤而無法處理的發佈要求數目。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
PublishIn.Success	訊息中介裝置成功處理的發佈要求數目。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
PublishIn.Throttle	因用戶端超過允許傳入訊息率而調節的發佈要求數目。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
PublishOut.AuthError	AWS IoT 無法授權之訊息中介裝置所提出的發佈要求數目。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
PublishOut.ClientError	由訊息中介裝置發送，但因訊息不符合 AWS IoT 限制中所定義的需求而拒絕的發佈要求數。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
PublishOut.Success	訊息中介裝置成功發出的發佈要求數目。Protocol 維度包含用來傳送 PUBLISH 訊息的協定。
Subscribe.AuthError	由用戶端提出且無法授權的訂閱要求數目。Protocol 維度包含用來傳送 SUBSCRIBE 訊息的協定。

指標	描述
`Subscribe.ClientError`	因 `SUBSCRIBE` 訊息不符合 AWS IoT限制中所定義的需求，而拒絕的訂閱要求數。`Protocol` 維度包含用來傳送 `SUBSCRIBE` 訊息的協定。
`Subscribe.ServerError`	因發生內部錯誤而拒絕的訂閱要求數目。`Protocol` 維度包含用來傳送 `SUBSCRIBE` 訊息的協定。
`Subscribe.Success`	訊息中介裝置成功處理的訂閱要求數目。`Protocol` 維度包含用來傳送 `SUBSCRIBE` 訊息的協定。
`Subscribe.Throttle`	因用戶端超過允許訂閱要求率而調節的訂閱要求數目。`Protocol` 維度包含用來傳送 `SUBSCRIBE` 訊息的協定。
`Unsubscribe.ClientError`	因 `UNSUBSCRIBE` 訊息不符合 AWS IoT限制中所定義的需求，而拒絕的取消訂閱要求數。`Protocol` 維度包含用來傳送 `UNSUBSCRIBE` 訊息的協定。
`Unsubscribe.ServerError`	因發生內部錯誤而拒絕的取消訂閱要求數目。`Protocol` 維度包含用來傳送 `UNSUBSCRIBE` 訊息的協定。
`Unsubscribe.Success`	訊息中介裝置成功處理的取消訂閱要求數目。`Protocol` 維度包含用來傳送 `UNSUBSCRIBE` 訊息的協定。
`Unsubscribe.Throttle`	因用戶端超過允許取消訂閱要求率而拒絕的取消訂閱要求數目。`Protocol` 維度包含用來傳送 `UNSUBSCRIBE` 訊息的協定。

> **Note**
>
> 訊息中介裝置指標會顯示在 AWS IoT 主控台的 Protocol Metrics (協定指標) 下。

Device Shadow (裝置影子) 指標

指標	描述
`DeleteThingShadow.Accepted`	成功處理的 DeleteThingShadow 要求數目。`Protocol` 維度包含用來提出要求的協定。
`GetThingShadow.Accepted`	成功處理的 GetThingShadow 要求數目。`Protocol` 維度包含用來提出要求的協定。
`UpdateThingShadow.Accepted`	成功處理的 UpdateThingShadow 要求數目。`Protocol` 維度包含用來提出要求的協定。

> **Note**
>
> 裝置影子指標會顯示在 AWS IoT 主控台的 Protocol Metrics (協定指標) 下。

工作指標

指標	描述
`ServerError`	執行工作時產生的伺服器錯誤數目。`JobId` 維度包含工作的 ID。

指標	描述
ClientError	執行工作時產生的用戶端錯誤數目。JobId 維度包含工作的 ID。
QueuedJobExecutionTotalCount	針對指定工作，狀態為 QUEUED 的工作執行總數。JobId 維度包含工作的 ID。
InProgressJobExecutionTotalCount	針對指定工作，狀態為 IN_PROGRESS 的工作執行總數。JobId 維度包含工作的 ID。
FailedJobExecutionTotalCount	針對指定工作，狀態為 FAILED 的工作執行總數。JobId 維度包含工作的 ID。
SuccededJobExecutionTotalCount	針對指定工作，狀態為 SUCCESS 的工作執行總數。JobId 維度包含工作的 ID。
CanceledJobExecutionTotalCount	針對指定工作，狀態為 CANCELED 的工作執行總數。JobId 維度包含工作的 ID。
RejectedJobExecutionTotalCount	針對指定工作，狀態為 REJECTED 的工作執行總數。JobId 維度包含工作的 ID。
RemovedJobExecutionTotalCount	針對指定工作，狀態為 REMOVED 的工作執行總數。JobId 維度包含工作的 ID。
QueuedJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 QUEUED 之任務執行的數量。JobId 維度包含工作的 ID。
InProgressJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 IN_PROGRESS 之任務執行的數量。JobId 維度包含工作的 ID。
FailedJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 FAILED 之任務執行的數量。JobId 維度包含工作的 ID。
SuccededJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 SUCCESS 之任務執行的數量。JobId 維度包含工作的 ID。
CanceledJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 CANCELED 之任務執行的數量。(如需更多有關 CloudWatch 指標的資訊，請參閱 Amazon CloudWatch 指標。) JobId 維度包含工作的 ID。
RejectedJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 REJECTED 之任務執行的數量。JobId 維度包含工作的 ID。
RemovedJobExecutionCount	其狀態已在 CloudWatch 決定的期間內變更為 REMOVED 之任務執行的數量。JobId 維度包含工作的 ID。

Device Defender Audit 指標

指標	描述
NonCompliantResources	在檢查中發現不相容的資源數量。在每次稽核執行的檢查時，系統都會回報不合規的資源數量。
ResourcesEvaluated	評估符合的資源數目。系統會為每次執行的稽核檢查報告評估的資源數量。

Device Defender Detect 指標

指標	描述
`Violations`	自從上次執行評估起，發現新違反安全性描述檔行為的數目。系統會回報帳戶、特定安全性設定檔，以及特定安全性設定檔特定行為的新違規次數。
`ViolationsCleared`	自上次評估執行以來已解決的安全性設定檔行為的違規次數。系統會報告已解決的帳戶、特定安全性設定檔以及特定安全性設定檔特定行為的違規次數。
`ViolationsInvalidated`	自上次執行評估起，資訊不再可用的安全性描述檔行為違反之數目 (因為報告裝置停止報告，或因為某些原因不再受到監控)。系統會報告整個帳戶，特定安全性設定檔以及特定安全性設定檔特定行為的無效違規次數。

裝置佈建指標

指標	描述
CreateKeysAndCertificateFailed	呼叫 `CreateKeysAndCertificate` MQTT API 時發生的失敗次數。
GetDeviceCredentialsSucceeded	成功的 `GetDeviceCredentials` 呼叫次數。
GetDeviceCredentialsFailed	失敗的 `GetDeviceCredentials` 呼叫次數。
DeviceProvisioningFailed	無法佈建的裝置數量
DeviceProvisioningSucceeded	成功佈建的裝置數量。
RegisterThingFailed	呼叫 MQTT `RegisterThing` API 時發生的失敗次數。

指標的維度

指標會使用命名空間並為以下維度提供指標：

維度	描述
`ActionType`	觸發請求，並由規則指定的動作類型。
`Protocol`	用來提出要求的協定。有效值為：MQTT 或 HTTP
`RuleName`	要求所觸發規則的名稱。
`JobId`	正在監控其進度或訊息連線成功/失敗的工作 ID。
`CheckName`	其結果正受到監控的 Device Defender 稽核檢查的名稱。
`ScheduledAuditName`	其檢查結果正受到監控之 Device Defender 排程稽核的名稱。若回報的結果是依隨需執行的稽核結果，則此值為 `OnDemand`。
`SecurityProfileName`	其行為正受到監控之 Device Defender Detect 安全性描述檔的名稱。

維度	描述
`BehaviorName`	被監控的 Device Defender Detect 安全性設定檔行為名稱。
TemplateName	佈建範本的名稱。

如何使用 AWS IoT 指標？

AWS IoT 回報的指標可提供資訊，您可透過不同方式加以分析。以下使用案例是根據案例，您有十個項目，每天連接到網際網路。每個天：

- 十項實物約在同時連接到 AWS IoT。
- 每個真的訂閱主題篩選，然後等待一小時才能中斷。在此期間，事與其他應用程式通訊和進一步了解世界各地的狀態。
- 每個動作會根據其新發現有一些感知發佈資料 `UpdateThingShadow`。
- 每個動作 AWS IoT 中斷。

這些是協助您開始的建議，而不是完整清單。

- 如何每天都在我的物件連線失敗時收到通知？ (p. 208)
- 如何每天都在我的物件並未推送資料時收到通知？ (p. 209)
- 如何每天都在我的物件的影子更新遭拒時收到通知？ (p. 209)

建立 CloudWatch 警示來監控 AWS IoT

您可以建立 CloudWatch 警示，其在警示變更狀態時傳送 Amazon SNS 訊息。警示會監看所指定時段內的單一指標，並根據與多個時段內給定閾值相對的指標值來執行一或多個動作。此動作是傳送到 Amazon SNS 主題或 Auto Scaling 政策的通知。警示僅會針對持續狀態變更觸發動作。CloudWatch 警示不會只觸發動作，因為它們處於特定狀態；狀態必須已變更，並且已在指定的時段間受到維護。如需詳細資訊，請參閱設定 Amazon Simple Notification Service。

如何每天都在我的物件連線失敗時收到通知？

1. 建立 Amazon SNS 主題，`arn:aws:sns:us-east-1:123456789012:things-not-connecting-successfully`。
2. 建立警示。

```
Prompt>aws cloudwatch put-metric-alarm \
    --alarm-name ConnectSuccessAlarm \
    --alarm-description "Alarm when my Things don't connect successfully" \
    --namespace AWS/IoT \
    --metric-name Connect.Success \
    --dimensions Name=Protocol,Value=MQTT \
    --statistic Sum \
    --threshold 10 \
    --comparison-operator LessThanThreshold \
    --period 86400 \
    --unit Count \
    --evaluation-periods 1 \
    --alarm-actions arn:aws:sns:us-east-1:1234567890:things-not-connecting-successfully
```

3. 測試警示。

```
Prompt>aws cloudwatch set-alarm-state --alarm-name ConnectSuccessAlarm --state-reason
 "initializing" --state-value OK
```

```
Prompt>aws cloudwatch set-alarm-state --alarm-name ConnectSuccessAlarm --state-reason
 "initializing" --state-value ALARM
```

如何每天都在我的物件並未推送資料時收到通知？

1. 建立 Amazon SNS 主題，`arn:aws:sns:us-east-1:123456789012:things-not-publishing-data`。
2. 建立警示。

```
Prompt>aws cloudwatch put-metric-alarm \
    --alarm-name PublishInSuccessAlarm\
    --alarm-description "Alarm when my Things don't publish their data \
    --namespace AWS/IoT \
    --metric-name PublishIn.Success \
    --dimensions Name=Protocol,Value=MQTT \
    --statistic Sum \
    --threshold 10 \
    --comparison-operator LessThanThreshold \
    --period 86400 \
    --unit Count \
    --evaluation-periods 1 \
    --alarm-actions arn:aws:sns:us-east-1:1234567890:things-not-publishing-data
```

3. 測試警示。

```
Prompt>aws cloudwatch set-alarm-state --alarm-name PublishInSuccessAlarm --state-reason
 "initializing" --state-value OK
```

```
Prompt>aws cloudwatch set-alarm-state --alarm-name PublishInSuccessAlarm --state-reason
 "initializing" --state-value ALARM
```

如何每天都在我的物件的影子更新遭拒時收到通知？

1. 建立 Amazon SNS 主題，`arn:aws:sns:us-east-1:1234567890:things-shadow-updates-rejected`。

   如需詳細資訊，請參閱設定 Amazon Simple Notification Service。
2. 建立警示。

```
Prompt>aws cloudwatch put-metric-alarm \
    --alarm-name UpdateThingShadowSuccessAlarm \
    --alarm-description "Alarm when my Things Shadow updates are getting rejected" \
    --namespace AWS/IoT \
    --metric-name UpdateThingShadow.Success \
    --dimensions Name=Protocol,Value=MQTT \
    --statistic Sum \
    --threshold 10 \
    --comparison-operator LessThanThreshold \
    --period 86400 \
    --unit Count \
    --evaluation-periods 1 \
```

```
    --alarm-actions arn:aws:sns:us-east-1:1234567890:things-shadow-updates-rejected
```

3. 測試警示。

```
Prompt>aws cloudwatch set-alarm-state --alarm-name UpdateThingShadowSuccessAlarm --
state-reason "initializing" --state-value OK
```

```
Prompt>aws cloudwatch set-alarm-state --alarm-name UpdateThingShadowSuccessAlarm --
state-reason "initializing" --state-value ALARM
```

使用 CloudWatch Logs 進行監控

在透過訊息中介裝置和規則引擎從您的裝置傳遞時，AWS IoT 會傳送有關每條訊息的進度事件。若要檢視這些日誌，請務必將 AWS IoT 設定為會產生 CloudWatch 所使用的日誌。

如需更多有關 CloudWatch Logs 的資訊，請參閱 CloudWatch Logs。如需更多有關支援的 AWS IoT CloudWatch Logs 資訊，請參閱 CloudWatch Logs 項目格式 (p. 214)。

若要啟用 AWS IoT 記錄功能，您必須建立 IAM 角色、向 AWS IoT 註冊該角色，再設定 AWS IoT 記錄功能。

> **Note**
>
> 在啟用 AWS IoT 的記錄功能前，請確認您對 CloudWatch Logs 存取權限有所了解。具有 CloudWatch Logs 存取權限的使用者可從您的裝置查看除錯資訊。如需更多資訊，請參閱 Amazon CloudWatch Logs 的驗證與存取控制。

建立記錄角色

請使用 IAM 主控台來建立記錄角色。

1. 請選擇導覽面版上的 Roles (角色)，然後選擇 Create new role (建立新角色)。
2. 選擇 AWS Service Role (AWS 服務角色)，然後在服務角色類型中選擇 AWS IoT (AWS IoT)。
3. 請選擇 AWSIoTLogging 角色，然後選擇 Next Step (下一步)。

   在 Select your use case (選取您的使用案例) 下，選擇 IoT，然後選擇 Next: Permissions (下一步：許可)。
4. 在顯示連接到服務角色之政策的頁面上，選擇 Next: Tags (下一步：標籤)。
5. 選擇 Next:Review (下一步：檢閱)。
6. 輸入該角色的名稱與描述，然後選擇 Create role (建立角色)。

記錄角色政策

以下政策文件提供了角色政策及信任政策，用以許可 AWS IoT 代表您提交日誌至 CloudWatch。

> **Note**
>
> 這些文件是在您建立記錄角色時為您建立的。

角色政策：

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

```
                "Effect": "Allow",
                "Action": [
                    "logs:CreateLogGroup",
                    "logs:CreateLogStream",
                    "logs:PutLogEvents",
                    "logs:PutMetricFilter",
                    "logs:PutRetentionPolicy"
                 ],
                "Resource": [
                    "*"
                ]
            }
        ]
    }
```

信任政策：

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

日誌層級

日誌層級會指定產生的日誌類型。

ERROR

導致操作失敗的錯誤。

日誌內僅含 ERROR 資訊。

WARN

會導致系統內出現不一致，但可能不會造成操作失敗的情況。

日誌內僅含 ERROR 和 WARN 資訊。

INFO

物件流的高層級資訊。

日誌包含 INFO、ERROR、WARN 資訊。

DEBUG

可能有助於偵錯問題的資訊。

日誌包含 DEBUG、INFO、ERROR、WARN 資訊。

DISABLED

已停用所有記錄功能。

設定 AWS IoT 日誌記錄

您可使用 AWS IoT 主控台、set-v2-logging-options CLI 指令，或 SetV2LoggingOptions API 以啟動記錄。用於進行 API 呼叫的主體必須擁有您記錄角色的 傳送角色許可 (p. 256)。記錄角色會傳遞至 set-v2-logging-options 或 SetV2LoggingOptions 以做為 `roleARN` 參數。

您可將記錄設定為全域或細部。全域記錄會設定一個適用所有記錄的記錄層級，無論觸發日誌的是何種資源。精細記錄可讓您針對特定的資源或資源集設定記錄層級。目前僅支援事物群組。您可使用 AWS IoT 主控台、CLI 或 API 以啟用全域記錄。您必須使用 CLI 或 API 以啟用細部記錄。

全域記錄

請使用 `set-v2-logging-options` CLI 命令來設定您的帳戶的記錄選項。`set-v2-logging-options` 採用三個引數：

`--role-arn`

您的記錄角色 ARN。記錄角色將授予 AWS IoT 權限，以在 CloudWatch Logs 中寫入至您的記錄。

`--default-log-level`

欲使用的日誌層級。有效值為：ERROR、WARN、INFO、DEBUG、DISABLED

`--disable-all-logs | --no-disable-all-logs`

設定為 true 時 (`--disable-all-logs`) 停用所有日誌。預設 (未使用的參數) 為 false。

例如：

```
aws iot set-v2-logging-options \
    --role-arn arn:aws:iam::<your-aws-account-num>:role/<IoTLoggingRole> \
    --default-log-level <INFO>
```

您可以使用 `get-v2-logging-options`CLI 命令來取得目前的記錄選項。

> **Note**
>
> AWS IoT 持續支援較舊的指令 (`set-logging-options` 和 `get-logging-options`)，以在您的帳戶設定並取得全域記錄。請注意，這些指令在使用時所產生的記錄包含純文字，而非 JSON 有效酬載，且記錄延遲通常會更高。這些較舊指令的實施將不會進一步改進。我們建議您使用「v2」版本以設定您的記錄選項，並且，若可能的話，變更使用較舊版本的舊版應用程式。

使用 AWS IoT 主控台設定全域記錄

1. 登入至 AWS IoT 主控台。如需詳細資訊，請參閱 登入 AWS IoT 主控台。 (p. 4)。
2. 在左側的導覽窗格中，選擇 Settings (設定)。

3. 在 Settings (設定) 頁面的 Logs (記錄) 區段中，選擇 Edit (編輯)。Logs (記錄) 區段將顯示您的角色設定與詳細資訊層級。
4. 在 Configure role setting (設定角色設定) 頁面中，選擇您想在 CloudWatch 記錄中顯示描述詳細資料等級的詳細資訊層級。

5. 選擇 Select (選取) 以指定您之前建立的角色，或選擇 Create Role (建立角色) 以建立一個用於記錄的角色。
6. 選擇 Update (更新) 以儲存您的設定。

   審核您的 CloudWatch 記錄，以查看收集的資訊層級是否符合您的需求。如果不符需求，您可在稍後自由變更記錄層級。

精細記錄

微調記錄可讓您指定目標的記錄層級。目標是由資源類型和資源名稱來定義。目前 AWS IoT 支援以物件群組做為目標。精細記錄可設定物件群組的記錄層級。例如，您可能有一個名叫「電話」的物件群組，其中包含代表不同類型電話的物件。您可能接著建立了另一個名叫「行動電話」的物件群組，並將之當做「電話」物

件群組的子群組。精細記錄能夠為「電話」群組 (及任何子群組) 內的所有物件設定一個記錄層級，並為「行動電話」群組內的物件設定另一個記錄層級。在此例中，指派了兩種不同的記錄層級給「行動電話」群組中的物件—其一是「電話」物件群組的記錄層級—另一個則是「行動電話」物件群組的層級，但指定給子物件群組的記錄層級會覆蓋掉指定給母物件群組的記錄層級。

請使用 `set-v2-logging-options`CLI 命令，以啟用更精細記錄並設定預設的記錄層級。其會使用以下選擇性引數：

--role-arn

允許 IAM 寫入 AWS IoT 的 CloudWatch Logs 角色。若未指定，AWS IoT 會使用與您的帳戶有所關聯的記錄角色。記錄角色在建立時即會與您的帳戶建立關聯性。如需詳細資訊，請參閱 建立記錄角色 (p. 210)。

--default-log-level

未指定時所使用的記錄層級。有效值為：`DEBUG`、`INFO`、`ERROR`、`WARN` 和 `DISABLED`。

--停用-所有-日誌 | --無停用-所有-日誌

設定為 true 時 (`--disable-all-logs`) 停用所有日誌。預設 (未使用的參數) 為 false。

`get-v2-logging-options` CLI 命令會傳回設定的 IAM 記錄角色、預設的記錄層級和 `disableAllLogs` 的值。

請使用 `set-v2-logging-level`CLI 命令來設定目標的精細記錄功能。其會使用以下引數：

--log-target

內含您所設定記錄的資源類型 (欄位 `targetType`) 以及實體名稱 (欄位 `targetName`) 的 JSON 物件。AWS IoT 目前支援資源類型的 `THING_GROUP`。最多可以設定 10 個記錄的目標。

--log-level

為指定資源產生日誌時所用的記錄層級。有效值為：`DEBUG`、`INFO`、`ERROR`、`WARN`、`DISABLED`

請使用 `list-v2-logging-levels`CLI 命令來取得目前設定的精細記錄層級清單。請呼叫 `delete-v2-logging-level`CLI 命令來刪除記錄層級。請使用 `delete-v2-logging-level`命令來刪除精細記錄功能。

CloudWatch Logs 項目格式

AWS IoT 的每個元件會產生自己的日誌。每個日誌項目都有一個 `eventType`，會顯示出是哪項操作導致產生日誌。本節會介紹以下 AWS IoT 元件產生的記錄：

- 訊息中介裝置 (p. 215)
- Device Shadow 服務 (p. 219)
- 規則引擎 (p. 221)
- 任務 (p. 225)
- 裝置佈建日誌 (p. 229)

所有 CloudWatch Logs 均擁有以下的通用屬性：

timestamp

用戶端連線至 AWS IoT 訊息中介裝置時的 UNIX 時間戳記。

logLevel

使用的日誌層級。如需更多詳細資訊，請參閱 the section called “日誌層級” (p. 211)。

traceId

隨機產生的識別符，可用於建立某個特定請求的所有日誌的關聯性。

accountId

您的 AWS 帳戶 ID。

status

請求的狀態。

eventType

產生日誌的事件類型。每個事件的事件類型的值會列在以下部分。

訊息中介裝置日誌

AWS IoT 訊息中介裝置會針對以下事件產生日誌：

Connect Log

當 MQTT 用戶端連線時，AWS IoT 訊息中介裝置產生的 `Connect` 日誌。

More Information (1)

例如：

```
{
    "timestamp": "2017-08-10 15:37:23.476",
    "logLevel": "INFO",
    "traceId": "20b23f3f-d7f1-feae-169f-82263394fbdb",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "Connect",
    "protocol": "MQTT",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "sourceIp": "205.251.233.181",
    "sourcePort": 13490
}
```

除了 CloudWatch Logs 常見的屬性外，`Connect` 記錄項目也包含了以下屬性：

eventType

`Connect`用於連線日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

principalId

提出請求的委託人 ID。

sourceIp

請求來源所在的 IP 地址。

sourcePort

請求來源所在的埠口。

Subscribe Log

當 MQTT 用戶端訂閱某個主題時，AWS IoT 訊息中介裝置產生的 `Subscribe` 日誌。

More Information (2)

例如：

```
{
    "timestamp": "2017-08-10 15:39:04.413",
    "logLevel": "INFO",
    "traceId": "7aa5c38d-1b49-3753-15dc-513ce4ab9fa6",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "Subscribe",
    "protocol": "MQTT",
    "topicName": "$aws/things/MyThing/shadow/#",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "sourceIp": "205.251.233.181",
    "sourcePort": 13490
}
```

除了 CloudWatch Logs 常見的屬性外，`Subscribe` 記錄項目也包含了以下屬性：

eventType

`Subscribe`用於訂閱日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

topicName

訂閱主題的名稱。

clientId

提出請求的用戶端 ID。

principalId

提出請求的委託人 ID。

sourceIp

請求來源所在的 IP 地址。

sourcePort

請求來源所在的埠口。

Publish-In Log

當 AWS IoT 訊息中介裝置接收到 MQTT 訊息時，它將產生一個 `Publish-In` 記錄。

More Information (3)

例如：

```
{
        "timestamp": "2017-08-10 15:39:30.961",
        "logLevel": "INFO",
        "traceId": "672ec480-31ce-fd8b-b5fb-22e3ac420699",
        "accountId": "123456789012",
        "status": "Success",
```

```
        "eventType": "Publish-In",
        "protocol": "MQTT",
        "topicName": "$aws/things/MyThing/shadow/get",
        "clientId": "abf27092886e49a8a5c1922749736453",
        "principalId":
 "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
        "sourceIp": "205.251.233.181",
        "sourcePort": 13490
    }
```

除了 CloudWatch Logs 常見的屬性外，`Publish-In` 記錄項目也包含了以下屬性：

eventType

`Publish-In`當訊息中介裝置收到訊息。

status

請求的狀態。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

topicName

訂閱主題的名稱。

clientId

提出請求的用戶端 ID。

principalId

提出請求的委託人 ID。

sourceIp

請求來源所在的 IP 地址。

sourcePort

請求來源所在的埠口。

Publish-Out Log

當訊息中介裝置發佈一段 MQTT 訊息時，它會產生一個 `Publish-Out` 記錄。

More Information (4)

例如：

```
{
    "timestamp": "2017-08-10 15:39:30.961",
    "logLevel": "INFO",
    "traceId": "672ec480-31ce-fd8b-b5fb-22e3ac420699",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "Publish-Out",
    "protocol": "MQTT",
    "topicName": "$aws/things/MyThing/shadow/get",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "sourceIp": "205.251.233.181",
    "sourcePort": 13490
}
```

除了 CloudWatch Logs 常見的屬性外，`Publish-Out` 記錄項目也包含了以下屬性：

eventType

`Publish-Out`當訊息中介裝置發佈訊息。

status

請求的狀態。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

topicName

訂閱主題的名稱。

clientId

提出請求的用戶端 ID。

principalId

提出請求的委託人 ID。

sourceIp

請求來源所在的 IP 地址。

sourcePort

請求來源所在的埠口。

Disconnect Log

當 MQTT 用戶端斷開連線時，AWS IoT 訊息中介裝置產生的 `Disconnect` 日誌。

More Information (5)

例如：

```
{
    "timestamp": "2017-08-10 15:37:23.476",
    "logLevel": "INFO",
    "traceId": "20b23f3f-d7f1-feae-169f-82263394fbdb",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "Disconnect",
    "protocol": "MQTT",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "sourceIp": "205.251.233.181",
    "sourcePort": 13490
}
```

除了 CloudWatch Logs 常見的屬性外，`Disconnect` 記錄項目也包含了以下屬性：

eventType

`Disconnect`用於連線日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

principalId

提出請求的委託人 ID。

sourceIp

請求來源所在的 IP 地址。

sourcePort

請求來源所在的埠口。

Device Shadow 日誌

AWS IoT Device Shadow 服務會產生下列事件的日誌：

GetThingShadow Logs

當收到影子的 get 要求時，Device Shadow 服務會產生 `GetThingShadow`日誌。

More Information (6)

例如：

```
{
    "timestamp": "2017-08-09 17:56:30.941",
    "logLevel": "INFO",
    "traceId": "b575f19a-97a2-cf72-0ed0-c64a783a2504",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "GetThingShadow",
    "protocol": "MQTT",
    "deviceShadowName": "MyThing",
    "topicName": "$aws/things/MyThing/shadow/get"
}
```

除了 CloudWatch Logs 常見的屬性外，`GetThingShadow` 記錄項目也包含了以下屬性：

eventType

`GetThingShadow`用於 GetThingShadow 日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

deviceShadowName

所要求影子的名稱。

topicName

請求所發布的主題名稱。

UpdateThingShadow Logs

當收到更新裝置影子的要求時，Device Shadow 服務會產生 `UpdateThingShadow`日誌。

More Information (7)

例如：

```
{
    "timestamp": "2017-08-07 18:43:59.436",
```

```
    "logLevel": "INFO",
    "traceId": "d0074ba8-0c4b-a400-69df-76326d414c28",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "UpdateThingShadow",
    "protocol": "MQTT",
    "deviceShadowName": "Jack",
    "topicName": "$aws/things/Jack/shadow/update"
}
```

除了 CloudWatch Logs 常見的屬性外，`UpdateThingShadow` 記錄項目也包含了以下屬性：

eventType

`UpdateThingShadow`適用於更新影子日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

deviceShadowName

要更新的影子名稱。

topicName

請求所發布的主題名稱。

DeleteThingShadow Logs

當收到刪除裝置影子的要求時，Device Shadow 服務會產生 `DeleteThingShadow`日誌。

More Information (8)

例如：

```
{
    "timestamp": "2017-08-07 18:47:56.664",
    "logLevel": "INFO",
    "traceId": "1a60d02e-15b9-605b-7096-a9f584a6ad3f",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "DeleteThingShadow",
    "protocol": "MQTT",
    "deviceShadowName": "Jack",
    "topicName": "$aws/things/Jack/shadow/delete"
}
```

除了 CloudWatch Logs 常見的屬性外，`DeleteThingShadow` 記錄項目也包含了以下屬性：

eventType

`DeleteThingShadow`用於 DeleteThingShadow 日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

deviceShadowName

要更新的影子名稱。

topicName

請求所發布的主題名稱。

Rules Engine 日誌

AWS IoT 規則引擎服務會針對以下事件產生日誌：

Rule Match Logs

當訊息中介裝置符合規則時，AWS IoT 規則引擎會產生 `RuleMatch` 日誌。

More Information (9)

例如：

```
{
    "timestamp": "2017-08-10 16:32:46.002",
    "logLevel": "INFO",
    "traceId": "30aa7ccc-1d23-0b97-aa7b-76196d83537e",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "RuleMatch",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "topicName": "rules/test",
    "ruleName": "JSONLogsRule",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167"
}
```

除了 CloudWatch Logs 常見的屬性外，`RuleMatch` 記錄項目也包含了以下屬性：

eventType

`RuleMatch`適用於規則符合日誌。

clientId

提出請求的用戶端 ID。

topicName

訂閱主題的名稱。

ruleName

符合的規則名稱。

principalId

提出請求的委託人 ID。

Function Execution Logs

當規則的 SQL 查詢呼叫外部函數時，規則引擎會產生 `FunctionExecution`記錄。當規則動作向 AWS IoT 或其他 Web 服務提出 HTTP 請求時，會呼叫外部函數 (例如呼叫 `get_thing_shadow` 或 `machinelearning_predict`)。

More Information (10)

`FunctionExecution` 記錄看起來會像以下所示：

```
{
    "timestamp": "2017-07-13 18:33:51.903",
    "logLevel": "DEBUG",
    "traceId": "180532b7-0cc7-057b-687a-5ca1824838f5",
    "status": "Success",
    "eventType": "FunctionExecution",
    "clientId": "N/A",
```

```
    "topicName":"rules/test",
    "ruleName": "ruleTestPredict",
    "ruleAction": "MachinelearningPredict",
    "resources": {
        "ModelId": "predict-model"
    },
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167"
}
```

除了 CloudWatch Logs 常見的屬性外，`FunctionExecution` 記錄項目也包含了以下屬性：

eventType

`FunctionExecution`適用於規則符合日誌。

clientId

`N/A`適用於 `FunctionExecution` 日誌。

topicName

訂閱主題的名稱。

ruleName

符合的規則名稱。

resources

該規則動作所使用的資源集合。

principalId

提出請求的委託人 ID。

Starting Execution Logs

在 AWS IoT 規則引擎開始觸發規則動作時，它會產生 `StartingExecution` 記錄。

More Information (11)

例如：

```
{
    "timestamp": "2017-08-10 16:32:46.002",
    "logLevel": "DEBUG",
    "traceId": "30aa7ccc-1d23-0b97-aa7b-76196d83537e",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "StartingRuleExecution",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "topicName": "rules/test",
    "ruleName": "JSONLogsRule",
    "ruleAction": "RepublishAction",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167"
}
```

除了 CloudWatch Logs 常見的屬性外，`StartingExecution` 記錄項目也包含了以下屬性：

eventType

`StartingRuleExecution`適用於啟動規則執行日誌。

clientId

提出請求的用戶端 ID。

topicName

訂閱主題的名稱。

ruleName

符合的規則名稱。

ruleAction

受到觸發的動作名稱。

principalId

提出請求的委託人 ID。

Rule Execution Logs

在 AWS IoT 規則引擎觸發規則動作時，它會產生 `RuleExecution` 記錄。

More Information (12)

例如：

```
{
    "timestamp": "2017-08-10 16:32:46.070",
    "logLevel": "INFO",
    "traceId": "30aa7ccc-1d23-0b97-aa7b-76196d83537e",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "RuleExecution",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "topicName": "rules/test",
    "ruleName": "JSONLogsRule",
    "ruleAction": "RepublishAction",
    "resources": {
        "RepublishTopic": "rules/republish"
    },
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167"
}
```

除了 CloudWatch Logs 常見的屬性外，`RuleExecution` 記錄項目也包含了以下屬性：

eventType

`RuleExecution`適用於規則執行日誌。

clientId

提出請求的用戶端 ID。

topicName

訂閱主題的名稱。

ruleName

符合的規則名稱。

ruleAction

受到觸發的動作名稱。

resources

該規則動作所使用的資源集合。

principalId

提出請求的委託人 ID。

Rule Not Found Logs

在 AWS IoT 規則引擎找不到擁有指定名稱的規則時，它會產生 `RuleNotFound` 錯誤記錄。

More Information (13)

例如：

```
{
    "timestamp": "2017-10-04 19:25:46.070",
    "logLevel": "ERROR",
    "traceId": "30aa7ccc-1d23-0b97-aa7b-76196d83537e",
    "accountId": "123456789012",
    "status": "Failure",
    "eventType": "RuleNotFound",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "topicName": "$aws/rules/example_rule",
    "ruleName": "example_rule",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "reason": "RuleNotFound",
    "details": "Rule example_rule not found"
}
```

除了 CloudWatch Logs 常見的屬性外，`RuleNotFound` 記錄項目也包含了以下屬性：

eventType

`RuleNotFound` 代表 rule-not-found (找不到規則) 記錄。

clientId

提出請求的用戶端 ID。

topicName

已發佈的主題名稱。

ruleName

找不到的規則名稱。

principalId

提出請求的委託人 ID。

reason

"RuleNotFound" 字串。

details

錯誤的簡要說明。

Rule Message Throttled Logs

當訊息受限時，AWS IoT 規則引擎會產生 `RuleMessageThrottled` 錯誤記錄。

More Information (14)

例如：

```
{
    "timestamp": "2017-10-04 19:25:46.070",
    "logLevel": "ERROR",
    "traceId": "30aa7ccc-1d23-0b97-aa7b-76196d83537e",
    "accountId": "123456789012",
    "status": "Failure",
    "eventType": "RuleMessageThrottled",
    "clientId": "abf27092886e49a8a5c1922749736453",
    "topicName": "$aws/rules/example_rule",
    "ruleName": "example_rule",
    "principalId": "145179c40e2219e18a909d896a5340b74cf97a39641beec2fc3eeafc5a932167",
    "reason": "RuleExecutionThrottled",
    "details": "Message for Rule example_rule throttled"
}
```

除了 CloudWatch Logs 常見的屬性外，`RuleMessageThrottled` 記錄項目也包含了以下屬性：

eventType

`RuleMessageThrottled` 是規則訊息調節日誌。

clientId

提出請求的用戶端 ID。

topicName

已發佈的主題名稱。

ruleName

欲觸發的規則名稱。

principalId

提出請求的委託人 ID。

reason

"RuleMessageThrottled" 字串。

details

錯誤的簡要說明。

工作日誌

AWS IoT 任務服務會針對以下事件產生日誌。記錄會在從裝置收到 MQTT 或 HTTP 請求時產生。

Get Pending Job Execution Logs

當任務服務收到執行任務的請求時，AWS IoT 任務服務會產生 `GetJobExecution` 日誌。More Information (16)

例如：

```
{
    "timestamp": "2018-06-13 17:45:17.197",
    "logLevel": "DEBUG",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "GetPendingJobExecution",
    "protocol": "MQTT",
    "clientId": "299966ad-54de-40b4-99d3-4fc8b52da0c5",
```

```
    "topicName": "$aws/things/299966ad-54de-40b4-99d3-4fc8b52da0c5/jobs/get",
    "clientToken": "24b9a741-15a7-44fc-bd3c-1ff2e34e5e82",
    "details": "The request status is SUCCESS."
}
```

除了 CloudWatch Logs 常見的屬性外，`GetPendingJobExecution` 記錄項目也包含了以下屬性：

eventType

`GetPendingJobExecution` 適用於取得待定任務執行日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

topicName

訂閱主題的名稱。

clientToken

用以確保請求冪等的，是一個唯一、區分大小寫的識別符。如需詳細資訊，請參閱如何確保冪等。

details

任務裝置的其他資訊。

Describe Job Execution Logs

當任務服務收到說明任務執行的請求時，AWS IoT 任務服務會產生 `DescribeJobExecution` 日誌。More Information (17)

例如：

```
{
    "timestamp": "2017-08-10 19:13:22.841",
    "logLevel": "DEBUG",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "DescribeJobExecution",
    "protocol": "MQTT",
    "clientId": "thingOne",
    "jobId": "002",
    "topicName": "$aws/things/thingOne/jobs/002/get",
    "clientToken": "myToken",
    "details": "The request status is SUCCESS."
}
```

除了 CloudWatch Logs 常見的屬性外，`GetJobExecution` 記錄項目也包含了以下屬性：

eventType

`DescribeJobExecution`適用於說明工作執行日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

jobId

執行工作的工作 ID。

topicName

用來提出請求的主題。

clientToken

用以確保請求冪等的，是一個唯一、區分大小寫的識別符。如需詳細資訊，請參閱如何確保冪等。

details

任務裝置的其他資訊。

Update Job Execution Logs

當任務服務收到更新說明任務的請求時，AWS IoT 任務服務會產生 `UpdateJobExecution` 日誌。More Information (18)

例如：

```
{
    "timestamp": "2017-08-10 19:25:14.758",
    "logLevel": "DEBUG",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "UpdateJobExecution",
    "protocol": "MQTT",
    "clientId": "thingOne",
    "jobId": "002",
    "topicName": "$aws/things/thingOne/jobs/002/update",
    "clientToken": "myClientToken",
    "versionNumber": "1",
    "details": "The destination status is IN_PROGRESS. The request status is SUCCESS."
}
```

除了 CloudWatch Logs 常見的屬性外，`UpdateJobExecution` 記錄項目也包含了以下屬性：

eventType

`UpdateJobExecution`適用於更新工作執行日誌。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

jobId

執行工作的工作 ID。

topicName

用來提出請求的主題。

clientToken

用以確保請求冪等的，是一個唯一、區分大小寫的識別符。如需詳細資訊，請參閱如何確保冪等。

versionNumber

工作執行的版本。

details

任務裝置的其他資訊。

Start Next Pending Job Execution Logs

在它接收到啟動下一個擱置任務執行的請求時，AWS IoT 任務服務會產生 `StartNextPendingJobExecution` 記錄。

More Information (19)

例如：

```
{
    "timestamp": "2018-06-13 17:49:51.036",
    "logLevel": "DEBUG",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "StartNextPendingJobExecution",
    "protocol": "MQTT",
    "clientId": "95c47808-b1ca-4794-bc68-a588d6d9216c",
    "topicName": "$aws/things/95c47808-b1ca-4794-bc68-a588d6d9216c/jobs/start-next",
    "clientToken": "bd7447c4-3a05-49f4-8517-dd89b2c68d94",
    "details": "The request status is SUCCESS."
}
```

除了 CloudWatch Logs 常見的屬性外，`StartNextPendingJobExecution` 記錄項目也包含了以下屬性：

eventType

`StartNextPendingJobExecution` 代表 start-next-pending-job (啟動下一個擱置任務) 執行記錄。

通訊協定

提出請求時使用的通訊協定。有效值為 `MQTT`或 `HTTP`。

clientId

提出請求的用戶端 ID。

topicName

用來提出請求的主題。

clientToken

用以確保請求冪等的，是一個唯一、區分大小寫的識別符。如需詳細資訊，請參閱如何確保冪等。

details

任務裝置的其他資訊。

Report Final Job Execution Count Logs

當工作完成時，AWS IoT 工作服務會產生 `ReportFinalJobExecutionCount` 日誌。

More Information (20)

例如：

```
{
    "timestamp": "2017-08-10 19:44:16.776",
```

```
    "logLevel": "INFO",
    "accountId": "123456789012",
    "status": "Success",
    "eventType": "ReportFinalJobExecutionCount",
    "jobId": "002",
    "details": "Job 002 completed. QUEUED job execution count: 0 IN_PROGRESS job
 execution count: 0 FAILED job execution count: 0 SUCCEEDED job execution count: 1
 CANCELED job execution count: 0 REJECTED job execution count: 0 REMOVED job execution
 count: 0"
}
```

除了 CloudWatch Logs 常見的屬性外，`ReportFinalJobExecutionCount` 記錄項目也包含了以下屬性：

eventType

`ReportFinalJobExecutionCount` 代表 report-final-job-execution-count (報告最終任務執行計數) 記錄。

jobId

執行工作的工作 ID。

details

任務裝置的其他資訊。

裝置佈建日誌

AWS IoT 裝置佈建服務會產生下列事件的日誌：

GetDeviceCredentials Logs

AWS IoT 裝置佈建服務會在用戶端呼叫 `GetDeviceCredential` 時產生日誌。

more info (16)

例如：

```
{
  "timestamp" : "2019-02-20 20:31:22.932",
  "logLevel" : "INFO",
  "traceId" : "8d9c016f-6cc7-441e-8909-7ee3d5563405",
  "accountId" : "123456789101",
  "status" : "Success",
  "eventType" : "GetDeviceCredentials",
  "deviceCertificateId" :
 "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
  "details" : "Additional details about this log."
}
```

ProvisionDevice Logs

AWS IoT 裝置佈建服務會在用戶端呼叫 `ProvisionDevice` 時產生日誌。

more info (16)

例如：

```
{
  "timestamp" : "2019-02-20 20:31:22.932",
  "logLevel" : "INFO",
```

```
  "traceId" : "8d9c016f-6cc7-441e-8909-7ee3d5563405",
  "accountId" : "123456789101",
  "status" : "Success",
  "eventType" : "ProvisionDevice",
  "provisioningTemplateName" : "myTemplate",
  "deviceCertificateId" :
 "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
  "details" : "Additional details about this log."
 }
```

檢視日誌

查看您的日誌

1. 瀏覽 https://console.aws.amazon.com/cloudwatch/。在導覽窗格中，選擇日誌。
2. 在 Filter (篩選條件) 文字方塊中，輸入 **`AWSIotLogsV2`**，然後按下 Enter 鍵。
3. 請按兩下 `AWSIotLogsV2`日誌群組。
4. 請選擇 Search Log Group (搜尋日誌群組)。會顯示為您的帳戶所產生的完整 AWS IoT 日誌清單。
5. 選擇展開圖示可觀看個別的串流。

您也可以在 Filter events (篩選條件事件) 中輸入一個查詢。以下是一些可以試試看的有趣查詢：

- `{ $.logLevel = "INFO" }`

  尋找日誌層級為 `INFO`的所有日誌。
- `{ $.status = "Success" }`

  尋找日誌狀態為 `Success`的所有日誌。
- `{ $.status = "Success" && $.eventType = "GetThingShadow" }`

  尋找狀態為 `Success`且事件類型為 `GetThingShadow` 的所有日誌。

如需有關建立篩選表達式的詳細資訊，請參閱 CloudWatch Logs查詢。

使用 AWS CloudTrail 記錄 AWS IoT API 呼叫

AWS IoT 與 AWS CloudTrail 服務整合，可記錄 AWS 中使用者、角色或 AWS IoT 採取的動作。CloudTrail 會將 AWS IoT 所有 API 呼叫擷取為事件，包括從 AWS IoT 主控台的呼叫，和從程式碼對 AWS IoT API 的呼叫。如果您建立追蹤記錄，就可以持續傳送 CloudTrail 事件至 Amazon S3 儲存貯體，包括 AWS IoT 的事件。如果您不設定追蹤記錄，仍然可以透過 CloudTrail 主控台中的 Event history (事件歷史記錄) 檢視最新的事件。透過使用 CloudTrail 所收集的資訊，您可以判斷向 AWS IoT 所提出的請求、提出請求的 IP 位址、由誰提出請求、何時提出請求以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱 AWS CloudTrail User Guide。

CloudTrail 中的 AWS IoT 資訊

當您建立帳戶時，系統會在您的 AWS 帳戶中啟用 CloudTrail。當 AWS IoT 中發生活動時，該活動會記錄在 CloudTrail 事件中，其他 AWS 服務事件則記錄於 Event history (事件歷程記錄)。您可以檢視、搜尋和下載 AWS 帳戶的最新事件。如需詳細資訊，請參閱使用 CloudTrail 事件歷程記錄檢視事件。

如需您 AWS 帳戶中正在進行事件的記錄 (包含 AWS IoT 的事件)，請建立線索。追蹤記錄可讓 CloudTrail 將日誌檔案交付到 Amazon S3 儲存貯體。依預設，當您在主控台建立追蹤時，該追蹤會套用到所有 AWS 區域。該追蹤會記錄來自 AWS 分割區所有 AWS 區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存

貯體。您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中所收集的事件資料。如需詳細資訊，請參閱：

- 建立追蹤的概觀
- CloudTrail 支援的服務和整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 接收多個區域的 CloudTrail 日誌檔案及接收多個帳戶的 CloudTrail 日誌檔案

> **Note**
>
> CloudTrail 不會記錄 AWS IoT 資料的平面動作 (裝置端)。使用 CloudWatch 監控這些動作。

AWS IoT 控制平面動作是由 CloudTrail 所記錄。舉例而言，對 CreateThing、ListThings、ListTopicRules 幾節的呼叫均會在 CloudTrail 日誌檔案中產生項目。

每一筆事件或記錄項目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：

- 該請求是否使用根或 IAM 使用者登入資料提出。
- 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登入資料.
- 該請求是否由另一項 AWS 服務提出。

如需更多資訊，請參閱 CloudTrail userIdentity 元素。AWS IoT 動作在 AWS IoTAPI 參考資料中有詳細紀載。

了解 AWS IoT 日誌檔項目

追蹤記錄是一種組態，能讓事件以日誌檔案的形式交付至您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一個或多個日誌項目。事件代表從任何來源的單一請求，並包含有關請求的動作、動作的日期和時間、請求參數等資訊。CloudTrail 日誌檔不是公有 API 呼叫的排序堆疊追蹤記錄，因此不會出現以任何特定順序顯示。

以下範例顯示的是展示 CloudTrail 動作的 `AttachPolicy` 日誌項目。


```
{
    "timestamp":"1460159496",
    "AdditionalEventData":"",
    "Annotation":"",
    "ApiVersion":"",
    "ErrorCode":"",
    "ErrorMessage":"",
    "EventID":"8bff4fed-c229-4d2d-8264-4ab28a487505",
    "EventName":"AttachPolicy",
    "EventTime":"2016-04-08T23:51:36Z",
    "EventType":"AwsApiCall",
    "ReadOnly":"",
    "RecipientAccountList":"",
    "RequestID":"d4875df2-fde4-11e5-b829-23bf9b56cbcd",
    "RequestParamters":{
        "principal":"arn:aws:iot:us-
east-1:123456789012:cert/528ce36e8047f6a75ee51ab7beddb4eb268ad41d2ea881a10b67e8e76924d894",
        "policyName":"ExamplePolicyForIoT"
    },
    "Resources":"",
    "ResponseElements":"",
    "SourceIpAddress":"52.90.213.26",
    "UserAgent":"aws-internal/3",
    "UserIdentity":{
```

```
        "type":"AssumedRole",
        "principalId":"AKIAI44QH8DHBEXAMPLE",
        "arn":"arn:aws:sts::12345678912:assumed-role/iotmonitor-us-east-1-beta-
InstanceRole-1C5T1YCYMHPYT/i-35d0a4b6",
        "accountId":"222222222222",
        "accessKeyId":"access-key-id",
        "sessionContext":{
            "attributes":{
                "mfaAuthenticated":"false",
                "creationDate":"Fri Apr 08 23:51:10 UTC 2016"
            },
            "sessionIssuer":{
                "type":"Role",
                "principalId":"AKIAI44QH8DHBEXAMPLE",
                "arn":"arn:aws:iam::123456789012:role/executionServiceEC2Role/iotmonitor-
us-east-1-beta-InstanceRole-1C5T1YCYMHPYT",
                "accountId":"222222222222",
                "userName":"iotmonitor-us-east-1-InstanceRole-1C5T1YCYMHPYT"
            }
        },
        "invokedBy":{
            "serviceAccountId":"111111111111"
        }
    },
    "VpcEndpointId":""
}
```


AWS IoT Core 的合規驗證

在多個 AWS 合規計劃中，第三方稽核人員會評估 AWS IoT Core 的安全與合規。這些計劃包括 SOC、PCI、FedRAMP、HIPAA 等等。

如需特定合規計劃範圍內的 AWS 服務清單，請參閱合規計劃內的 AWS 服務。如需一般資訊，請參閱 AWS 合規計劃。

您可使用 AWS Artifact 下載第三方稽核報告。如需詳細資訊，請參閱在 AWS Artifact 中下載報告。

您使用 AWS IoT 時的合規責任取決於資料的敏感度、您的公司的合規目標，以及適用的法律和法規。AWS 提供以下資源協助您處理合規事宜：

- 安全與合規快速入門指南 – 這些部署指南討論在 AWS 上部署以安全及合規為重心之基準環境的架構考量和步驟。
- HIPAA 安全與合規架構白皮書 – 本白皮書說明公司可如何運用 AWS 來建立 HIPAA 合規的應用程式。
- AWS 合規資源 – 這組手冊和指南可能適用於您的產業和位置。
- AWS Config – 此 AWS 服務可評定資源組態與內部實務、業界準則和法規的合規狀態。
- AWS Security Hub – 此 AWS 服務可供您檢視 AWS 中的安全狀態，可助您檢查是否符合安全產業標準和最佳實務。

AWS IoT Core 中的彈性

AWS 全球基礎設施是以 AWS 區域與可用區域為中心建置的。AWS 區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援網路連線相互連結。透過可用區域，您所設計與操作的應用程式和資料庫，就能夠在可用區域之間自動容錯移轉，而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域與可用區域的詳細資訊，請參閱 AWS 全球基礎設施。

AWS IoT Core 會將裝置的相關資訊儲存在裝置登錄中。它也會存放 CA 憑證、裝置憑證和裝置影子資料。發生硬體或網路故障時，這些資料不會自動複寫。裝置登錄更新時，AWS IoT Core 會發佈 MQTT 事件。您可以使用這些訊息來備份您的登錄資料，並將其儲存在某處，例如 DynamoDB 資料表。您必須負責儲存 AWS IoT Core 為您建立或您自行建立的憑證。裝置影子會存放裝置的狀態相關資料，並且可以在裝置重新連上線時重新傳送。

AWS IoT 中的基礎設施安全

AWS IoT 為受管服務集合，受到 Amazon Web Services：安全程序概觀白皮書所述的 AWS 全球網路安全程序保護。

您可使用 AWS 發佈的 API 呼叫，透過網路存取 AWS IoT。用戶端必須支援 Transport Layer Security (TLS) 1.0 或更新版本。建議使用 TLS 1.2 或更新版本。用戶端也必須支援具備完美轉送私密 (PFS) 的密碼套件，例如臨時 Diffie-Hellman (DHE) 或橢圓曲線臨時 Diffie-Hellman (ECDHE)。現代系統 (如 Java 7 和更新版本) 大多會支援這些模式。如需詳細資訊，請參閱AWS IoT 中的傳輸安全性 (p. 178)。

請求必須使用存取金鑰 ID 和與 IAM 委託人相關聯的私密存取金鑰來簽署。或者，您可以使用 AWS Security Token Service (AWS STS) 來產生暫時安全登入資料來簽署請求。

AWS IoT Core 中的漏洞分析和管理

IoT 機群可由大量的裝置組成具有多樣化的功能、長時間在線上，並且散佈在多個地理位置。這些特點使叢集設定複雜且極易出錯。由於裝置通常受限於運算能力、記憶體和儲存功能，因此限制了裝置本身的加密和其他形式的安全性的使用。此外，裝置通常會使用具有已知漏洞的軟體。這些因素讓 IoT 機群成為對駭客極具吸引力的目標，且難以持續保護裝置機群。

AWS IoT Device Defender 藉由提供識別安全性問題和從最佳實務偏離的工具來因應這些挑戰。您可以使用 AWS IoT Device Defender 來分析、稽核和監控連線的裝置，以偵測異常行為，並降低安全風險。AWS IoT Device Defender 可以稽核裝置叢集，以確保他們遵守安全最佳實務並偵測裝置上的異常行為。它可讓您的 AWS IoT 裝置機群實施一致的安全政策，並在裝置受到入侵時快速回應。如需詳細資訊，請參閱AWS IoT Device Defender (p. 506)。

AWS IoT Core 中的安全性最佳實務

本節包含有關 AWS IoT Core 的安全性最佳實務。如需詳細資訊，請參閱 IoT 解決方案的十大安全黃金規則。

在 AWS IoT 中保護 MQTT 連線

AWS IoT Core 是一項受管的雲端服務，可讓連線裝置輕鬆安全地與雲端應用程式及其他裝置互動。AWS IoT Core 支援 HTTP、WebSocket，及 MQTT，是一種輕量型通訊協定，經過專門設計可容忍間歇連線。如果您是使用 MQTT 連線到 AWS IoT 訊息中介裝置，則您的每個連線都必須與一個識別符 (稱為用戶端 ID) 關聯。MQTT 用戶端 ID 是獨一無二的，可識別 MQTT 連線。如果使用另一個已經佔用的用戶端 ID 來建立新連線，則 AWS IoT 訊息中介裝置會捨棄舊的連線以供新連線使用。您可以使用 AWS IoT 中的授權功能，以確保您機群中的裝置不會意外獲得授權，以已在使用的用戶端 ID，透過開放 MQTT 連線來中斷其他裝置的連線。每個 AWS 帳戶和每個 AWS 區域中的用戶端 ID 都必須是唯一的，因此您無需在 AWS 帳戶之外或 AWS 帳戶的區域之間強制執行用戶端 ID 的全域唯一性。

在您的裝置機群上捨棄 MQTT 連線的影響和嚴重性，取決於多個因素。其中包括：

- 您的使用案例 (例如，您的裝置傳送哪些資料到 AWS IoT，以及資料的多寡和傳送資料的頻率)。

- 您的 MQTT 用戶端組態 (例如，自動重新連線設定、關聯的退避計時，以及 MQTT 持久性工作階段的使用)。
- 裝置資源限制。
- 連線中斷的根本原因，以及其積極性和持久性。

為了避免用戶端 ID 衝突及其潛在的負面影響，確保每個裝置或行動應用程式都有一個 AWS IoT 或 IAM 政策，以限制哪些用戶端 ID 允許用於 MQTT 連線到 AWS IoT 訊息中介裝置。

您機群中的所有裝置都必須具備僅擁有授權預期動作的權限，包括但不限於，例如以特定範圍與內容發佈訊息或訂閱主題的 AWS IoT MQTT 動作。每個使用案例使用的特定許可政策各不相同，因此您需要確定最符合您業務和安全要求的許可政策。

為了簡化許可政策的建立和管理，您可以使用 AWS IoT 政策變數 (p. 149) 和 IAM 政策變數。政策變數可以放置在政策中，當政策接受評估時，該變數可由來自裝置的請求值取代。透過使用政策變數，您可以建立一個單一政策，以授與許可給多個裝置。您可以根據您的 AWS IoT 帳戶組態、身分驗證機制，以及用於連接到 AWS IoT 訊息中介裝置的網路通訊協定，依使用案例識別相關的政策變數。不過，若要撰寫最佳的許可政策，您需要考慮您使用案例的特定情形和威脅模型。

例如，如果您已在 AWS IoT 登錄檔中註冊您的裝置，您可以使用 AWS IoT 政策中實物政策變數，以根據實物屬性 (例如實物名稱、實物類型和實物屬性值) 來授與或拒絕許可。當物件連接至 AWS IoT 時，將自 MQTT Connect 訊息內的用戶端 ID 取得物件名稱。當物件透過 MQTT 使用 TLS 交互身分驗證連接至 AWS IoT，或經 WebSocket 通訊協定的 MQTT 使用已驗證的 Amazon Cognito 身分，物件政策變數會被取代。您可以使用 `AttachThingPrincipal` API 來連接憑證和驗證的 Amazon Cognito 身分到實物。`iot:Connection.Thing.ThingName` 是一種實用的實物政策變數，可強制執行用戶端 ID 限制。以下範例 AWS IoT 政策需要註冊的實物名稱，以作為用戶端 ID，以供 MQTT 連接到 AWS IoT 訊息中介裝置：

```
{
    "Version":"2012-10-17",
    "Statement":[
    {
        "Effect":"Allow",
        "Action":"iot:Connect",
        "Resource":[
            "arn:aws:iot:us-east-1:123456789012:client/${iot:Connection.Thing.ThingName}"
        ]
    }
    ]
}
```

如果您想要識別持續的用戶端 ID 衝突，您可以啟用並使用適用於 AWS IoT 的CloudWatch Logs。針對每個因為用戶端 ID 衝突而導致被 AWS IoT 訊息中介程式中斷的 MQTT 的連線，會產生類似以下的日誌記錄：

```
{
    "timestamp": "2019-04-28 22:05:30.105",
    "logLevel": "ERROR",
    "traceId": "02a04a93-0b3a-b608-a27c-1ae8ebdb032a",
    "accountId": "123456789012",
    "status": "Failure",
    "eventType": "Disconnect",
    "protocol": "MQTT",
    "clientId": "clientId01",
    "principalId": "1670fcf6de55adc1930169142405c4a2493d9eb5487127cd0091ca0193a3d3f6",
    "sourceIp": "203.0.113.1",
    "sourcePort": 21335,
    "reason": "DUPLICATE_CLIENT_ID",
    "details": "A new connection was established with the same client ID"
}
```

您可以使用 CloudWatch Logs 篩選條件 (例如 `{$.reason= "DUPLICATE_CLIENT_ID" }`) 來搜尋用戶端 ID 衝突的執行個體，或設定CloudWatch 指標篩選條件與對應的 CloudWatch 警示，以供持續監控和報告。

您可以使用 AWS IoT Device Defender 來識別過於寬鬆的 AWS IoT 和 IAM 政策。AWS IoT Device Defender 也提供了稽核檢查，如果您機群中的多個裝置使用了相同的用戶端 ID 連接到 AWS IoT 訊息中介裝置，該檢查將會通知您。

另請參閱

- AWS IoT Core
- AWS IoT 的安全性功能
- AWS IoT 政策變數
- IAM 政策變數
- Amazon Cognito Identity
- AWS IoT 登錄檔
- AWS IoT Device Defender
- 適用於 AWS IoT 的CloudWatch Logs

讓裝置的時鐘保持同步

在裝置上保持準確的時間是很重要的。X.509 憑證具有到期資料和時間。裝置上的時鐘用來驗證伺服器憑證是否仍然有效。如果您正在建置商用 IoT 裝置，請記住您的產品在售出前可能會長期存放。在這段期間，即時時鐘可能會漂移，電池可能會放電，因此在工廠設定時間並不足夠。

對大多數系統而言，這表示裝置的軟體需要包含 NTP (網路時間通訊協定) 用戶端，並且應該等到與 NTP 伺服器同步後再嘗試與 AWS IoT Core 連線。如果無法這麼做，系統應該提供方法讓使用者能夠設定裝置的時間，以便讓後續連線成功。

一旦裝置與 NTP 伺服器同步，就可以開啟與 AWS IoT Core 的連線。允許多少時鐘誤差，取決於您嘗試使用連線進行什麼操作。

驗證伺服器憑證

裝置與 AWS IoT 互動的第一件事，就是開啟安全連線。當您將裝置連線到 AWS IoT，您需要確保您正在與 AWS IoT 進行通訊，而非其他冒充 AWS IoT 的伺服器。每個 AWS IoT 伺服器都會使用為 `iot.amazonaws.com` 網域核發的憑證來佈建。此憑證是由受信任的憑證授權單位核發給 AWS IoT，用來驗證我們的身分和網域擁有權。

AWS IoT Core 在裝置連線時所做的第一件事之一，就是向裝置傳送伺服器憑證。裝置可以驗證其預期連線至 `iot.amazonaws.com`，以及該連線端的伺服器是否擁有來自該網域之受信任授權單位的憑證。

TLS 憑證採用 X.509 格式，並包含各種資訊，例如組織的名稱、位置、網域名稱和有效期間。有效期間以稱為 `notBefore` 和 `notAfter` 的一對時間值來指定。AWS IoT Core 等服務會對其伺服器憑證使用有限的有效期 (例如一年)，並在舊的憑證過期之前開始提供新的憑證。

使用每個裝置單一身分

使用每個用戶端單一身分。裝置通常使用 X.509 用戶端憑證。Web 與行動應用程式使用 Amazon Cognito Identity。這可讓您將精細的許可套用至您的裝置。

一個範例情況是由行動電話裝置組成的應用程式，該裝置從兩個不同的智慧型家用物件接收狀態更新 - 燈泡和電熱器。燈泡會傳送其電池電量的狀態，而電熱器會傳送報告溫度的訊息。

由於 AWS IoT 會個別驗證裝置並個別處理每個連線，因此您可以用授權政策的形式來套用精細的存取控制。您可以定義電熱器的政策，以允許電熱器發佈至主題空間。您可以為燈泡定義個別的政策，以允許燈泡發佈至不同的主題空間。最後，您可以定義行動應用程式的政策，只允許它連接和訂閱電熱器和燈泡的主題，以接收來自這些裝置的訊息。

盡可能套用最低權限的政策，並盡可能降低每個裝置的許可範圍。所有裝置或使用者在 AWS IoT 中都應該有 AWS IoT 政策，只允許它與已知 clientId 連線、發佈和訂閱已識別和固定的主題集。

使用及時佈建

手動建立和佈建每個裝置可能非常耗時。AWS IoT 提供了一種方法，可定義範本以便在裝置第一次連線至 AWS IoT 時佈建裝置。如需詳細資訊，請參閱即時佈建 (p. 475)。

連接裝置

裝置會使用您帳戶專屬的完整網域名稱 (FQDN) 連線至 AWS IoT。AWS IoT 的所有連線必須使用 TLS 1.2 版加以保護。如需詳細資訊，請參閱 AWS IoT中的傳輸安全性。

AWS IoT 提供三種類型的端點，而這些端點是由下列三種它們公開的服務類型所區分：

- `Data` – 用來將資料傳送至 AWS IoT 的訊息中介裝置、裝置影子和規則引擎元件，以及從這些元件中接收資料。
- `Credential_Provider` – 用來將裝置的內建 X.509 憑證交換為臨時登入資料，以直接與其他 AWS 服務連線。如需連線至其他 AWS 服務的詳細資訊，請參閱授權直接呼叫 AWS 服務。
- `Jobs` – 用來讓裝置可以使用任務裝置 MQTT 和 HTTPS API 與 AWS IoT 任務服務互動。

每個 AWS IoT 客戶都有每個服務類型的預設端點。您可以使用 DescribeEndpoint API 來取得您的預設端點。下列清單包含有效的端點類型：

- `iot:Data-ATS` – `Data` 服務類型的端點。
- `iot:CredentialProvider` – `Credential_Provider` 服務類型的端點。
- `iot:Jobs` – `Jobs` 服務類型的端點。

> **Important**
>
> 每個客戶都有另一個端點類型：`iot:Data`。使用此端點可擷取使用 VeriSign 憑證以提供回溯相容性的舊資料端點。強烈建議客戶使用較新的 `iot:Data-ATS` 端點類型，以避免與普遍不信任 Symantec 憑證授權單位相關的問題。如需詳細資訊，請參閱伺服器身分驗證。

使用 `endpointType` 參數指定要從 `DescribeEndpoint` API 取得的端點類型。下列 AWS CLI 範例會取得您的預設 `iot:Data-ATS` 端點。

```
aws iot describe-endpoint --endpoint-type iot:Data-ATS
```

這個命令會傳回下列格式的端點：*`account-specific-prefix`*`-ats.iot.`*`region`*`.amazonaws.com`

您也可以藉由使用可設定端點功能 (目前處於公開測試階段)，來使用自己的 FQDN (例如 *`example.com`*) 和關聯的伺服器憑證，將裝置連線至 AWS IoT。下節說明如何針對您的自訂網域和 AWS 受管網域使用可設定的端點。

主題

- 可設定端點 (Beta 版) (p. 237)

可設定端點 (Beta 版)

> 此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

您可以建立多個端點，將裝置連線至 AWS IoT。您也可以使用網域組態來設定特定詳細資訊 (例如網域名稱)。您可以使用網域組態來簡化如下任務。

- 將裝置遷移至 AWS IoT

- 維護個別裝置類型的不同網域組態，以支援異質裝置機群
- 維護品牌身分 (例如透過網域名稱)，同時將應用程式基礎設施遷移至 AWS IoT

在 Beta 版中，您也可以設定完整網域名稱 (FQSN) 和關聯的伺服器憑證。您也可以將自訂授權方或增強的自訂授權方建立關聯。如需詳細資訊，請參閱自訂授權方和增強的自訂授權方。

> **Note**
>
> AWS IoT 會運用伺服器名稱指示 (SNI) TLS 延伸來套用網域組態。裝置在連線並傳遞與網域組態中指定之網域名稱相同的伺服器名稱時，必須使用此延伸。若要測試此服務，請使用 GitHub 中每個 AWS IoT 裝置開發套件的 v2 版本。

主題

- 建立和設定 AWS 受管網域 (p. 238)
- 建立和設定自訂網域 (p. 238)
- 管理網域組態 (p. 240)

建立和設定 AWS 受管網域

> 此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

您可以使用 CreateDomainConfiguration API，在 AWS 受管網域上建立可設定端點。AWS 受管網域的網域組態包含下列項目：

- `domainConfigurationName` – 識別網域組態的使用者定義名稱。

  > **Note**
  >
  > 以 `IoT:` 開頭的網域組態名稱會保留給預設端點，且無法使用。此外，此值對您的區域也必須是唯一的。

- `defaultAuthorizerName` – 要在端點上使用的自訂授權方名稱。
- `allowAuthorizerOveride` – 布林值，用來指定裝置是否可以在請求的 HTTP 標頭中指定不同的授權方來覆寫預設授權方。如果指定了 `defaultAuthorizerName` 的值，則需要此值。
- `serviceType` – 可能值為 `DATA`、`CREDENTIAL_PROVIDER` 及 `JOB`。如果您指定 `Data`，則 AWS IoT 會傳回端點類型為 `iot:Data-Beta` 的端點。這是可設定端點 Beta 版的特殊端點類型。此端點提供 ATS 簽署的伺服器憑證。您無法建立可設定的 `iot:Data` (VeriSign) 端點。

下列 AWS CLI 命令會建立 `Data` 端點的網域組態。

```
aws iot create-domain-configuration -domain-configuration-name "myDomainConfigurationName"
-service-type "DATA"
```

建立和設定自訂網域

> 此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

網域組態可讓您指定自訂完整網域名稱 (FQDN) 來連線至 AWS IoT。自訂網域可讓您管理自己的伺服器憑證，以便您可以管理詳細資料，例如用來簽署憑證的根憑證授權單位 (CA)、簽章演算法、憑證鏈深度，以及憑證的生命週期。

使用自訂網域來設定網域組態的工作流程包含下列三個階段。

1. 在 AWS Certificate Manager 中註冊伺服器憑證 (p. 239)
2. 建立網域組態 (p. 239)
3. 建立 DNS 記錄 (p. 240)

在 AWS Certificate Manager 中註冊伺服器憑證

使用自訂網域建立網域組態之前，您必須在 AWS Certificate Manager (ACM) 中註冊伺服器憑證鏈。您可以使用三種類型的伺服器憑證。

- ACM 產生的公有憑證 (p. 239)
- 由公有 CA 簽署的外部憑證 (p. 239)
- 由私有 CA 簽署的外部憑證 (p. 239)

> **Note**
>
> AWS IoT 會考慮一個要由公有 CA 簽署的憑證，如果它包含在 Mozilla 信任的 ca-bundle 中。

針對多個網域使用一個憑證

如果您打算使用一個憑證來涵蓋多個子網域，請在通用名稱 (CN) 或主體替代名稱 (SAN) 欄位中使用萬用字元網域。例如，使用 **`*.iot.example.com`** 來涵蓋 dev.iot.example.com、qa.iot.example.com 和 prod.iot.example.com。每個 FQDN 都需要自己的網域組態，但多個網域組態可以使用相同的萬用字元值。CN 或 SAN 必須涵蓋您要用作自訂網域的 FQDN。此涵蓋範圍可以是完全相符或萬用字元相符。

下列各節說明如何取得每種類型的憑證。每個憑證資源都需要向 ACM 註冊的 ARN，您在建立網域組態時會使用此 ARN。

ACM 產生的公有憑證

您可以使用 RequestCertificate API，為您的自訂網域產生公有憑證。當您以這種方式產生憑證時，ACM 會驗證您對自訂網域的擁有權。如需詳細資訊，請參閱《AWS Certificate Manager User Guide》中的請求公有憑證。

由公有 CA 簽署的外部憑證

如果您已經擁有由公有 CA (包含在 Mozilla 信任的 ca-bundle 中的 CA) 簽署的伺服器憑證，則可以使用 ImportCertificate API，將憑證鏈直接匯入至 ACM。若要進一步了解此任務以及必要條件和憑證格式需求，請參閱匯入憑證。

由私有 CA 簽署的外部憑證

如果您已經擁有由私有 CA 簽署或自我簽署的伺服器憑證，則可以使用憑證來建立您的網域組態，但也必須在 ACM 中建立額外的公有憑證，以驗證網域的擁有權。若要這樣做，請在 ACM 使用 ImportCertificate API 來註冊您的伺服器憑證鏈。若要進一步了解此任務以及必要條件和憑證格式需求，請參閱匯入憑證。

將您的憑證匯入至 ACM 之後，請使用 RequestCertificate API，為您的自訂網域產生公有憑證。當您以這種方式產生憑證時，ACM 會驗證您對自訂網域的擁有權。如需詳細資訊，請參閱請求公有憑證。當您建立網域組態時，請使用此公有憑證做為您的驗證憑證。

建立網域組態

您可以使用 CreateDomainConfiguration API，在自訂網域上建立可設定端點。自訂網域的網域組態包含下列項目：

- `domainConfigurationName` – 識別網域組態的使用者定義名稱。

  Note

  以 `IoT:` 開頭的網域組態名稱會保留給預設端點，且無法使用。此外，此值對您的區域也必須是唯一的。
- `domain` – 您裝置用來連線至 AWS IoT 的 FQDN。

  Note

  AWS IoT 會運用伺服器名稱指示 (SNI) TLS 延伸來套用網域組態。裝置在連線並傳遞與網域組態中指定之網域名稱相同的伺服器名稱時，必須使用此延伸。
- `serverCertificateArns` – 您向 ACM 註冊之伺服器憑證鏈的 ARN。Beta 版僅支援一個伺服器憑證。
- `validationCertificateArn` – 您在 ACM 中產生之公有憑證的 ARN，此憑證用驗證自訂網域的擁有權。如果您使用公開簽署或 ACM 產生的伺服器憑證，則不需要此引數。
- `defaultAuthorizerName` – 要在端點上使用的自訂授權方名稱。
- `allowAuthorizerOveride` – 布林值，用來指定裝置是否可以在請求的 HTTP 標頭中指定不同的授權方來覆寫預設授權方。如果指定了 `defaultAuthorizerName` 的值，則需要此值。
- `serviceType` – 可能值為 `DATA`、`CREDENTIAL_PROVIDER` 及 `JOB`。如果您指定 `Data`，則 AWS IoT 會傳回端點類型為 `iot:Data-Beta` 的端點。這是可設定端點 Beta 版的特殊端點類型。您無法建立可設定的 `iot:Data` (VeriSign) 端點。

下列 AWS CLI 命令會建立 iot.example.com 的網域組態。

```
aws iot create-domain-configuration --domainConfigurationName myDomainConfigurationName --
serviceType Data
 --domainName iot.example.com --serverCertificateArns serverCertARN --
validationCertificateArn validationCertArn
```

Note

在建立您的網域組態之後，最多可能需要 15 分鐘，AWS IoT 才能提供您的自訂伺服器憑證。

建立 DNS 記錄

在註冊您的伺服器憑證鏈並建立您的網域組態之後，請建立 DNS 記錄別名，讓您的自訂網域可以指向 AWS IoT 網域。此別名必須指向類型為 `iot:Data-Beta` 的 AWS IoT 端點。這是可設定端點 Beta 版的特殊端點類型。您可以使用 DescribeEndpoint API 來取得 Beta 端點。

下列 AWS CLI 命令顯示如何取得您的 Beta 端點。

```
aws iot describe-endpoint --endpoint-type iot:Data-Beta
```

取得您的 `iot:Data-Beta` 端點之後，請建立從您的自訂網域到此 AWS IoT 端點的 `ALIAS` 記錄。如果您在同一個帳戶中建立多個自訂網域，請對此相同的 `iot:Data-Beta` 端點以別名稱呼這些網域。

管理網域組態

此功能目前為公開測試版，並且僅在美國東部 (維吉尼亞北部) 區域提供。

您可以使用下列 API 來管理現有組態的生命週期。

- ListDomainConfigurations

- DescribeDomainConfiguration
- UpdateDomainConfiguration
- DeleteDomainConfiguration

檢視網域組態

使用 ListDomainConfigurations API 來傳回您帳戶中所有網域組態的分頁清單。您可以使用 DescribeDomainConfiguration API 來查看特定網域組態的詳細資訊。此 API 採用單一 `domainConfigurationName` 參數，並傳回所指定組態的詳細資訊。

更新網域組態

若要更新網域組態的狀態或自訂授權方，請使用更 UpdateDomainConfiguration API。您可以將狀態設為 `ENABLED` 或 `DISABLED`。如果您停用網域組態，連線至該網域的裝置會收到驗證錯誤。

> **Note**
>
> 目前您無法在網域組態中更新伺服器憑證。若要變更網域組態的憑證，您必須刪除並重新建立它。

刪除網域組態

刪除網域設定之前，請使用 UpdateDomainConfiguration API，將狀態設為 `DISABLED`。這可協助您避免意外刪除端點。停用網域組態之後，請使用 DeleteDomainConfiguration API 將其刪除。

刪除網域組態之後，AWS IoT 不再提供與該自訂網域相關聯的伺服器憑證。

AWS IoT 的訊息中介裝置

AWS IoT 訊息中介裝置可讓用戶端與 AWS IoT 通訊，以及讓 AWS IoT 與用戶端通訊。用戶端透過在主題上發佈訊息來傳送資料。用戶端透過訂閱主題接收訊息。當訊息中介裝置收到訊息時，會將該訊息轉寄給訂閱該主題的所有用戶端。

訊息中介裝置會維護一份包含所有用戶端工作階段以及每個工作階段之訂閱的清單。訊息於主題發佈時，中介裝置會檢查對應主題之訂閱的工作階段。接著，中介裝置會將發佈訊息轉發給目前連接至用戶端的所有工作階段。

如果您的使用案例不需要 AWS IoT，請參閱 AWS 傳訊取得有關其他更符合您需求之 AWS 傳訊服務的資訊。

通訊協定

訊息中介裝置支援使用 MQTT 通訊協定發佈與訂閱，也支援使用 HTTPS 發佈。兩種通訊協定均透過 IP 版本 4 及 IP 版本 6 支援。訊息中介裝置也支援透過 WebSocket 通訊協定的 MQTT。連接至訊息中介裝置的方式，取決於您使用的通訊協定。不同的通訊協定需要不同的身分驗證機制。

通訊協定、連接埠映射和身分驗證

下表為 AWS IoT 支援的所有通訊協定、身分驗證方法以及每個通訊協定使用的連接埠。

通訊協定、身分驗證和連接埠對應

通訊協定	身分驗證	連接埠	ALPN ProtocolName
MQTT	X.509 用戶端憑證	8883, 443†	`x-amzn-mqtt-ca`
HTTPS	X.509 用戶端憑證	8443, 443†	`x-amzn-http-ca`
HTTPS	SigV4	443	無
經 WebSocket 的 MQTT	SigV4	443	無

†在連接埠 443 上使用 X.509 用戶端憑證身分驗證連接的用戶端，必須實作應用程式層通訊協定交涉 (ALPN) TLS 延伸模組，並將用戶端傳送的 ALPN ProtocolNameList 中列出的 ALPN ProtocolName 寫在 `ClientHello` 訊息內。

MQTT

MQTT 是一種廣為採用的輕量型傳訊通訊協定，專用於受限裝置。如需更多詳細資訊，請參閱 MQTT。AWS IoT 訊息中介裝置支援服務品質 (QoS) 層級 0 和 1。

雖然 AWS IoT 訊息中介裝置實作以 MQTT 版本 3.1.1 為基礎，但卻有偏離規範的部分，內容如下：

- 在 AWS IoT 中，訂閱 QoS 0 的主題表示訊息會發送零次或多次。一則訊息可能會傳送超過一次。傳送超過一次的訊息在發送時可能會使用不同的封包 ID。在這些情況下，DUP 旗標就不會設置。
- AWS IoT 不支援以 QoS 2 發佈及訂閱。請求 QoS 2 時，AWS IoT 訊息中介裝置不會傳送 PUBACK 或 SUBACK。
- 在回應連線請求時，訊息中介裝置會傳送 CONNACK 訊息。此訊息包含了一個旗標，指出連線是否會恢復先前的工作階段。
- 當用戶端訂閱主題，訊息中介裝置傳送 SUBACK 以及用戶端開始接收符合的新訊息這段期間可能會發生延遲。

- MQTT 規範提供了一項規定，讓發佈者可以請求中介裝置保留發送至主題的最新訊息，並將此訊息傳送給未來所有的主題訂閱者。AWS IoT 不支援保留的訊息。如果請求保留訊息，連線就會中斷。
- 訊息中介裝置使用用戶端 ID 來識別每一個用戶端。用戶端 ID 以做為 MQTT 承載的一部分從用戶端傳遞至訊息中介裝置。具有相同用戶端 ID 的兩個用戶端，不得同時連線至訊息中介裝置。當用戶端使用另一個用戶端正在使用的用戶端 ID 連線至訊息中介裝置，系統會接受新的用戶端連線，而先前連線的用戶端會中斷連線。
- 在極少數情況下，訊息中介裝置可能會以不同的封包 ID 重新傳送相同的邏輯 PUBLISH 訊息。
- 訊息中介裝置不保證接收訊息及 ACK 的順序。

您可以使用其中一個 AWS IoT 軟體開發套件 (p. 678) 透過 MQTT 連接至 AWS IoT。如需如何使用 MQTT 連接至 AWS IoT 的範例，請參閱各軟體開發套件中的 `basicPubSub` 範例。

如需 MQTT 訊息之驗證和連接埠對應的詳細資訊，請參閱通訊協定、連接埠映射和身分驗證 (p. 242)。

主題

訊息中介裝置使用主題將訊息從發佈用戶端路由至訂閱用戶端。主題是以 UTF-8 編碼的階層字串。正斜線 (/) 用於在主題階層中分隔階層。例如：

```
Sensor/temperature/room1
```

本主題會提及房間 1 中的溫度感應器。

> **Note**
>
> 請勿在您的主題中使用個人識別資訊。

以下表格列出在您訂閱時可用於主題篩選條件的萬用字元。

主題萬用字元

萬用字元	描述
#	必須是您所訂閱主題的最後一個字元。做為萬用字元，比對目前的樹及所有子樹。例如，訂閱 `Sensor/#` 會收到發佈至 `Sensor/`、`Sensor/temp`、`Sensor/temp/room1` 的訊息，但不會收到發佈至 `Sensor` 的訊息。
+	完全符合主題階層中的一個項目。例如，訂閱 `Sensor/+/room1` 會收到發佈至 `Sensor/temp/room1`、`Sensor/moisture/room1` 等的訊息。

每個 AWS 帳戶和區域對的主題命名空間都是隔離的。例如，AWS 帳戶的 `Sensor/temp/room1`主題與另一個 AWS 帳戶的 `Sensor/temp/room1` 主題並不相關。區域也是如此。在 `Sensor/temp/room1` 中同一 AWS 帳戶中的 us-east-1 主題，與 us-east-2 中的相同主題無關。AWS IoT 不支援跨 AWS 帳戶和區域傳送和接收訊息。

預留主題

除了這裡列出的主題之外，任何以 $ 開頭的主題均視為預留主題，而且不支援發佈和訂閱。只要嘗試發佈或訂閱任何以 $ 開頭的主題，都會導致終止連線。

事件主題

主題	允許操作	描述
$aws/events/presence/connected/*clientId*	訂閱	當具有指定用戶端 ID 的 MQTT 用戶端連線至 AWS IoT，AWS

主題	允許操作	描述
		IoT 就會發佈至此主題。如需詳細資訊，請參閱 連線/中斷連線事件 (p. 667)。
$aws/events/presence/disconnected/*clientId*	訂閱	當具有指定用戶端 ID 的 MQTT 用戶端與 AWS IoT 中斷連線，AWS IoT 就會發佈至此主題。如需詳細資訊，請參閱 連線/中斷連線事件 (p. 667)。
$aws/events/subscriptions/subscribed/*clientId*	訂閱	當具有指定用戶端 ID 的 MQTT 用戶端訂閱 MQTT 主題，AWS IoT 就會發佈至此主題。如需更多詳細資訊，請參閱 Subscribe/Unsubscribe Events (p. 668)。
$aws/events/subscriptions/unsubscribed/*clientId*	訂閱	當具有指定用戶端 ID 的 MQTT 用戶端取消訂閱 MQTT 主題，AWS IoT 就會發佈至此主題。如需更多詳細資訊，請參閱 Subscribe/Unsubscribe Events (p. 668)。

規則主題

主題	允許操作	描述
$aws/rules/*ruleName*	發佈	裝置或應用程式會發佈至此主題，以直接觸發規則。如需更多詳細資訊，請參閱 基本擷取 (p. 331)。

物件影子主題

主題	允許操作	描述
$aws/things/*<thingName>*/shadow/delete	發佈/訂閱	裝置或應用程式會發佈至此主題，以刪除影子。如需詳細資訊，請參閱 /delete。
$aws/things/*<thingName>*/shadow/delete/accepted	訂閱	當影子刪除時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /delete/accepted。
$aws/things/*<thingName>*/shadow/delete/rejected	訂閱	拒絕刪除影子的要求時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /delete/rejected。
$aws/things/*<thingName>*/shadow/get	發佈/訂閱	應用程式或物件會發佈空白訊息至此主題，以取得影子。如需詳細資訊，請參閱陰影 MQTT 主題。

主題	允許操作	描述
$aws/things/<thingName>/shadow/get/accepted	訂閱	當要求影子成功時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /get/accepted。
$aws/things/<thingName>/shadow/get/rejected	訂閱	當要求影子被拒絕時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /get/rejected。
$aws/things/<thingName>/shadow/update	發佈/訂閱	物件或應用程式會發佈至此主題，以更新影子。如需詳細資訊，請參閱 /update。
$aws/things/<thingName>/shadow/update/accepted	訂閱	當更新影子成功時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /update/accepted。
$aws/things/<thingName>/shadow/update/rejected	訂閱	當更新影子被拒絕時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /update/rejected。
$aws/things/<thingName>/shadow/update/delta	訂閱	當偵測到回報的區段與所需的區段之間發生差異時，Device Shadow 服務會傳送訊息至此主題。如需詳細資訊，請參閱 /update/delta。
$aws/things/<thingName>/shadow/update/documents	訂閱	每當影子成功執行更新，AWS IoT 就會發佈狀態文件至此主題：如需詳細資訊，請參閱 /update/documents。

任務主題

主題	允許操作	描述
$aws/things/<thingName>/jobs/get	發佈	裝置發佈訊息到這個主題來提出 `GetPendingJobExecutions` 請求。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/get/accepted	訂閱	裝置訂閱此主題，以接收來自 `GetPendingJobExecutions` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/get/accepted	訂閱	裝置訂閱此主題，以接收送往 `GetPendingJobExecutions` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。

主題	允許操作	描述
$aws/things/<thingName>/jobs/start-next	發佈	裝置發佈訊息到這個主題來提出 `StartNextPendingJobExecution` 請求。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/start-next/accepted	訂閱	裝置訂閱此主題，以接收送往 `StartNextPendingJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/start-next/rejected	訂閱	裝置訂閱此主題，以接收送往 `StartNextPendingJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/jobId/get	發佈	裝置發佈訊息到這個主題來提出 `DescribeJobExecution` 請求。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/<jobId>/get/accepted	訂閱	裝置訂閱此主題，以接收送往 `DescribeJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/<jobId>/get/rejected	訂閱	裝置訂閱此主題，以接收送往 `DescribeJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/<jobId>/update	發佈	裝置發佈訊息到這個主題來提出 `UpdateJobExecution` 請求。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/<jobId>/update/accepted	訂閱	裝置訂閱此主題，以接收送往 `UpdateJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/<jobId>/update/rejected	訂閱	裝置訂閱此主題，以接收送往 `UpdateJobExecution` 請求的成功回應。如需詳細資訊，請參閱使用 AWS IoT 任務 API。

主題	允許操作	描述
$aws/things/<thingName>/jobs/notify	訂閱	裝置訂閱此主題，以在物件的待處理執行清單新增或移除工作執行時接收通知。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/things/<thingName>/jobs/notify-next	訂閱	裝置訂閱此主題，以在物件的下一個待處理工作執行變更時接收通知。如需詳細資訊，請參閱使用 AWS IoT 任務 API。
$aws/events/job/<jobId>/completed	訂閱	工作完成時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/job/<jobId>/canceled	訂閱	工作取消時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/job/<jobId>/deleted	訂閱	工作刪除時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/job/<jobId>/cancellation_in_progress	訂閱	開始取消工作時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/job/<jobId>/deletion_in_progress	訂閱	開始刪除工作時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/succeeded	訂閱	工作執行成功時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/failed	訂閱	工作執行失敗時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/rejected	訂閱	工作執行遭拒時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/canceled	訂閱	工作執行取消時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/timed_out	訂閱	工作執行逾時時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。
$aws/events/jobExecution/<jobId>/removed	訂閱	工作執行移除時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。

主題	允許操作	描述
$aws/events/jobExecution/<jobId>/deleted	訂閱	工作執行刪除時，工作服務會發佈這個主題的事件。如需詳細資訊，請參閱工作事件。

機群佈建主題

主題	允許操作	描述
$aws/events/presence/connected/<clientId>	訂閱	當具有指定用戶端 ID 的 MQTT 用戶端連線至 AWS IoT，AWS IoT 就會發佈至此主題。如需詳細資訊，請參閱「??? (p. 667)」
$aws/certificates/create/cbor	發布	您可以發佈至此主題，以呼叫 `CreateKeysAndCertificate` MQTT API。
$aws/certificates/create/json	發布	您可以發佈至此主題，以呼叫 `CreateKeysAndCertificate` MQTT API。
$aws/provisioning- templates/{{templateName}}/provision/cbor	發布	您可以發佈至此主題，以呼叫 `RegisterThing` MQTT API。
$aws/provisioning- templates/{{templateName}}/provision/json	發布	您可以發佈至此主題，以呼叫 `RegisterThing` MQTT API。
$aws/certificates/create/cbor/accepted	訂閱	在成功呼叫 `CreateKeysAndCertificate` MQTT API 時，AWS IoT 會發佈至此主題。
$aws/certificates/create/cbor/rejected	訂閱	在呼叫 `CreateKeysAndCertificate` MQTT API 失敗時，AWS IoT 會發佈至此主題。
$aws/certificates/create/json/accepted	訂閱	在成功呼叫 `CreateKeysAndCertificate` MQTT API 時，AWS IoT 會發佈至此主題。
$aws/certificates/create/json/rejected	訂閱	在呼叫 `CreateKeysAndCertificate` MQTT API 失敗時，AWS IoT 會發佈至此主題。
$aws/provisioning- templates/{{templateName}}/provision/cbor/accepted	訂閱	在成功呼叫 `RegisterThing` MQTT API 時，AWS IoT 會發佈至此主題。

主題	允許操作	描述
$aws/provisioning- templates/{{templateName}}/provision/cbor/rejected	訂閱	在呼叫 `RegisterThing` MQTT API 失敗時，AWS IoT 會發佈至此主題。
$aws/provisioning- templates/{{templateName}}/provision/json/accepted	訂閱	在成功呼叫 `RegisterThing` MQTT API 時，AWS IoT 會發佈至此主題。
$aws/provisioning- templates/{{templateName}}/provision/json/rejected	訂閱	在呼叫 `RegisterThing` MQTT API 失敗時，AWS IoT 會發佈至此主題。

MQTT 持久性工作階段

持久性工作階段代表 MQTT 訊息中介裝置的持續連線。當用戶端使用持久性工作階段來連接到 AWS IoT 訊息中介裝置時，訊息中介裝置會儲存用戶端在連線期間所進行的所有訂閱。當用戶端中斷連線時，訊息中介裝置會存放未認可的 QoS 1 訊息，以及發佈到用戶端所訂閱之主題的新 QoS 1 訊息。當用戶端重新連線至持久性工作階段時，所有訂閱都會恢復，而且所有存放的訊息都會傳送至用戶端，其傳送速率為每秒最多10則訊息。

您可以傳送將 `cleanSession` 旗標設為 0 的 CONNECT 訊息，來建立 MQTT 持久性工作階段。如果傳送 CONNECT 訊息的用戶端沒有工作階段，則會建立新的持久性工作階段。如果用戶端已有工作階段，則會恢復它。

裝置需要查看 CONNACK (連線認可) 訊息中的 `sessionPresent` 屬性，來判定是否需要持久性工作階段。若將 `sessionPresent` 設為 1，則持久性工作階段會呈現，而且存放的訊息會遞送到用戶端。若將 `sessionPresent` 設為 0，則不會呈現任何持久性工作階段，而且用戶端必須重新訂閱其主題篩選條件。

持久性工作階段的預設到期期間為 1 小時。當訊息中介裝置偵測到用戶端中斷連線 (MQTT 中斷連線或逾時) 時，到期期間即會開始。您可以透過標準限制增加程序來增加持久性工作階段到期期間。如果用戶端未在到期期間內恢復其工作階段，則工作階段會終止，並捨棄任何相關聯的存放訊息。到期期間是預估值，工作階段最多可比設定的持續時間多保留 30 分鐘，但不會少於設定的持續時間。如需詳細資訊，請參閱 AWS 服務限制。任何針對持久性工作階段存放的訊息將按照標準簡訊速率計費。如需詳細資訊，請參閱 AWS IoT 定價。

經 WebSocket 通訊協定的 MQTT

AWS IoT 支援經 WebSocket 通訊協定的 MQTT，可讓透過瀏覽器和遠端的應用程式跟使用 AWS 登入資料的 AWS IoT 連線裝置互相收發資料。AWS 登入資料由 AWS 簽章版本 4 指定。WebSocket 支援 TCP 連接埠 443，可讓訊息通過大多數的防火牆和 web 代理伺服器。

WebSocket 連線在用戶端上是透過傳送 HTTP GET 請求來啟動。您使用的 URL 格式如下：

```
wss://<endpoint>.iot.<region>.amazonaws.com/mqtt
```

wss

指定 WebSocket 通訊協定。

endpoint

專屬於您 AWS 帳戶的 AWS IoT 端點。您可以使用 AWS IoT CLI describe-endpoint 命令找出此端點。

region

您 AWS 帳戶的 AWS 區域。

mqtt

指定您經 WebSocket 通訊協定傳送 MQTT 訊息。

伺服器回應時，用戶端會傳送更新請求，向伺服器表示將使用 WebSocket 通訊協定來通訊。伺服器確認更新請求後，所有通訊都會使用 WebSocket 通訊協定。您使用的 WebSocket 實作會成為傳輸通訊協定。您透過 WebSocket 通訊協定傳送的資料為 MQTT 訊息。

在 Web 應用程式中使用 WebSocket 通訊協定

大多數 web 瀏覽器提供的 WebSocket 實作並不允許修改 HTTP 標頭，因此您必須在查詢字串加上簽章版本 4 的資訊。如需更多詳細資訊，請參閱 Adding Signing Information to the Query String (將簽署資訊加入查詢字串)。

以下 JavaScript 定義了一些用於產生簽章版本 4 請求的公用程式函式。

```
 /**
      * utilities to do sigv4
      * @class SigV4Utils
      */
     function SigV4Utils() {}

   SigV4Utils.getSignatureKey = function (key, date, region, service) {
       var kDate = AWS.util.crypto.hmac('AWS4' + key, date, 'buffer');
       var kRegion = AWS.util.crypto.hmac(kDate, region, 'buffer');
       var kService = AWS.util.crypto.hmac(kRegion, service, 'buffer');
       var kCredentials = AWS.util.crypto.hmac(kService, 'aws4_request', 'buffer');
       return kCredentials;
   };

   SigV4Utils.getSignedUrl = function(host, region, credentials) {
       var datetime = AWS.util.date.iso8601(new Date()).replace(/[:\-]|\.\d{3}/g, '');
       var date = datetime.substr(0, 8);

       var method = 'GET';
       var protocol = 'wss';
       var uri = '/mqtt';
       var service = 'iotdevicegateway';
       var algorithm = 'AWS4-HMAC-SHA256';

       var credentialScope = date + '/' + region + '/' + service + '/' + 'aws4_request';
       var canonicalQuerystring = 'X-Amz-Algorithm=' + algorithm;
       canonicalQuerystring += '&X-Amz-Credential=' +
encodeURIComponent(credentials.accessKeyId + '/' + credentialScope);
       canonicalQuerystring += '&X-Amz-Date=' + datetime;
       canonicalQuerystring += '&X-Amz-SignedHeaders=host';

       var canonicalHeaders = 'host:' + host + '\n';
       var payloadHash = AWS.util.crypto.sha256('', 'hex')
       var canonicalRequest = method + '\n' + uri + '\n' + canonicalQuerystring + '\n' +
canonicalHeaders + '\nhost\n' + payloadHash;

       var stringToSign = algorithm + '\n' + datetime + '\n' + credentialScope + '\n' +
AWS.util.crypto.sha256(canonicalRequest, 'hex');
       var signingKey = SigV4Utils.getSignatureKey(credentials.secretAccessKey, date,
region, service);
       var signature = AWS.util.crypto.hmac(signingKey, stringToSign, 'hex');

       canonicalQuerystring += '&X-Amz-Signature=' + signature;
       if (credentials.sessionToken) {
           canonicalQuerystring += '&X-Amz-Security-Token=' +
encodeURIComponent(credentials.sessionToken);
```

```
    }

    var requestUrl = protocol + '://' + host + uri + '?' + canonicalQuerystring;
    return requestUrl;
};
```

建立簽章版本 4 請求

1. 建立簽章版本 4 的正式請求。

   以下 JavaScript 程式碼可建立正式請求：

```
var datetime = AWS.util.date.iso8601(new Date()).replace(/[:\-]|\.\d{3}/g, '');
    var date = datetime.substr(0, 8);

    var method = 'GET';
    var protocol = 'wss';
    var uri = '/mqtt';
    var service = 'iotdevicegateway';
    var algorithm = 'AWS4-HMAC-SHA256';

    var credentialScope = date + '/' + region + '/' + service + '/' + 'aws4_request';
    var canonicalQuerystring = 'X-Amz-Algorithm=' + algorithm;
    canonicalQuerystring += '&X-Amz-Credential=' +
 encodeURIComponent(credentials.accessKeyId + '/' + credentialScope);
    canonicalQuerystring += '&X-Amz-Date=' + datetime;
    canonicalQuerystring += '&X-Amz-SignedHeaders=host';

    var canonicalHeaders = 'host:' + host + '\n';
    var payloadHash = AWS.util.crypto.sha256('', 'hex')
    var canonicalRequest = method + '\n' + uri + '\n' + canonicalQuerystring + '\n' +
 canonicalHeaders + '\nhost\n' + payloadHash;
```

2. 建立登入字串、產生簽署金鑰，以及簽署字串。

   使用您在前一個步驟建立的正式 URL，將其結合至登入字串。你需要建立一組字串，其中包含雜湊演算法、日期、登入資料範圍，以及正式請求的 SHA。接著，產生簽署金鑰並簽署字串，如下列 JavaScript 程式碼所示。

```
    var stringToSign = algorithm + '\n' + datetime + '\n' + credentialScope + '\n' +
 AWS.util.crypto.sha256(canonicalRequest, 'hex');
    var signingKey = SigV4Utils.getSignatureKey(credentials.secretAccessKey, date,
 region, service);
    var signature = AWS.util.crypto.hmac(signingKey, stringToSign, 'hex');
```

3. 將簽署資訊加入請求。

   以下 JavaScript 程式碼示範如何將簽署資訊加入查詢字串。

```
    canonicalQuerystring += '&X-Amz-Signature=' + signature;
```

4. 如果您有工作階段登入資料 (來自 STS 伺服器、AssumeRole 或 Amazon Cognito)，請在簽署後將工作階段符記附加至 URL 字串末端。

```
canonicalQuerystring += '&X-Amz-Security-Token=' +
 encodeURIComponent(credentials.sessionToken);
```

5. 將通訊協定、主機與 URI 置於 `canonicalQuerystring` 前面：

```
    var requestUrl = protocol + '://' + host + uri + '?' + canonicalQuerystring;
```

6. 開啟 WebSocket。

   以下 JavaScript 程式碼示範如何建立 Paho MQTT 用戶端並呼叫 CONNECT 至 AWS IoT。`endpoint`引數是您 AWS 帳戶特有的端點。`clientId`是在同時連接至您 AWS 帳戶的所有客戶端中，一種獨特的文字識別符。

```
var client = new Paho.MQTT.Client(requestUrl, clientId);
    var connectOptions = {
        onSuccess: function(){
            // connect succeeded
        },
        useSSL: true,
        timeout: 3,
        mqttVersion: 4,
        onFailure: function() {
            // connect failed
        }
    };
    client.connect(connectOptions);
```

在行動應用程式中使用 WebSocket 通訊協定

在建立 WebSocket 連線時，我們建議使用其中一個 AWS IoT 裝置 SDK 來讓您的裝置連線至 AWS IoT。以下 AWS IoT 裝置 SDK 支援以 WebSocket 為基礎的 MQTT 與 AWS IoT支援支援 連線：

- Node.js
- iOS
- Android

關於使用經 WebSocket 通訊協定之 MQTT 將 web 應用程式連接至 AWS IoT 的參考實作，請參閱 AWS 實驗室 WebSocket 範例。

如果您使用的程式設計或腳本語言目前未受支援，只要初始 WebSocket 升級請求 (HTTP POST) 使用簽章版本 4 來簽署，就可以使用任何現有的 WebSocket 庫。有些 MQTT 用戶端 (例如 Eclipse Paho for JavaScript) 可原生支援 WebSocket 通訊協定。

HTTP

訊息中介裝置支援使用 REST API 搭配 HTTP 1.0 和 1.1 通訊協定進行連線。如需 HTTP 要求之驗證和連接埠對應的詳細資訊，請參閱通訊協定、連接埠映射和身分驗證 (p. 242)。用戶端可以將 POST 訊息傳送至 `<AWS IoT Endpoint>/topics/<url_encoded_topic_name>?qos=1"` 以發佈。

例如，您可以使用 curl 來模擬訊息的傳送。例如：

```
curl --tlsv1.2 --cacert root-CA.crt --cert 4b7828d2e5-certificate.pem.crt --key
 4b7828d2e5-private.pem.key -X POST -d "{ \"message\": \"Hello, world\" }" "https://
a1pn10j0v8htvw.iot.us-east-1.amazonaws.com:8443/topics/my/topic"
```

--tlsv1.2

使用 TLSv1.2 (SSL). curl 必須與 OpenSSL 一起安裝，而且您必須使用第 1.2 版的 TLS。

--cacert <filename>

用以驗證對等的 CA 憑證檔案名稱。

--cert <filename>

用戶端憑證檔案名稱。

--key <filename>

私有金鑰檔案名稱。

-X POST

請求的類型 (此處指 POST)。

-d <data>

您要發佈的 HTTP POST 資料。

"https://..."

即 URL。此處指物件的 REST API 端點。

若要找出實物的端點，請在 AWS IoT 主控台中選擇 Registry (登錄檔) 以展開您的選項。選擇 Things (物件)，接著選擇物件，再選擇 Interact (互動)。) 在端點後方加上連接埠 (:8443)，接著是關鍵字 "topics"，最後在查詢字串 (?qos = 1) 指定服務品質。

AWS IoT 的規則

裝置與 AWS 服務互動的能力來自於規則。規則的分析和動作的執行均是以 MQTT 的主題流為依歸。您可以使用規則來支援任務，例如：

- 擴增或篩選從裝置接收的資料
- 將從裝置接收的資料寫入 Amazon DynamoDB 資料庫。
- 將檔案儲存至 Amazon S3。
- 使用 Amazon SNS 傳送推送通知給所有使用者。
- 發佈資料到 Amazon SQS 佇列。
- 呼叫 Lambda 函式來擷取資料。
- 使用 Amazon Kinesis 處理來自大量裝置的訊息。
- 傳送資料到 Amazon Elasticsearch Service。
- 擷取 CloudWatch 指標。
- 變更 CloudWatch 警示。
- 將來自 MQTT 訊息的資料傳送至 Amazon Machine Learning，以根據 Amazon ML 模型進行預測。
- 傳送訊息到 Salesforce IoT 輸入串流。
- 傳送訊息資料至 AWS IoT Analytics 頻道。
- 開始執行 Step Functions 狀態機器。
- 傳送訊息資料至 AWS IoT Events 輸入。
- 傳送 AWS IoT SiteWise 中的訊息資料和資產屬性。
- 將訊息資料傳送至 Web 應用程式或服務。

您的規則可以使用經由發佈/訂閱 AWS IoT 的訊息中介裝置 (p. 242) 的 MQTT 訊息，或使用 基本擷取 (p. 331) 功能，您就可以安全地將裝置資料傳送至上方列出的 AWS 服務，而不會產生訊息費用。(基本擷取 (p. 331) 功能透過從擷取路徑移除發佈/訂閱訊息中介裝置來將資料流程最佳化，因此更具成本效益並同時保持 AWS IoT 的安全性和資料處理功能。)

在 AWS IoT 能夠執行上述動作之前，必須先授與許可，允許其代表您存取 AWS 資源。執行動作時，需要支付所用的 AWS 服務的標準費用。

授與 AWS IoT 所需的存取權

您可以使用 IAM 角色來控制各項規則可存取的 AWS 資源。建立規則之前，必須先建立一個允許存取所需的 AWS 資源的 IAM 角色與政策。AWS IoT 執行角色時便會擔任該角色。

建立 IAM 角色 (AWS CLI)

1. 將下列信任政策文件 (其授與 AWS IoT 許可擔任該角色) 儲存至名為 `iot-role-trust.json` 的檔案：

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

```
}
```

使用 create-role 命令來建立用於指定 `iot-role-trust.json` 檔案的 IAM 角色：

```
aws iam create-role --role-name my-iot-role --assume-role-policy-document file://iot-role-trust.json
```

此令命的輸出結果如下所示：

```
{
  "Role": {
    "AssumeRolePolicyDocument": "url-encoded-json",
    "RoleId": "AKIAIOSFODNN7EXAMPLE",
    "CreateDate": "2015-09-30T18:43:32.821Z",
    "RoleName": "my-iot-role",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:role/my-iot-role"
  }
}
```

2. 將下列 JSON 儲存至名為 `iot-policy.json` 的檔案。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "dynamodb:*",
    "Resource": "*"
  }]
}
```

此 JSON 為政策文件範本，會授與 AWS IoT 管理員存取 DynamoDB 的權限。

使用 create-policy 命令，當 AWS IoT 在擔任該角色時，授與其存取您的 AWS 資源的權限，以 `iot-policy.json` 檔案來傳遞：

```
aws iam create-policy --policy-name my-iot-policy --policy-document file://my-iot-policy.json
```

如更多有關如何在 AWS IoT 政策中授與 AWS 服務存取權限的資訊，請參閱 建立 AWS IoT 規則 (p. 256)。

create-policy 命令的輸出結果含有該政策的 ARN。您需要將該政策連接到某個角色。

```
{
  "Policy": {
    "PolicyName": "my-iot-policy",
    "CreateDate": "2015-09-30T19:31:18.620Z",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
    "DefaultVersionId": "v1",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:policy/my-iot-policy",
    "UpdateDate": "2015-09-30T19:31:18.620Z"
  }
}
```

3. 請使用 attach-role-policy 命令將政策連接到您的角色：

```
aws iam attach-role-policy --role-name my-iot-role --policy-arn
 "arn:aws:iam::123456789012:policy/my-iot-policy"
```

傳送角色許可

IAM 角色是規則定義的一部分，會授與規則動作所指定之資源的存取許可。當觸發該項規則的動作時，規則引擎便會擔任該角色。必須在同一個 AWS 帳戶中將該角色定義為規則。

當建立或替換某項規則時，實際上就是在將某個角色傳送至規則引擎。執行此項操作的使用者需要擁有 `iam:PassRole`許可。為了確保擁有該許可，請建立一項授與 `iam:PassRole` 許可的政策，並將之連接到您的 IAM 使用者。以下政策顯示了如何允許角色的 `iam:PassRole`許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::123456789012:role/myRole"
      ]
    }
  ]
}
```

在此政策範例中，將 `iam:PassRole`許可授與了角色 `myRole`。該角色是使用角色的 ARN 來指定。必須將此政策連接至您的 IAM 角色或是您的使用者所屬的角色。如需詳細資訊，請參閱處理受管政策的相關文章。

> **Note**
>
> Lambda 函式使用的是資源型政策，即是政策直接連接至 Lambda 函式本身。在您建立呼叫 Lambda 函式的規則時，並不會傳遞角色，因此建立該規則的使用者不需擁有 `iam:PassRole` 許可。如需 Lambda 函式授權的詳細資訊，請參閱使用資源政策授與許可的相關文章。

建立 AWS IoT 規則

您可以設定規則來路由連網物件的資料。規則包含下列項目：

規則名稱

規則名稱。

> **Note**
>
> 我們不建議在規則名稱中使用個人識別資訊。

其他說明

該項規則的文字說明。

> **Note**
>
> 我們不建議在規則描述中使用個人識別資訊。

SQL 陳述式

簡化的 SQL 語法，用於篩選 MQTT 主題所收到的訊息，並將資料推送到他處。如需詳細資訊，請參閱 AWS IoTSQL 參考 (p. 283)。

SQL 版本

評估規則時所用的 SQL 規則引擎版本。雖然此項屬性為選用，但強烈建議您指定 SQL 的版本。如果未設定該屬性，會預設使用 `2015-10-08`。如需更多詳細資訊，請參閱 SQL 版本 (p. 328)。

一個或多個動作

執行規則時 AWS IoT 執行的動作。例如，您可以將資料插入 DynamoDB 表格、將資料寫入 Amazon S3 儲存貯體、發佈至 Amazon SNS 主題、呼叫 Lambda 函式。

錯誤動作

在無法執行規則的動作時，AWS IoT 所執行的動作。

建立規則時，請注意在主題上發佈的資料量。如果您建立的規則內含萬用字元組主題模式，可能會與您的大量訊息相符合，而您可能會需要增加目標動作所用的 AWS 資源的容量。此外，如果您建立了重新發佈規則，而其中包含萬用字元主題模式，最後可能會造成循環規則，導致無限迴圈。

Note

規則的建立和更新是屬於管理員層級的動作。任何擁有許可而能建立或更新規則的使用者，均能夠存取規則所處理的資料。

建立規則 (AWS CLI)

請使用 create-topic-rule 命令來建立規則：

```
aws iot create-topic-rule --rule-name my-rule --topic-rule-payload file://my-rule.json
```

以下為範例承載檔案，含有一項規則，會將所有發送至 `iot/test` 主題的訊息插入指定的 DynamoDB 表格。SQL 陳述式會篩選訊息以及 ARN 授與 AWS IoT 許可，以寫入 DynamoDB 表格。

```
{
  "sql": "SELECT * FROM 'iot/test'",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "dynamoDB": {
          "tableName": "my-dynamodb-table",
          "roleArn": "arn:aws:iam::123456789012:role/my-iot-role",
          "hashKeyField": "topic",
          "hashKeyValue": "${topic(2)}",
          "rangeKeyField": "timestamp",
          "rangeKeyValue": "${timestamp()}"
      }
  }]
}
```

以下為範例承載檔案，含有一項規則，會將所有發送至 `iot/test`主題的訊息插入指定的 S3 儲存貯體。SQL 陳述式會篩選訊息以及 ARN 授與 AWS IoT 許可，以寫入 Amazon S3 儲存貯體。

```
{
  "awsIotSqlVersion": "2016-03-23",
  "sql": "SELECT * FROM 'iot/test'",
  "ruleDisabled": false,
  "actions": [
```

```
    {
      "s3": {
        "roleArn": "arn:aws:iam::123456789012:role/aws_iot_s3",
        "bucketName": "my-bucket",
        "key": "myS3Key"
      }
    }
  ]
}
```

以下為搭配規則的範例酬載檔案，該檔案會將資料推送至 Amazon Elasticsearch Service：

```
{
  "sql":"SELECT *, timestamp() as timestamp FROM 'iot/test'",
  "ruleDisabled":false,
  "awsIotSqlVersion": "2016-03-23",
  "actions":[
    {
      "elasticsearch":{
        "roleArn":"arn:aws:iam::123456789012:role/aws_iot_es",
        "endpoint":"https://my-endpoint",
        "index":"my-index",
        "type":"my-type",
        "id":"${newuuid()}"
      }
    }
  ]
}
```

以下為範例承載檔案，含有會呼叫 Lambda 函式的規則：

```
{
  "sql": "expression",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "lambda": {
          "functionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-lambda-
function"
      }
  }]
}
```

以下為範例承載檔案，含有會將資料發佈至 Amazon SNS 主題的規則：

```
{
  "sql": "expression",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "sns": {
          "targetArn": "arn:aws:sns:us-west-2:123456789012:my-sns-topic",
          "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
      }
  }]
}
```

以下為範例承載檔案，含有的規則會將資料重新發佈至不同的 MQTT 主題：

```
{
  "sql": "expression",
```

```
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "republish": {
          "topic": "my-mqtt-topic",
          "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
      }
  }]
}
```

以下為範例承載檔案，含有會將資料推送至 Amazon Kinesis Data Firehose 串流的規則：

```
{
  "sql": "SELECT * FROM 'my-topic'",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "firehose": {
          "roleArn": ""arn:aws:iam::123456789012:role/my-iot-role",
          "deliveryStreamName": "my-stream-name"
      }
  }]
}
```

以下為範例承載檔案，若 MQTT 承載內的資料被分類為 1，其規則會使用 Amazon Machine Learning`machinelearning_predict` 函式重新發佈至某個主題。

```
{
  "sql": "SELECT * FROM 'iot/test' where machinelearning_predict('my-model',
 'arn:aws:iam::123456789012:role/my-iot-aml-role', *).predictedLabel=1",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
    "republish": {
        "roleArn": "arn:aws:iam::123456789012:role/my-iot-role",
        "topic": "my-mqtt-topic"
    }
  }]
}
```

以下是範例承載檔案，含有的規則會將訊息發佈至 Salesforce IoT 雲端輸入串流：

```
{
  "sql": "expression",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "salesforce": {
          "token": "ABCDEFGHI123456789abcdefghi123456789",
          "url": "https://ingestion-cluster-id.my-env.sfdcnow.com/streams/stream-id/
connection-id/my-event"
      }
  }]
}
```

以下是具有開始執行 Step Functions 狀態機器之規則的範例承載檔案。

```
{
  "sql": "expression",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
```

```
  "actions": [{
      "stepFunctions": {
          "stateMachineName": "myCoolStateMachine",
          "executionNamePrefix": "coolRunning",
          "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
      }
  }]
}
```

檢視您的規則

請使用 list-topic-rules 命令來列出規則：

```
aws iot list-topic-rules
```

請使用 get-topic-rule 命令來取得規則的相關資訊：

```
aws iot get-topic-rule --rule-name my-rule
```

刪除規則

規則結束使用後即可刪除。

刪除規則 (AWS CLI)

請使用 delete-topic-rule 命令來刪除規則：

```
aws iot delete-topic-rule --rule-name my-rule
```

AWS IoT規則動作

AWS IoT 規則動作用於指定在觸發規則時要採取的動作。可以將動作定譯為將資料寫入 DynamoDB 資料庫或 Kinesis 串流，或是呼叫 Lambda 函式等等。支援以下動作：

- `cloudwatchAlarm` 變更 CloudWatch 警示。
- `cloudwatchMetric` 擷取 CloudWatch 指標。
- `dynamoDB` 將資料寫入 DynamoDB 資料庫。
- `dynamoDBv2` 將資料寫入 DynamoDB 資料庫。
- `elasticsearch` 將資料寫入 Amazon Elasticsearch Service 網域。
- `firehose` 將資料寫入 Amazon Kinesis Data Firehose 串流。
- `http` 將資料發佈至 HTTPS 端點。
- `iotAnalytics` 以傳送資料至 AWS IoT 分析管道。
- `iotEvents` 以傳送資料至 AWS IoT Events 輸入。
- `iotSiteWise` 將資料傳送至 AWS IoT SiteWise 中的資產屬性。
- `kinesis` 將資料寫入 Kinesis 串流。
- `lambda` 叫用 Lambda 函式。

- `republish`可在另一個 MQTT 主題上重新發佈訊息。
- `s3` 將資料寫入 Amazon S3 儲存貯體。
- `salesforce`可將訊息寫入至 Salesforce IoT 輸入流。
- `sns`可將資料寫入為推送通知。
- `sqs`可將資料寫入 SQS 佇列。
- `stepFunctions` 開始執行 Step Functions 狀態機器。

> **Note**
>
> 為防止間歇性錯誤發生，AWS IoT 規則引擎可能會多次試圖執行動作。如果所有嘗試都失敗了，則該訊息將被忽略，而錯誤將顯示在您的 CloudWatch 記錄中。您可以為失敗發生之後叫用的每一個規則指定一個錯誤動作。如需詳細資訊，請參閱 錯誤處理 (錯誤動作) (p. 280)。

每一個動作都有詳細說明。

CloudWatch 警示動作

CloudWatch Alarm Action

CloudWatch 警示動作允許您變更 CloudWatch 警示狀態。您可以指定此呼叫狀態變更的原因和值。

More information (1)

在建立含有 AWS IoT 警示動作的 CloudWatch 規則時，請務必指定下列資訊：

roleArn

IAM 角色，允許存取 CloudWatch 警示。

alarmName

CloudWatch 警示名稱。

stateReason

警示變更的原因。

stateValue

警示狀態的值。可接受的值為 `OK`、`ALARM`、`INSUFFICIENT_DATA`。

> **Note**
>
> 請務必確認與該規則相連的角色擁有授與 `cloudwatch:SetAlarmState`許可的政策。

下列 JSON 範例顯示如何定義 CloudWatch 規則內的 AWS IoT 警示動作：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "cloudwatchAlarm": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_cw",
                "alarmName": "IotAlarm",
                "stateReason": "Temperature stabilized.",
                "stateValue": "OK"
```

```
            }
        }]
    }
}
```

如需詳細資訊，請參閱 CloudWatch 警示。

CloudWatch 指標動作

CloudWatch Metric Action

CloudWatch 指標動作可擷取 CloudWatch 指標。您可以指定指標命名空間、名稱、值、單位、時間戳記。

More information (2)

在建立含有 AWS IoT 指標動作的 CloudWatch 規則時，請務必指定下列資訊：

roleArn

IAM 角色，允許存取 CloudWatch 指標。

metricNamespace

CloudWatch 指標命名空間名稱。

metricName

CloudWatch 指標名稱。

metricValue

CloudWatch 指標值。

metricUnit

CloudWatch 支援的指標單位。

metricTimestamp

Unix 時間戳記 (選用)。

Note

請務必確認關聯至規則的角色擁有授予 `cloudwatch:PutMetricData` 權限的政策。

下列 JSON 範例顯示如何定義 CloudWatch 規則內的 AWS IoT 指標動作：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "cloudwatchMetric": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_cw",
                "metricNamespace": "IotNamespace",
                "metricName": "IotMetric",
                "metricValue": "1",
                "metricUnit": "Count",
                "metricTimestamp": "1456821314"
            }
```

```
        }]
    }
}
```

如需詳細資訊，請參閱 CloudWatch 指標。

DynamoDB 動作

DynamoDB Action

`dynamoDB` 動作可將全部或部分的 MQTT 訊息寫入至 DynamoDB 表格。

More information (3)

建立 DynamoDB 規則時，請務必指定以下資訊：

hashKeyType

雜湊索引鍵 (也稱為分割區索引鍵) 的資料類型。有效值為：`"STRING"`或 `"NUMBER"`。

hashKeyField

雜湊索引鍵 (也稱為分割區索引鍵) 的名稱。

hashKeyValue

雜湊索引鍵的值。

rangeKeyType

選用。範圍索引鍵 (也稱為排序索引鍵) 的資料類型。有效值為：`"STRING"`或 `"NUMBER"`。

rangeKeyField

選用。範圍索引鍵 (也稱為排序索引鍵) 的名稱。

rangeKeyValue

選用。範圍索引鍵的值。

operation

選用。欲執行的操作類型。其遵循的是替代範本，因此能為 `${operation}`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。

payloadField

選用。承載寫入的欄位名稱。如果省略該值，承載會寫入至 `payload`欄位。

表格

DynamoDB 表格的名稱。

roleARN

IAM 角色，允許存取 DynamoDB 表格。該角色至少必須允許 `dynamoDB:PutItem`IAM 動作。

寫入 DynamoDB 表格的資料產自於規則的 SQL 陳述式。`hashKeyValue`和 `rangeKeyValue` 欄位通常是由表達式 (如「${topic()}」或「${timestamp()}」) 組成。

Note

非 JSON 資料會以二進位資料的形式寫入 DynamoDB。DynamoDB 主控台會以 Base64 編碼文字來顯示資料。

請務必確認關聯至規則的角色擁有授予 `dynamodb:PutItem` 權限的政策。

下列 JSON 範例顯示了如何定義 `dynamoDB` 規則內的 AWS IoT 動作：

```
{
    "topicRulePayload": {
        "ruleDisabled": false,
        "sql": "SELECT * AS message FROM 'some/topic'",
        "description": "A test Dynamo DB rule",
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "dynamoDB": {
                "hashKeyField": "key",
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_dynamoDB",
                 "tableName": "my_ddb_table",
                 "hashKeyValue": "${topic()}",
                 "rangeKeyValue": "${timestamp()}",
                 "rangeKeyField": "timestamp"
             }
        }]
    }
}
```

如需詳細資訊，請參閱 Amazon DynamoDB 入門指南。

DynamoDBv2 動作

DynamoDBv2 Action

`dynamoDBv2` 動作可將全部或部分的 MQTT 訊息寫入至 DynamoDB 表格。承載的每個屬性都會寫入 DynamoDB 資料庫不同的欄中。

More information (4)

建立 DynamoDB 規則時，請務必指定以下資訊：

roleARN

IAM 角色，允許存取 DynamoDB 表格。該角色至少必須允許 `dynamoDB:PutItem`IAM 動作。

tableName

DynamoDB 表格的名稱。

Note

MQTT 訊息承載一定要包含一個與該表格主要分割區索引鍵相符的根層級索引鍵，以及一個與該表格的主要排序索引鍵相符的根層級索引鍵 (若有定義)。

寫入 DynamoDB 表格的資料產自於規則的 SQL 陳述式。

Note

請務必確認關聯至規則的角色擁有授予 `dynamodb:PutItem` 權限的政策。

下列 JSON 範例顯示了如何定義 `dynamoDB` 規則內的 AWS IoT 動作：

```
{
    "topicRulePayload": {
```

```
        "sql": "SELECT * AS message FROM 'some/topic'",
        "ruleDisabled": false,
        "description": "A test DynamoDBv2 rule",
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "dynamoDBv2": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_dynamoDBv2",
                "putItem": {
                    "tableName": "my_ddb_table"
                }
             }
        }]
    }
}
```

如需詳細資訊，請參閱 Amazon DynamoDB 入門指南。

Elasticsearch 動作

Elasticsearch Action

`elasticsearch` 動作可從 MQTT 訊息將資料寫入 Amazon Elasticsearch Service 網域。接下來，Elasticsearch 中的資料可使用例如 Kibana 的工具來進行查詢和視覺化。

More information (5)

當您建立含有 AWS IoT 動作的 `elasticsearch` 規則時，請務必指定下列資訊：

endpoint

您的 Amazon Elasticsearch Service 網域端點。

index

您想儲存資料的 Elasticsearch 索引。

type

欲存放文件的類型。

id

各文件的專屬識別符。

Note

請務必確認關聯至規則的角色擁有授予 `es:ESHttpPut` 權限的政策。

下列 JSON 範例顯示了如何定義 `elasticsearch` 規則內的 AWS IoT 動作：

```
{
  "topicRulePayload":{
    "sql":"SELECT *, timestamp() as timestamp FROM 'iot/test'",
    "ruleDisabled":false,
    "awsIotSqlVersion": "2016-03-23",
    "actions":[{
        "elasticsearch":{
          "roleArn":"arn:aws:iam::123456789012:role/aws_iot_es",
          "endpoint":"https://my-endpoint",
          "index":"my-index",
          "type":"my-type",
          "id":"${newuuid()}"
```

```
        }
    }]
  }
}
```

如需詳細資訊，請參閱 Amazon Elasticsearch Service 開發人員指南。

Firehose 動作

Firehose Action

`firehose` 動作會從觸發規則的 MQTT 訊息發送資料至 Kinesis Data Firehose 串流。

More information (6)

當您建立含有 `firehose` 動作的規則時，請務必指定下列資訊：

deliveryStreamName

要寫入訊息資料的 Kinesis Data Firehose 串流。

roleArn

允許存取 IAM 的 Kinesis Data Firehose 角色。

separator

用於區隔寫入至 Kinesis Data Firehose 串流記錄的字元分隔符號。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r\n' (Windows 換行)、',' (逗號)。

Note

請務必確保關聯至規則的角色擁有授予 `firehose:PutRecord` 權限的政策。

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `firehose` 規則：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "firehose": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_firehose",
                "deliveryStreamName": "my_firehose_stream"
            }
        }]
    }
}
```

如需詳細資訊，請參閱 Kinesis Data Firehose 開發人員指南。

HTTP 動作

HTTP Action

此 `http` 動作會從觸發規則的 MQTT 訊息傳送資料至您的 Web 應用程式或服務，以便進一步處理，而不需撰寫任何程式碼。必須先驗證您傳送資料至其中的端點，然後規則引擎才能使用它。

More information (7)

當您建立含有 `http` 動作的規則時，請務必指定下列資訊：

url

透過 HTTP POST 傳送訊息的 HTTPS URL。您可以在 URL 中使用替代範本。

confirmationUrl

選用。若已指定，則 AWS IoT 會使用確認 URL 來建立相符的主題規則目的地。您必須先啟用主題規則目的地，才能在 `http` 動作中使用它。如需詳細資訊，請參閱 使用主題規則目的地 (p. 281)。如果您使用替代範本，則必須手動建立主題規則目的地，然後才能使用 `http` 動作。`confirmationUrl` 必須是 `url` 的字首。

`url` 與 `confirmationUrl` 之間的關係如下所述：

- 若對 `url` 進行硬式編碼且未提供 `confirmationUrl`，則我們會隱含地將 `url` 欄位視為 `confirmationUrl`。 AWS IoT 會建立 `url` 的主題規則目的地。
- 若對 `url` 和 `confirmationUrl` 進行硬式編碼，則 `url` 必須以 `confirmationUrl` 開頭。AWS IoT 會建立 `confirmationUrl` 的主題規則目的地。
- 如果 `url` 包含替代範本，則您必須指定 `confirmationUrl` 且 `url` 必須以 `confirmationUrl` 開頭。如果 `confirmationUrl` 包含替代範本，則您必須手動建立主題規則目的地，然後才能使用 `http` 動作。如果 `confirmationUrl` 未包含替代範本，則 AWS IoT 會建立 `confirmationUrl` 的主題規則目的地。

標頭

選用。您想要包含在 HTTP 請求中的任何標頭。

金鑰

標題的金鑰。不支援替代範本。

值

標頭的值。支援替代範本。

auth

選用。規則引擎用來連線至 `url` 引數中指定之端點 URL 的身分驗證。目前，簽章版本 4 是唯一支援的身分驗證類型。如需詳細資訊，請參閱 HTTP 授權。

下列 JSON 範例顯示了如何建立含有 `http` 動作的 AWS IoT 規則：

```
{
    "topicRulePayload": {
        "sql": "select * from 'a/b'",
        "awsIotSqlVersion": "2016-03-23",
        "ruleDisabled": false,
        "actions": [{
            "http": {
                "url": "https://www.example.com/subpath",
                "confirmationUrl": "https://www.example.com",
                "headers": [{
                    "key": "static_header_key",
                    "value": "static_header_value"
                },
                {
                    "key": "substitutable_header_key",
                    "value": "${value_from_payload}"
                }]
            }
        }]
```

```
    }
}
```

當承載採用 JSON 格式時，預設內容類型為 application/json。否則，它是 application/octet-stream。您可以在標頭中使用重要的 content-type 指定確切的內容類型 (不區分大小寫) 來覆寫它。

HTTP action retry logic

AWS IoT 規則引擎會根據下列規則重試 `http` 動作：

- 規則引擎會嘗試傳送訊息至少一次。
- 規則引擎最多重試兩次。嘗試次數上限為三次。
- 在下列情況下，規則引擎不會嘗試重試：
  - 上一次嘗試提供了大於 16384 個位元組的回應。
  - 下游 Web 服務或應用程式在嘗試之後關閉 TCP 連線。
  - 完成結束請求的總時間超過請求逾時限制。
  - 請求傳回了 429、500-599 以外的 HTTP 狀態碼。

> **Note**
>
> 標準資料傳輸成本適用於重試。

IoT 分析動作

IoT Analytics Action

`iotAnalytics` 動作會從觸發規則的 MQTT 訊息發送資料至 AWS IoT Analytics 頻道。

More information (8)

當您建立含有 `iotAnalytics` 動作的規則時，請務必指定下列資訊：

channelName

要寫入訊息資料的 AWS IoT Analytics 頻道名稱。

roleArn

允許存取 IAM 頻道的 AWS IoT Analytics 角色。

您指定連接到該角色的政策應該如下所示：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iotanalytics:BatchPutMessage",
            "Resource": [
                "arn:aws:iotanalytics:us-west-2:<your-account-number>:channel/
mychannel"
            ]
        }
    ]
}
```

而且具有如下所示的信任關係：

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

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `iotAnalytics` 規則：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "iotAnalytics": {
                "channelName": "mychannel",
                "roleArn": "arn:aws:iam::123456789012:role/analyticsRole",
            }
        }]
    }
}
```

如需詳細資訊，請參閱 AWS IoT Analytics 使用者指南。

AWS IoT Analytics 主控台也有 Quick start (快速入門) 功能，讓您按一下即可建立通道、資料存放區和管道。在您開啟 AWS IoT Analytics 主控台時尋找此頁面：

IoT 事件動作

IoT Events Action

`iotEvents` 動作會從觸發規則至 AWS IoT Events 輸入的 MQTT 訊息傳送資料。

More information (9)

當您建立含有 `iotEvents` 動作的規則時，請務必指定下列資訊：

inputName

AWS IoT Events 輸入的名稱。

messageId

選用。使用此項目以確保 AWS IoT Events 偵測器僅處理一個帶有 messageId 的輸入 (訊息)。

roleArn

授予 AWS IoT 權限以傳送輸入至 AWS IoT Events 偵測器的角色 ARN。(`"Action":"iotevents:BatchPutMessage"`)。

以下是應該連接到角色的範例信任政策：

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": "iotevents:BatchPutMessage",
        "Resource": [ * ]
    }
}
```

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `iotEvents` 規則：

```
{
    "topicRulePayload": {
      "sql": "expression",
      "ruleDisabled": false,
      "description": "An AWS IoT Events test rule",
      "awsIotSqlVersion": "2016-03-23",
      "actions": [{
          "iotEvents": {
            "inputName": "MyIoTEventsInput",
            "messageId": "1234567890",
            "roleArn": "arn:aws:iam::123456789012:role/aws_iot_events"
          },
      }]
    }
}
```

如需詳細資訊，請參閱 AWS IoT Events 開發人員指南。

IoT SiteWise 動作

IoT SiteWise Action

`iotSiteWise` 動作會從觸發規則的 MQTT 訊息傳送資料至 AWS IoT SiteWise 中的資產屬性。

More information (10)

在建立含有 `iotSiteWise`動作的規則時，請務必指定下列資訊：

putAssetPropertyValueEntries

資產屬性值項目的清單，其中每個項目都包含下列資訊：

entryId

選用。此項目的唯一識別符。您可以定義 `entryId`，以更好地追蹤在發生失敗時哪則訊息造成錯誤。接受替代範本。預設為新的 UUID。

propertyAlias

與資產屬性相關聯的屬性別名。您必須指定 `propertyAlias`，或同時指定 `aliasId` 和 `propertyId`。接受替代範本。

assetId

AWS IoT SiteWise 資產的 ID。您必須指定 `propertyAlias`，或同時指定 `aliasId` 和 `propertyId`。接受替代範本。

propertyId

資產屬性的 ID。您必須指定 `propertyAlias`，或同時指定 `aliasId` 和 `propertyId`。接受替代範本。

propertyValues

要插入的屬性值清單，其中每個屬性值包含以下格式的時間戳記、品質和值 (TQV)：

timestamp

包含下列資訊的時間戳記結構：

timeInSeconds

包含自 epoch (UTC) 起之時間 (以秒為單位) 的字串。接受替代範本。

offsetInNanos

選用。包含奈米秒時間偏移的字串。接受替代範本。

quality

選用。描述值品質的字串。接受替代範本。必須是 `GOOD`、 `BAD`或 `UNCERTAIN`。

值

包含下列其中一個值欄位的值結構，視資產屬性的資料類型而定：

booleanValue

選用。包含值項目之布林值的字串。接受替代範本。

doubleValue

選用。包含值項目之雙精確度值的字串。接受替代範本。

integerValue

選用。包含值項目之整數值的字串。接受替代範本。

stringValue

選用。值項目的字串值。接受替代範本。

roleArn

許可 AWS IoT 將資產屬性值傳送至 AWS IoT SiteWise 之角色的 ARN (`"Action": "iotsitewise:BatchPutAssetPropertyValue"`)。

以下是應該連接到角色的範例信任政策：

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iotsitewise:BatchPutAssetPropertyValue",
            "Resource": "*"
        }
    ]
}
```

若要改善安全性，您可以在 `Condition` 屬性中指定 AWS IoT SiteWise 資產階層路徑。下列範例是指定資產階層路徑的信任政策。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iotsitewise:BatchPutAssetPropertyValue",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iotsitewise:assetHierarchyPath": [
                        "root node asset ID/*"
                    ]
                }
            }
        }
    ]
}
```

下列 JSON 範例顯示了如何建立含有 `iotSiteWise` 動作的 AWS IoT 規則：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "iotSiteWise": {
                "putAssetPropertyValueEntries": [
                    {
                        "propertyAlias": "/some/property/alias",
                        "propertyValues": [
                            {
                                "timestamp": {
                                    "timeInSeconds": "${my.payload.timeInSeconds}"
                                },
                                "value": {
                                    "integerValue": "${my.payload.value}"
                                }
                            }
                        ]
                    }
                ],
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_sitewise"
            },
        }]
    }
}
```

如需詳細資訊，請參閱 AWS IoT SiteWise 開發人員指南。

Kinesis 動作

Kinesis Action

`kinesis` 動作可從 MQTT 訊息將資料寫入 Kinesis 串流。

More information (11)

在建立含有 AWS IoT 動作的 `kinesis` 規則時，請務必指定下列資訊：

stream

要寫入資料的 Kinesis 串流。

partitionKey

分割區索引鍵用來決定資料要寫入哪個碎片。分割區索引鍵通常是由表達式 (如「${topic()}」或「${timestamp()}」) 組成。

Note

請務必確認與該規則相連的政策擁有 `kinesis:PutRecord`許可。

下列 JSON 範例顯示了如何定義 `kinesis` 規則內的 AWS IoT 動作：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "kinesis": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_kinesis",
                "streamName": "my_kinesis_stream",
                "partitionKey": "${topic()}"
            }
        }],
    }
}
```

如需詳細資訊，請參閱 Kinesis 開發人員指南。

Lambda 動作

Lambda Action

`lambda` 動作會呼叫 Lambda 函式，傳送觸發規則的 MQTT 訊息。Lambda 函式是以非同步方式執行。

More information (12)

Lambda 函式是以非同步方式執行。

若要讓 AWS IoT 呼叫 Lambda 函數，必須設定一個將 `lambda:InvokeFunction` 許可授與 AWS IoT 的政策。您只能呼叫與您 Lambda 政策存在之相同區域中定義的 Lambda 函數。Lambda 函數會使用資源型政策，因此您必須將政策連接到 Lambda 函數本身。請使用下列 CLI 命令來連接用於授與 `lambda:InvokeFunction`權限的政策：

```
aws lambda add-permission --function-name "function_name" --region "region" --principal
 iot.amazonaws.com --source-arn arn:aws:iot:us-east-2:account_id:rule/rule_name --
source-account "account_id" --statement-id "unique_id" --action "lambda:InvokeFunction"
```

以下為 `add-permission`命令的引數：

--function-name

正要新增許可來更新資源政策的 Lambda 函式名稱。

--region

您帳戶的 AWS 區域。

--principal

取得許可的委託人。此應由 `iot.amazonaws.com` 允許 AWS IoT 許可來呼叫 Lambda 函式。

--source-arn

該項規則的 ARN。您可以使用 `get-topic-rule`CLI 命令來取得規則的 ARN。

--source-account

定義規則的 AWS 帳戶。

--statement-id

專屬的陳述式識別符。

--action

想在此陳述式中允許的 Lambda 動作。如要允許 AWS IoT 叫用 Lambda 函數，請指定 `lambda:InvokeFunction`。

Note

若您在未提供來源 ARN 的情況下為 AWS IoT 主體新增權限，則任何搭配您 Lambda 動作建立規則的 AWS 帳戶都可以觸發規則，以從 AWS IoT 叫用您的 Lambda 函數。

如需詳細資訊，請參閱 Lambda 許可模型。

建立含有 `lambda` 動作的規則時，您必須指定觸發規則時要叫用的 Lambda 函式。

下列 JSON 範本所示範的是呼叫 Lambda 函式的規則：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "lambda": {
                "functionArn": "arn:aws:lambda:us-
east-2:123456789012:function:myLambdaFunction"
             }
        }]
    }
}
```

若您不指定您 Lambda 函數的版本或別名，則會執行最新的函數版本。如果您想要執行特定版本的 Lambda 函數，您可以指定版本或別名。若要指定版本或別名，請將版本或別名附加到 Lambda 函數的 ARN。例如：

```
"arn:aws:lambda:us-east-2:123456789012:function:myLambdaFunction:someAlias"
```

如需版本控制與別名的詳細資訊，請參閱 AWS Lambda 函數版本控制與別名。如需 AWS Lambda 的詳細資訊，請參閱 AWS Lambda 開發人員指南。

Republish 動作

Republish Action

`republish`動作可將觸發角色的訊息重新發佈至另一個 MQTT 主題。

More information (13)

當您建立含有 `republish` 動作的規則時，請務必指定下列資訊：

topic

要將訊息重新發佈至該 MQTT 主題。如果您要重新發佈到預留主題 (開頭為 `$`)，請改用 `$$`。例如，若您正在重新發佈裝置陰影主題 (例如 `$aws/things/MyThing/shadow/update`)，請將主題指定為 `$aws/things/MyThing/shadow/update`。

roleArn

允許發佈至 MQTT 主題的 IAM 角色。

qos

選用。重新發佈訊息時使用服務品質 (QoS) 層級。有效值為 `0`或 `1`。預設值為 `0`。

Note

請務必確認關聯至規則的角色擁有授予 `iot:Publish` 權限的政策。

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "republish": {
                "topic": "another/topic",
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_republish",
                "qos": 1
            }
        }]
    }
}
```

S3 動作

S3 Action

`s3` 動作會將觸發規則的 MQTT 訊息的資料寫入至 Amazon S3 儲存貯體。

More information (14)

在搭配 `s3` 動作建立 AWS IoT 規則時，除可選的 `cannedacl` 之外，您必須指定以下資訊：

bucket

要寫入資料的 Amazon S3 儲存貯體。

cannedacl

選用。Amazon S3 固定的 ACL，負責控制物件金鑰所辨識之物件的存取權限。如需更多資訊 (包含允許的值)，請參閱 S3 Canned ACL。

金鑰

寫入資料的檔案路徑。例如，如果此引數的值為「${topic()}/${timestamp()}」，則訊息傳送的主題就會是「this/is/my/topic,」。若目前的時間戳記為 1460685389，則該資料會寫入至 Amazon S3 上的「this/is/my/topic」資料夾中一個名為「1460685389」的檔案。

> **Note**
>
> 使用靜態金鑰會導致每次呼叫規則時便會覆寫 Amazon S3 內的一個檔案。更為常用的案例是使用訊息時間戳記，或另一個獨一無二的訊息識別符，以便每一個接收到的訊息都會將新檔案儲存在 Amazon S3 中。

roleArn

允許存取 IAM 儲存貯體的 Amazon S3 角色。

> **Note**
>
> 請務必確認關聯至規則的角色擁有授予 `s3:PutObject` 權限的政策。

下列 JSON 範例顯示了如何定義 `s3` 規則內的 AWS IoT 動作：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "s3": {
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_s3",
                "bucketName": "my-bucket",
                "key": "${topic()}/${timestamp()}"
                "cannedacl": "public-read"
            }
        }]
    }
}
```

如需詳細資訊，請參閱 Amazon S3 開發人員指南。

Salesforce 動作

Salesforce Action

`salesforce` 動作會從觸發規則至 Salesforce IoT 輸入串流的 MQTT 訊息傳送資料。

More information (15)

當您建立含有 `salesforce` 動作的規則時，請務必指定下列資訊：

url

該 URL 會透過 Salesforce IoT 輸入串流公開。在您建立輸入串流時，便可從 Salesforce IoT 平台使用 URL。若需更多資訊，請參閱 Salesforce IoT 文件。

token

用於驗證特定 Salesforce IoT 輸入串流存取的權杖。在您建立輸入串流時，您便可在 Salesforce IoT 平台使用權杖。若需更多資訊，請參閱 Salesforce IoT 文件。

Note

這些參數不支援替換。

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `salesforce` 規則：

```
{
    "topicRulePayload": {
      "sql": "expression",
      "ruleDisabled": false,
      "awsIotSqlVersion": "2016-03-23",
      "actions": [{
          "salesforce": {
              "token": "ABCDEFGHI123456789abcdefghi123456789",
              "url": "https://ingestion-cluster-id.my-env.sfdcnow.com/streams/stream-
id/connection-id/my-event"
          }
      }]
    }
}
```

若需更多資訊，請參閱 Salesforce IoT 文件。

SNS 動作

SNS Action

`sns`動作會從觸發規則的 MQTT 訊息以 SNS 推送通知的形式傳送資料。

More information (16)

當您建立含有 `sns` 動作的規則時，請務必指定下列資訊：

messageFormat

訊息的格式。可接受的值為「JSON」和「RAW」。 該屬性的預設值為「RAW」。 SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。

roleArn

允許存取 SNS 的 IAM 角色。

targetArn

推送通知傳送到的 SNS 主題或個別裝置。

Note

請務必確保與規則關聯的政策擁有 `sns:Publish` 權限。

下列 JSON 範例顯示了如何定義 `sns` 規則內的 AWS IoT 動作：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
```

```
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "sns": {
                "targetArn": "arn:aws:sns:us-east-2:123456789012:my_sns_topic",
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_sns"
            }
        }]
    }
}
```

如需詳細資訊，請參閱 Amazon SNS 開發人員指南。

SQS 動作

SQS Action

`sqs` 動作會從觸發規則至 SQS 佇列的 MQTT 訊息傳送資料。

More information (17)

當您建立含有 `sqs` 動作的規則時，請務必指定下列資訊：

queueUrl

要寫入訊息資料的 SQS 佇列 URL。

useBase64

如果希望 MQTT 訊息資料在寫入 SQS 佇列前為 Base64 編碼，請設定為 `true`。否則，請設定為 `false`。

roleArn

允許存取 SQS 佇列的 IAM 角色。

Note

請務必確認關聯至規則的角色擁有授予 `sqs:SendMessage` 權限的政策。

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `sqs` 規則：

```
{
    "topicRulePayload": {
        "sql": "SELECT * FROM 'some/topic'",
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [{
            "sqs": {
                "queueUrl": "https://sqs.us-east-2.amazonaws.com/123456789012/
my_sqs_queue",
                "roleArn": "arn:aws:iam::123456789012:role/aws_iot_sqs",
                "useBase64": false
            }
        }]
    }
}
```

SQS 動作並不支援 SQS FIFO 佇列。因為規則引擎是一項完全分散式的服務，因此在 SQS 動作觸發時無法保證訊息順序。

如需詳細資訊，請參閱 Amazon SQS 開發人員指南。

Step Functions 動作

Step Functions Action

`stepFunctions` 動作會開始 Step Functions 狀態機器的執行。

More information (18)

當您建立含有 `stepFunctions` 動作的規則時，請務必指定下列資訊：

executionNamePrefix

選用。指定給狀態機器執行的名稱包含此字首，後為 UUID。若狀態機器執行的名稱並未提供，則 Step Functions 會建立一個唯一的名稱。

stateMachineName

將啟動執行的 Step Functions 狀態機器的名稱。

roleArn

授與物聯網 (AWS IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。

以下是應該連接到角色的範例信任政策：

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": "states:StartExecution",
        "Resource": [ * ]
    }
}
```

下列 JSON 範例顯示了如何建立含有 AWS IoT 動作的 `stepFunctions` 規則：

```
{
    "topicRulePayload": {
      "sql": "expression",
      "ruleDisabled": false,
      "description": "A step functions test rule",
      "awsIotSqlVersion": "2016-03-23",
      "actions": [{
          "stepFunctions": {
              "executionNamePrefix": "myExecution",
              "stateMachineName": "myStateMachine",
              "roleArn": "arn:aws:iam::123456789012:role/aws_iot_step_functions"
          }
      }]
    }
}
```

如需詳細資訊，請參閱 Step Functions 開發人員指南。

規則的故障診斷

如果您遇到規則方面的問題，請啟用 CloudWatch Logs。對記錄進行分析後，即可判斷該問題是否在於授權，或者是否為 WHERE 子句條件不相符之類的問題。如需更多資訊，請參閱設定 CloudWatchLogs。

錯誤處理 (錯誤動作)

當 AWS IoT 收到裝置的訊息時，規則引擎會檢查該訊息是否符合規則。若符合，則會對該項規則的 SQL 陳述式進行評估，並觸發該規則的動作，傳送 SQL 陳述式的結果。

如果觸發動作時發生問題，規則引擎會觸發錯誤動作 (如果有為指定規則動作)。這可能在以下情況時發生：

- 規則未擁有存取 Amazon S3 儲存貯體的許可。
- 使用者的錯誤導致 DynamoDB 佈建的傳輸量超量。

錯誤動作訊息格式

每一項規則和訊息會產生單條訊息。例如，如果相同規則中的兩個規則動作失敗，則錯誤動作會收到包含那兩個錯誤的訊息。

錯誤動作訊息將如下所示：

```
{
  "ruleName": "TestAction",
  "topic": "testme/action",
  "cloudwatchTraceId": "7e146a2c-95b5-6caf-98b9-50e3969734c7",
  "clientId": "iotconsole-1511213971966-0",
  "base64OriginalPayload": "ewogICJtZXNzYWdlIjogIkhlbGxvIHZyb20gQVdTIElvVCBjb25zb2xlIgp9",
  "failures": [
    {
      "failedAction": "S3Action",
      "failedResource": "us-east-1-s3-verify-user",
      "errorMessage": "Failed to put S3 object. The error received was The
 specified bucket does not exist (Service: Amazon S3; Status Code: 404; Error
 Code: NoSuchBucket; Request ID: 9DF5416B9B47B9AF; S3 Extended Request ID:
 yMah1cwPhqTH267QLPhTKeVPKJB8BO5ndBHzOmWtxLTM6uAvwYYuqieAKyb6qRPTxP1tHXCoR4Y=).
 Message arrived on: error/action, Action: s3, Bucket: us-
east-1-s3-verify-user, Key: \"aaa\". Value of x-amz-id-2:
 yMah1cwPhqTH267QLPhTKeVPKJB8BO5ndBHzOmWtxLTM6uAvwYYuqieAKyb6qRPTxP1tHXCoR4Y="
    }
  ]
}
```

ruleName

觸發錯誤動作的規則名稱。

topic

收到原始訊息的主題。

cloudwatchTraceId

指向 CloudWatch 內的錯誤記錄的專屬識別資料。

clientId

訊息發佈者的用戶端 ID。

base64OriginalPayload

原始訊息酬載 Base64 編碼。

failures

failedAction

無法完成的動作名稱 (例如，「S3Action」)。

failedResource

資源名稱 (例如，S3 儲存貯體名稱)。

errorMessage

對該項錯誤的敘述和說明。

錯誤動作範例

以下範例為有加入錯誤動作的規則。以下規則中有個動作會將訊息寫入至 DynamoDB 表格，還有一項錯誤動作，會將資料寫入 Amazon S3 儲存貯體：

```
{
    "sql" : "SELECT * FROM ..."
    "actions" : [{
        "dynamoDB" : {
            "table" : "PoorlyConfiguredTable",
            "hashKeyField" : "AConstantString",
            "hashKeyValue" : "AHashKey"}}
    ],
    "errorAction" : {
        "s3" : {
            "roleArn": "arn:aws:iam::123456789012:role/aws_iot_s3",
            "bucket" : "message-processing-errors",
            "key" : "${replace(topic(), '/', '-') + '-' + timestamp() + '-' + newuuid()}"
        }
    }
}
```

可以在錯誤動作的 SQL 陳述式中使用任何函式或替代方式，除了外部函式之外 (如 `get_thing_shadow`、`aws_lambda` 和 `machinelearning_predict`)。

如需規則的相關資訊或是了解如何指定錯誤動作，請參閱建立AWS IoT規則。

若需更多有關使用 CloudWatch 監控規則成功或失敗的資訊，請參閱 AWS IoT 指標與維度 (p. 202)。

使用主題規則目的地

目的地是定義規則引擎可以路由資料的資源。目的地可以跨規則重複使用，且可能需要確認或組態，才能使用目的地。目的地可讓規則引擎將資料傳送至未與 AWS IoT 原生整合的其他服務。

主題規則目的地是用來驗證您是否擁有或可以存取要路由資料至其中的端點。當您建立具有端點的規則動作 (例如 `http` 動作) 或更新現有規則動作的端點時，AWS IoT 會將確認訊息傳送到包含唯一字符的端點。若要驗證您的端點，您必須使用字符呼叫 `ConfirmTopicRuleDestination`，以驗證您是否擁有或可以存取端點。字符有效時間為三天，三天後您需要建立新的 `http` 動作，或呼叫 `UpdateTopicRuleDestination` API 重新啟動確認程序。您也可以瀏覽至 `enableUrl` 來確認您的主題規則目的地，或從 AWS IoT 主控台中的 Destination (目的地) 頁面提供字符。

當 AWS IoT 收到傳送至端點的字符時，表示確認了目的地。您必須啟用目的地，然後規則引擎才能使用它。目的地可以處於下列其中一種狀態：

ENABLED

目的地已啟用。目的地必須處於 `ENABLED` 狀態，才能在規則中使用它。您只能啟用處於 DISABLED 狀態的目的地。

DISABLED

目的地已確認，但已停用。如果您想暫時防止流量傳至端點，而不必再次進行確認程序，這會很有用。您只能停用處於 ENABLED 狀態的目的地。

IN_PROGRESS

正在確認目的地。

ERROR

目的地確認逾時。

在確認並啟用它們之後，即可在您的帳戶中搭配任何規則使用目的地。

建立主題規則目的地

使用 AWS IoT 來建立具有 `http` 動作的規則時，就會建立目的地。您也可以使用 `CreateTopicRuleDestination` API 或 AWS IoT 主控台建立目的地。

當您建立目的地時，AWS IoT 會驗證端點 URL 是否有效。如果 URL 有效，則確認訊息會傳送至該 URL。確認請求的格式如下：

```
HTTP POST {confirmationUrl}/?confirmationToken={confirmationToken}
Headers:
x-amz-rules-engine-message-type: DestinationConfirmation
x-amz-rules-engine-destination-arn:"arn:aws:iot:us-east-1:123456789012:ruledestination/
http/7a280e37-b9c6-47a2-a751-0703693f46e4"
Content-Type: application/json
Body:
{
    "arn":"arn:aws:iot:us-east-1:123456789012:ruledestination/http/7a280e37-b9c6-47a2-
a751-0703693f46e4",
    "confirmationToken": "AYADeMXLrPrNY2wqJAKsFNn-…NBJndA",
    "enableUrl": "https://iot.us-east-1.amazonaws.com/confirmdestination/
AYADeMXLrPrNY2wqJAKsFNn-…NBJndA",
    "messageType": "DestinationConfirmation"
}
```

此訊息的欄位定義如下：

arn

要確認之主題規則目的地的 ARN。

confirmationToken

確認字符。範例中的字符會被截斷。您的字符將更長。

enableUrl

您瀏覽以確認主題規則目的地的 URL。

messageType

訊息的類型。

確認主題規則目的地

您可對包含在確認訊息的 `enableUrl` 提出 HTTP GET 請求來確認目的地。您可以從確認消息中使用字符呼叫 `ConfirmTopicRuleDestination`，或者您可以使用 AWS IoT 主控台。

字符必須有效，目的地才能置於 ENABLED 狀態中。如果字符已過期，您必須呼叫 UpdateTopicRuleDestination 以重新啟動確認程序。

> Note
>
> 如果您透過呼叫 ConfirmTopicRuleDestination 來確認主題規則目的地，則目的地狀態將會是 DISABLED，而且您必須先呼叫 UpdateTopicRuleDestination 再使用它來明確啟用您的目的地。

停用主題規則目的地

若要停用目的地，請呼叫 UpdateTopicRuleDestination，並將主題規則目的地的狀態設為 DISABLED。

啟用主題規則目的地

若要啟用目的地，請呼叫 UpdateTopicRuleDistination，並將主題規則的狀態設為 ENABLED。您不需要重新驗證 URL。

傳送新的確認訊息

若要觸發目的地的新確認訊息，請呼叫 UpdateTopicRuleDistination，並將主題規則目的地的狀態設為 IN_PROGRESS。

刪除主題規則目的地

若要刪除主題規則目的地，請呼叫 DeleteTopicRuleDestination。

AWS IoTSQL 參考

在 AWS IoT 中，規則是使用類似 SQL 的語法來定義。SQL 陳述式是由三種類型的子句所組成：

SELECT

必要.從傳入的訊息承載擷取資訊並執行轉換。

FROM

MQTT 訊息主題篩選條件。系統會為每則傳送到符合此處指定篩選條件之 MQTT 主題的訊息觸發該規則。對於透過訊息觸發的規則是必要的，該訊息是透過訊息中介裝置傳遞的。若為僅使用 基本擷取 (p. 331) 功能觸發的規則，則為選用。

WHERE

選用。新增條件邏輯，其會判斷是否執行某規則指定的動作。

SQL 陳述式範例如下所示：

```
SELECT color AS rgb FROM 'a/b' WHERE temperature > 50
```

MQTT 訊息 (也稱作傳入承載) 的範例如下所示：

```
{
    "color":"red",
    "temperature":100
}
```

如果此訊息是發佈在 `'a/b'`主題上，便會觸發規則，SQL 陳述式惠受到評估。如果 `color`屬性大於 50，SQL 陳述式會擷取 `"temperature"` 屬性的值。WHERE 子句會指定條件 `temperature > 50`。`AS`關鍵字將 `"color"` 屬性重新命名為 `"rgb"`。結果 (也稱作傳出酬載) 會如以下所示：

```
{
    "rgb":"red"
}
```

該資料接下來會轉送給該規則的動作，並發送資料以進行更多處理作業。如需規則動作的詳細資訊，請參閱 AWS IoT規則動作 (p. 260)。

資料類型

AWS IoT 規則引擎可支援所有 JSON 資料類型。

支援的資料類型

類型	意義
`Int`	分散的 `Int`。最大 34 位數。
`Decimal`	精度為 34 位的 `Decimal`，最小非零量級為 1E-999，最大量級為 9.999…E999。 Note 有些函數傳回的 `Decimal`為雙精度，而非 34 位數的精度。
`Boolean`	`True` 或 `False`.
`String`	UTF-8 字串。
`Array`	一系列類型無需相同的值。
`Object`	由鍵和值組成的 JSON 值。鍵必須為字串。值可以為任何類型。
`Null`	`Null`是由 JSON 定義。此為用來代表缺少某個值的實際值。您可以在 SQL 陳述式中使用 `Null`的關鍵字來明確建立 `Null` 的值。例如：`"SELECT NULL AS n FROM 'a/b'"`
`Undefined`	並非值。無法以 JSON 明確顯示，除非是省略該值。舉例而言，在物件 `{"foo": null}`中，「foo」鍵會傳回 NULL，但「bar」鍵傳回 `Undefined`。SQL 語言在內部將 `Undefined`視為一個值，但無法以 JSON 顯示，因此序列化為 JSON 時，結果會是 `Undefined`。 `{"foo":null, "bar":undefined}` 序列化為 JSON 如下：

類型	意義
	`{"foo":null}` 同樣地，`Undefined`由自身序列化時，會轉換為空白字串。以無效引數 (例如，錯誤類型、錯誤的引數數量等等) 呼叫的函數會傳回 `Undefined`。

轉換

下表列出的是當某個類型的值轉換為另一種類型 (給函數指定錯誤類型的值) 所得的結果。舉例而言，如果絕對值的函數「abs」(預期為 `Int`或 `Decimal`) 得到的是 `String`，其會嘗試按照下列規則將 `String` 轉換為 `Decimal`。此例中，'abs("-5.123")' 會視為 'abs(-5.123)'。

> **Note**
>
> 不會企圖轉換為 `Array`、`Object`、`Null`、`Undefined`。

To Decimal

引數類型	結果
`Int`	無小數點的 `Decimal`。
`Decimal`	來源值。
`Boolean`	`Undefined`。(您可以明確使用 Cast 函數轉換 true = 1.0、false = 0.0。)
`String`	SQL 引擎會嘗試將字串剖析為 `Decimal`。AWS IoT 會嘗試剖析符合正規表達式的字串：`^-?\d+(\.\d+)?((?i)E-?\d+)?$`。「0」、「-1.2」、「5E-12」均為自動轉換為 `Decimal` 的範例字串。
Array	`Undefined`。
物件	`Undefined`。
Null	`Null`。
未定義	`Undefined`。

To Int

引數類型	結果
`Int`	來源值。
`Decimal`	四捨五入到最接近 `Int`的來源值。
`Boolean`	`Undefined`。(您可以明確使用 Cast 函數轉換 true = 1.0、false = 0.0。)
`String`	SQL 引擎會嘗試將字串剖析為 `Decimal`。AWS IoT 會嘗試剖析符合正規表達式的字串：`^-?\d+(\.\d+)?((?i)E-?\d+)?$`。「0」、「-1.2」、「5E-12」均為自動轉換為 `Decimal` 的範例字串。AWS IoT 會嘗試將 `String` 轉換為

引數類型	結果
	`Decimal`，然後再截斷該 `Decimal` 的小數位以產生 `Int`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Null`。
未定義	`Undefined`。

To Boolean

引數類型	結果
`Int`	`Undefined`。(您可以明確使用 `cast`函數轉換 0 = False、any_nonzero_value = True。)
`Decimal`	`Undefined`。(您可以明確使用 Cast 函數轉換 0 = False、any_nonzero_value = True。)
`Boolean`	原始的值。
`String`	「true」= True 而「false」= False (不區分大小寫)。其他字串值為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

To String

引數類型	結果
`Int`	`Int`以標準表示法的字串顯示方式。
`Decimal`	代表 `Decimal`值的字串，可能是採取科學表示法。
`Boolean`	「true」或「false」。所有小寫。
`String`	原始的值。
Array	序列化為 JSON 的 `Array`。該結果字串為以逗號分隔的清單，並包含在方括弧內。`String` 在括號中。`Decimal`、`Int`、`Boolean` 和 `Null` 則不是。
物件	序列化為 JSON 的物件。該結果字串為以逗號分隔的鍵值組清單，並以大括號為開頭和結尾。`String` 在括號中。`Decimal`、`Int`、`Boolean` 和 `Null` 則不是。
Null	`Undefined`。
未定義	未定義。

運算子

下列運算子可用於 SELECT 和 WHERE 子句。

AND 運算子

傳回 `Boolean`結果。執行邏輯 AND 運算。如果左右運算元為 true，即傳回 true。否則會傳回 false。需要 `Boolean` 運算元或不區分大小寫的「true」或「false」字串運算元。

語法： *`expression`* `AND` *`expression`*。

AND 運算子

左運算元	右運算元	輸出
`Boolean`	`Boolean`	`Boolean`。如果兩個運算元皆為 true 即為 true。否則即為 false。
`String/Boolean`	`String/Boolean`	如果所有字串均為「true」或「false」(不區分大小寫)，他們會轉換為 `Boolean`，並以 *`boolean`* `AND` *`boolean`* 的方式正常處理。
其他值	其他值	`Undefined`。

OR 運算子

傳回 `Boolean`結果。執行邏輯 OR 運算。如果右運算元或左運算元有一個為 true 即傳回 true。否則會傳回 false。需要 `Boolean` 運算元或不區分大小寫的「true」或「false」字串運算元。

語法： *`expression`* `OR` *`expression`*。

OR 運算子

左運算元	右運算元	輸出
`Boolean`	`Boolean`	`Boolean`。如果有一個運算元為 true 即為 true。否則即為 false。
`String/Boolean`	`String/Boolean`	如果所有字串均為「true」或「false」(不區分大小寫)，它們會轉換為布林值，並以 *`boolean`* `OR` *`boolean`* 的方式正常處理。
其他值	其他值	`Undefined`。

NOT 運算子

傳回 `Boolean`結果。執行邏輯 NOT 運算。如果運算元為 false 即傳回 true。否則即傳回 true。需要 `Boolean` 運算元或不區分大寫的「true」或「false」字串運算元。

語法： `NOT` *`expression`*。

NOT 運算子

運算元	輸出
`Boolean`	`Boolean`。如果運算元為 false 即為 true。否則為 true。

運算元	輸出
String	如果字串為「true」或「false」(不區分大小寫)，則會轉換為對應的布林值，並傳回相反的值。
其他值	Undefined。

> 運算子

傳回 Boolean結果。如果左運算元大於右運算元即傳回 true。兩個運算元均轉換為 Decimal，再做比較。

語法： *expression* > *expression*。

> 運算子

左運算元	右運算元	輸出
Int/Decimal	Int/Decimal	Boolean。如果左運算元大於右運算元即為 true。否則即為 false。
String/Int/Decimal	String/Int/Decimal	如果所有字串均可轉換為 Decimal，即 Boolean。如果左運算元大於右運算元即傳回 true。否則即為 false。
其他值	Undefined。	Undefined。

>= 運算子

傳回 Boolean結果。如果左運算元大於或等於右運算元，即傳回 true。兩個運算元均轉換為 Decimal，再做比較。

語法： *expression* >= *expression*。

>= 運算子

左運算元	右運算元	輸出
Int/Decimal	Int/Decimal	Boolean。如果左運算元大於或等於右運算元，即為 true。否則即為 false。
String/Int/Decimal	String/Int/Decimal	如果所有字串均可轉換為 Decimal，即 Boolean。如果左運算元大於或等於右運算元，即傳回 true。否則即為 false。
其他值	Undefined。	Undefined。

< 運算子

傳回 Boolean結果。如果左側運算元少於右運算元。兩個運算元均轉換為 Decimal，再做比較。

語法： *expression* < *expression*。

< 運算子

左運算元	右運算元	輸出
Int/Decimal	Int/Decimal	Boolean。如果左運算元小於右運算元即為 true。否則即為 false。

左運算元	右運算元	輸出
`String/Int/Decimal`	`String/Int/Decimal`	如果所有字串均可轉換為 `Decimal`，即 `Boolean`。如果左側運算元少於右運算元。否則即為 false。
其他值	`Undefined`	`Undefined`

<= 運算子

傳回 `Boolean`結果。如果左運算元小於或等於右運算元，即傳回 true。兩個運算元均轉換為 `Decimal`，再做比較。

語法： *`expression`* `<=` *`expression`*。

<= 運算子

左運算元	右運算元	輸出
`Int/Decimal`	`Int/Decimal`	`Boolean`。如果左運算元小於或等於右運算元，即為 true。否則即為 false。
`String/Int/Decimal`	`String/Int/Decimal`	如果所有字串均可轉換為 `Decimal`，即 `Boolean`。如果左運算元小於或等於右運算元，即傳回 true。否則即為 false。
其他值	`Undefined`	`Undefined`

<> 運算子

傳回 `Boolean`結果。如果左右運算元不相等，即傳回 true。否則即傳回 false。

語法： *`expression`* `<>` *`expression`*。

<> 運算子

左運算元	右運算元	輸出
`Int`	`Int`	如果左運算元不等於右運算元即為 true。否則即為 false。
`Decimal`	`Decimal`	如果左運算元不等於右運算元即為 true。否則為 false。在做比較前，先將 `Int` 轉換為 `Decimal`。
`String`	`String`	如果左運算元不等於右運算元即為 true。否則即為 false。
Array	Array	如果各運算元的項目不相等且順序不同，即為 true。否則即為 false
物件	物件	如果各運算元的鍵和值不相同，即為 true。否則即為 false。鍵/值的順序不重要。
Null	Null	False.
任何值	`Undefined`	未定義。
`Undefined`	任何值	未定義。
類型不符合	類型不符合	True。

= 運算子

傳回 Boolean結果。如果兩個左右運算元相同，即傳回 true。否則即傳回 false。

語法： *expression* = *expression*。

= 運算子

左運算元	右運算元	輸出
Int	Int	如果左運算元等於右運算元即為 true。否則即為 false。
Decimal	Decimal	如果左運算元等於右運算元即為 true。否則為 false。在做比較前，先將 Int 轉換為 Decimal。
String	String	如果左運算元等於右運算元即為 true。否則即為 false。
Array	Array	如果各運算元的項目相等且順序相同，即為 true。否則即為 false。
物件	物件	如果各運算元的鍵和值相同，即為 true。否則即為 false。鍵/值的順序不重要。
任何值	Undefined	Undefined。
Undefined	任何值	Undefined。
類型不符合	類型不符合	False.

+ 運算子

「+」是過載的運算子。可以用來連接或新增字串。

語法： *expression* + *expression*。

+ 運算子

左運算元	右運算元	輸出
String	任何值	將右運算元轉換為字串，並連接至左運算元的尾端。
任何值	String	將左運算元轉換為字串，並將右運算元連接至轉換後的左運算元的尾端。
Int	Int	Int值。將運算元相加。
Int/Decimal	Int/Decimal	Decimal值。將運算元相加。
其他值	其他值	Undefined。

- 運算子

從左運算元中減去右運算元。

語法： *expression* - *expression*。

- 運算子

左運算元	右運算元	輸出
Int	Int	Int值。從左運算元中減去右運算元。
Int/Decimal	Int/Decimal	Decimal值。從左運算元中減去右運算元。
String/Int/Decimal	String/Int/Decimal	如果所有字串都正確轉換為小數，則會傳回 Decimal 值。從左運算元中減去右運算元。否則即傳回 Undefined。
其他值	其他值	Undefined。
其他值	其他值	Undefined。

* 運算子

將左運算元乘以右運算元。

語法： *expression* * *expression*。

* 運算子

左運算元	右運算元	輸出
Int	Int	Int值。將左運算元乘以右運算元。
Int/Decimal	Int/Decimal	Decimal值。將左運算元乘以右運算元。
String/Int/Decimal	String/Int/Decimal	如果所有字串都正確轉換為小數，則會傳回 Decimal 值。將左運算元乘以右運算元。否則即傳回 Undefined。
其他值	其他值	Undefined。

/ 運算子

將左運算元除以右運算元。

語法： *expression* / *expression*。

/ 運算子

左運算元	右運算元	輸出
Int	Int	Int值。將左運算元除以右運算元。
Int/Decimal	Int/Decimal	Decimal值。將左運算元除以右運算元。
String/Int/Decimal	String/Int/Decimal	如果所有字串都正確轉換為小數，則會傳回 Decimal 值。將左運算元除以右運算元。否則即傳回 Undefined。
其他值	其他值	Undefined。

% 運算子

傳回左運算元除以右運算元的餘數。

語法： *expression* % *expression*。

% 運算子

左運算元	右運算元	輸出
Int	Int	Int值。傳回左運算元除以右運算元的餘數。
String/Int/Decimal	String/Int/Decimal	如果所有字串都正確轉換為小數，則會傳回 Decimal 值。傳回左運算元除以右運算元的餘數。否則會傳回 Undefined。
其他值	其他值	Undefined。

函數

您可以在 SQL 表達式的 SELECT 或 WHERE 子句中使用下列內建函數。

abs(Decimal)

傳回某個數字的絕對值。受 SQL 版本 2015-10-8 和更新版本支援。

範例：abs(-5)傳回 5。

引數類型	結果
Int	Int，引數的覺對值。
Decimal	Decimal，引數的覺對值。
Boolean	Undefined。
String	Decimal。結果為引數的覺對值。如果字串無法轉換，則結果為 Undefined。
Array	Undefined。
物件	Undefined。
Null	Undefined。
未定義	Undefined。

accountid()

將擁有此規則的帳戶 ID 傳回為 String。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

accountid() = "123456789012"

acos(Decimal)

以弧度傳回數字的反餘弦值。Decimal 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：acos(0)= 1.5707963267948966

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的反向餘弦值。傳回的虛數結果為 `Undefined`。
`Decimal`	`Decimal`(使用雙精度)，引數的反向餘弦值。傳回的虛數結果為 `Undefined`。
`Boolean`	`Undefined`。
`String`	`Decimal`，引數的反向餘弦值。如果字串無法轉換，則結果為 `Undefined`。傳回的虛數結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

asin(Decimal)

以弧度傳回數字的反正弦值。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`asin(0)`= 0.0

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的反向正弦值。傳回的虛數結果為 `Undefined`。
`Decimal`	`Decimal`(使用雙精度)，引數的反向正弦值。傳回的虛數結果為 `Undefined`。
`Boolean`	`Undefined`。
`String`	`Decimal`(使用雙精度)，引數的反向正弦值。如果字串無法轉換，則結果為 `Undefined`。傳回的虛數結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

atan(Decimal)

以弧度傳回數字的反正切值。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`atan(0)`= 0.0

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的反向正切值。傳回的虛數結果為 `Undefined`。
`Decimal`	`Decimal`(使用雙精度)，引數的反向正切值。傳回的虛數結果為 `Undefined`。
`Boolean`	`Undefined`。
`String`	`Decimal`，引數的反向正切值。如果字串無法轉換，則結果為 `Undefined`。傳回的虛數結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

atan2(Decimal, Decimal)

以弧度傳回 x 軸正軸和兩個引數所定義的 (x, y) 點之間的角度。 逆時針角度為正值 (上半象限，y > 0)，順時鐘的角度為負值 (下半象限 y < 0)。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`atan2(1, 0)`= 1.5707963267948966

引數類型	引數類型	結果
`Int/Decimal`	`Int/Decimal`	`Decimal`(使用雙精度度。
`Int/Decimal/String`	`Int/Decimal/String`	`Decimal`，所述之點的結果為 `Undefined`。
其他值	其他值	`Undefined`。

aws_lambda (functionArn、inputJson)

呼叫指定的 Lambda 函數，其會將 `inputJson` 傳送至 Lambda 函數，並傳回 Lambda 函數產生的 JSON。

引數

引數	描述
`functionArn`	Lambda 函數呼叫的 ARN。Lambda 函數必須傳回 JSON 資料。
`inputJson`	傳送到 Lambda 函數的 JSON 輸入。

您必須授與 AWS IoT `lambda:InvokeFunction` 許可，以呼叫指定的 Lambda 函數。下列範例顯示了如何使用 AWS CLI 授與 `lambda:InvokeFunction`的許可：

```
aws lambda add-permission --function-name "function_name"
```

```
--region "region"
--principal iot.amazonaws.com
--source-arn arn:aws:iot:us-east-1:account_id:rule/rule_name
--source-account "account_id"
--statement-id "unique_id"
--action "lambda:InvokeFunction"
```

以下為 add-permission命令的引數：

--function-name

正要新增許可來更新資源政策的 Lambda 函式名稱。

--region

您帳戶的 AWS 區域。

--principal

取得許可的委託人。此應由 `iot.amazonaws.com` 允許 AWS IoT 許可來呼叫 Lambda 函式。

--source-arn

該項規則的 ARN。您可以使用 get-topic-ruleCLI 命令來取得規則的 ARN。

--source-account

定義規則的 AWS 帳戶。

--statement-id

專屬的陳述式識別符。

--action

想在此陳述式中允許的 Lambda 動作。如要允許 AWS IoT 叫用 Lambda 函數，請指定 `lambda:InvokeFunction`。

Note

若您在未提供來源 ARN 的情況下為 AWS IoT 主體新增權限，則任何搭配您 Lambda 動作建立規則的 AWS 帳戶都可以觸發規則，以從 AWS IoT 叫用您的 Lambda 函數。如需詳細資訊，請參閱 Lambda 許可模型。

指定的 JSON 訊息酬載，如以下所示：

```
{
    "attribute1": 21,
    "attribute2": "value"
}
```

`aws_lambda` 函數可用於呼叫 Lambda 函數，如下所示：

```
SELECT
aws_lambda("arn:aws:lambda:us-east-1:account_id:function:lambda_function",
 {"payload":attribute1) as output FROM 'a/b'
```

若您想傳遞完整的 MQTT 訊息酬載，您可以使用「*」來指定 JSON 酬載。例如：

```
SELECT
```

```
aws_lambda("arn:aws:lambda:us-east-1:account_id:function:lambda_function", *) as output
 FROM 'a/b'
```

`payload.inner.element`會從發佈在「a/b」主題上的訊息選擇資料。

`some.value` 會從 Lambda 函數產生的輸出結果中選擇資料。

> **Note**
>
> 該規則引擎會限制 Lambda 函數的執行期間。規則的 Lambda 函數呼叫應該會在 2000 毫秒內完成。

bitand (Int、Int)

在兩個 `Int`(轉換的) 引數的位元表現上執行按位元的 AND 運算。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`bitand(13, 5)`= 5

引數類型	引數類型	結果
`Int`	`Int`	`Int`，兩個引數按位元
`Int/Decimal`	`Int/Decimal`	`Int`，兩個引數按位元會無條件捨去至最接近 `Int`，結果會為 `Unde`
`Int/Decimal/String`	`Int/Decimal/String`	`Int`，兩個引數按位元小數，並會無條件捨去敗，則結果為 `Undef`
其他值	其他值	`Undefined`。

bitor(Int, Int)

在兩個引數的位元表現上執行按位元的 OR 運算。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`bitor(8, 5)`= 13

引數類型	引數類型	結果
`Int`	`Int`	`Int`，兩個引數按位元
`Int/Decimal`	`Int/Decimal`	`Int`，兩個引數按位元無條件捨去至最接近的 `Undefined`。
`Int/Decimal/String`	`Int/Decimal/String`	`Int`，兩個引數按位元數，並會無條件捨去至敗，則結果為 `Undef`
其他值	其他值	`Undefined`。

bitxor(Int, Int)

在兩個 `Int`(轉換的) 引數的位元表現上執行按位元的 XOR 運算。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`bitor(13, 5)`= 8

引數類型	引數類型	結果
`Int`	`Int`	`Int`，兩個引數按位元
`Int/Decimal`	`Int/Decimal`	`Int`，兩個引數按位元件捨去至最接近的 `In`
`Int/Decimal/String`	`Int/Decimal/String`	`Int` 是在兩個引數上的無條件捨去至最接近的 `Undefined`。
其他值	其他值	`Undefined`。

bitnot(Int)

在 `Int`(轉換的) 引數的位元表現上執行按位元的 NOT 運算。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`bitnot(13)`= 2

引數類型	結果
`Int`	`Int`，引數按位元的 NOT 運算。
`Decimal`	`Int`，引數按位元的 NOT 運算。`Decimal`值會無條件捨去至最接近 `Int` 的值。
`String`	`Int`，引數按位元的 NOT 運算。字串會轉換為小數，並會無條件捨去至最接近的 `Int` (整數)。如果任何轉換失敗，則結果為 `Undefined`。
其他值	其他值。

cast()

將一個值從某個資料類型轉換至另一個類型。除了增加了將數字和布林值相互轉換的能力外，轉換行為大致如同標準轉換。如果 AWS IoT 無法判斷該如何從一種類型轉換為另一種，結果會是`Undefined`。受 SQL 版本 2015-10-8 和更新版本支援。格式：cast (*`value`* as *`type`*)。

例如：

`cast(true as Decimal)` = 1.0

在呼叫 `cast` 時，以下關鍵字可能出現在「as」之後：

對於 SQL 版本 2015-10-8 與 2016-03-23

關鍵字	結果
`Decimal`	將值轉換為 `Decimal`。
Bool	將值轉換為 `Boolean`。
`Boolean`	將值轉換為 `Boolean`。

關鍵字	結果
`String`	將值轉換為 `String`。
Nvarchar	將值轉換為 `String`。
Text	將值轉換為 `String`。
Ntext	將值轉換為 `String`。
varchar	將值轉換為 `String`。
`Int`	將值轉換為 `Int`。
整數	將值轉換為 `Int`。

此外，對於 SQL 版本 2016-03-23

關鍵字	結果
`Decimal`	將值轉換為 `Decimal`。
Bool	將值轉換為 `Boolean`。
`Boolean`	將值轉換為 `Boolean`。

轉換規則：

轉換為小數

引數類型	結果
`Int`	無小數點的 `Decimal`。
`Decimal`	來源值。
`Boolean`	true = 1.0、false = 0.0。
`String`	嘗試將字串剖析為 `Decimal`。AWS IoT 會嘗試剖析字串以符合正規表示式：^-?\d+(\.\d+)?((?i)E-?\d+)?$。「0」、「-1.2」、「5E-12」均為自動轉換為 Decimal 的範例字串。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

轉換為 Int

引數類型	結果
`Int`	來源值。
`Decimal`	無條件捨去至最接近 `Int` 的來源值。

引數類型	結果
`Boolean`	true = 1.0、false = 0.0。
`String`	嘗試將字串剖析為 `Decimal`。AWS IoT 會嘗試剖析字串以符合正規表示式：^-?\d+(\.\d+)?((?i)E-?\d+)?$。「0」、「-1.2」、「5E-12」均為自動轉換為 Decimal 的範例字串。AWS IoT 會嘗試將字串轉換為 `Decimal`，並無條件捨去至最接近的 `Int`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

轉換到 **`Boolean`**

引數類型	結果
`Int`	0 = False，any_nonzero_value = True。
`Decimal`	0 = False，any_nonzero_value = True。
`Boolean`	來源值。
`String`	「true」= True 而「false」= False (不區分大小寫)。其他字串值 = `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

轉換為字串

引數類型	結果
`Int`	`Int`以標準表示法的字串顯示方式。
`Decimal`	代表 `Decimal`值的字串，可能是採取科學表示法。
`Boolean`	「true」或「false」，均為小寫。
`String`	"true"=True 及 "false"=False (區分大小寫)。其他字串值 = `Undefined`。
Array	序列化為 JSON 的陣列。結果字串是以方括號括住，並以逗號分隔的清單。`String` 在括號中。`Decimal`、`Int` 和 `Boolean` 則不是。
物件	序列化為 JSON 的物件。JSON 字串是首尾以大括號括住，並以逗號分隔的鍵值組清單。`String` 在括號中。`Decimal`、`Int`、`Boolean` 和 `Null` 則不是。

引數類型	結果
Null	Undefined。
未定義	Undefined。

ceil(Decimal)

將指定的 Decimal無條件進位至最近的 Int。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

ceil(1.2) = 2

ceil(11.2) = -1

引數類型	結果
Int	Int，引數值。
Decimal	Int，無條件進位至最接近的 Decimal 的 Int 值。
String	Int。此字串會轉換成 Decimal並無條件進入至最接近的 Int。如果字串無法轉換為 Decimal，則結果為 Undefined。
其他值	Undefined。

chr(String)

傳回指定的 Int引數對應到的 ASCII 字元。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

chr(65)= "A"。

chr(49)= "1"。

引數類型	結果
Int	對應到指定的 ASCII 值的字元。如果引數並非有效的 ASCII 值，結果會為 Undefined。
Decimal	對應到指定的 ASCII 值的字元。Decimal的引數值會無條件捨去至最接近 Int 的值。如果引數並非有效的 ASCII 值，結果會為 Undefined。
Boolean	Undefined。
String	如果 String可以轉換為 Decimal，要無條件捨去到最接近的 Int。如果引數並非有效的 ASCII 值，結果會為 Undefined。
Array	Undefined。
物件	Undefined。

引數類型	結果
Null	`Undefined`。
其他值	`Undefined`。

clientid()

傳回傳送訊息的 MQTT 用戶端的 ID，如果訊息不是透過 MQTT 傳送，則為 `n/a`。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`clientid()` = "123456789012"

concat()

連接陣列或字串。此函數會接受任意數量的引數，並傳回 `String`或 `Array`。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`concat()` = `Undefined`。

`concat(1)` = "1"。

`concat([1, 2, 3], 4)`= [1, 2, 3, 4]。

`concat([1, 2, 3], "hello")`= [1, 2, 3, "hello"]

`concat("con", "cat")`= "concat"

`concat(1, "hello")`= "1hello"

`concat("he","is","man")`= "heisman"

`concat([1, 2, 3], "hello", [4, 5, 6])`= [1, 2, 3, "hello", 4, 5, 6]

引數數量	結果
0	`Undefined`。
1	傳回的引數未經修改。
2+	如果任一引數為 `Array`，則結果會是包含所有引數的單一陣列。若沒有列出引數，且至少有一個引數是 `String`，則結果為所有引數 `String` 表示法的串聯。使用上面列出的標準轉換將引數轉換為字串。 。

cos(Decimal)

以弧度傳回數字的餘弦值。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`cos(0)` = 1。

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的餘弦值。傳回的虛數結果為 `Undefined`。
`Decimal`	`Decimal`(使用雙精度)，引數的餘弦值。傳回的虛數結果為 `Undefined`。
`Boolean`	`Undefined`。
`String`	`Decimal`(使用雙精度)，引數的餘弦值。如果字串無法轉換為 `Decimal`，則結果為 `Undefined`。傳回的虛數結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

cosh(Decimal)

以弧度傳回數字的雙曲餘弦值。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`cosh(2.3)`= 5.037220649268761。

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的雙曲餘弦值。傳回的虛數結果為 `Undefined`。
`Decimal`	`Decimal`(使用雙精度)，引數的雙曲餘弦值。傳回的虛數結果為 `Undefined`。
`Boolean`	`Undefined`。
`String`	`Decimal`(使用雙精度)，引數的雙曲餘弦值。如果字串無法轉換為 `Decimal`，則結果為 `Undefined`。傳回的虛數結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

encode(value, encodingScheme)

根據編碼機制，使用 `encode`函數將承載 (可能並非 JSON 資料) 編碼為字串表現形式。受 SQL 版本 2016-03-23 和更新版本支援。

值

任何有效的表達式，如 AWS IoTSQL 參考 (p. 283)的定義。無論酬載是否為 JSON 格式，您都可以指定 * 以編碼整個酬載。若您提供了運算式，則在編碼之前，評估結果將轉換為字串。

encodingScheme

代表您想要使用的編碼機制的文字字串。目前僅支援 `'base64'`。

endswith(String, String)

傳回 `Boolean`，指出第一個 `String` 引數是否以第二個 `String` 引數結尾。如果引數為 `Null`或 `Undefined`，則結果為 `Undefined`。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`endswith("cat","at")`= true。

引數類型 1	引數類型 2	結果
`String`	`String`	如果第一個引數以第二 false。
其他值	其他值	所有引數都將使用標準 數以第二個引數結尾， 數為 `Null`或 `Undefi`

exp(Decimal)

傳回 e 次方的 `Decimal` 引數。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`exp(1)`= e。

引數類型	結果
`Int`	`Decimal`(使用雙精度)，e ^ 引數。
`Decimal`	`Decimal`(使用雙精度)，e ^ 引數。
`String`	`Decimal`(使用雙精度)，e ^ 引數。如果 `String` 無法轉換為 `Decimal`，則結果為 `Undefined`。
其他值	`Undefined`。

get

從集合類型 (陣列、字串、物件) 擷取值。第一個引數不會套用任何轉換。轉換會按表格中的記錄套用至第二個引數。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

`get(["a", "b", "c"], 1)` = "b"

`get({"a":"b"}, "a")` = "b"

`get("abc", 1)` = "b"

引數類型 1	引數類型 2	結果
Array	任何類型 (轉換為 `Int`)	第二個引數 (轉換為 A 的項目。如果轉換不成 引在 `Array`的邊界之外 `Undefined`。
字串	任何類型 (轉換為 `Int`)	第二個引數 (轉換為 I 字元。如果轉換不成功 引在字串的邊界之外 (`Undefined`。
物件	`String`(未套用轉換)	對應至第二個引數所指 的值。
其他值	任何值	`Undefined`。

get_dynamodb(tableName, partitionKeyName, partitionKeyValue, sortKeyName, sortKeyValue, roleArn)

從 DynamoDB 資料表擷取資料。`get_dynamodb()` 允許您在評估規則時查詢 DynamoDB 資料表。您可以使用從 DynamoDB 中擷取的資料來篩選或增強訊息承載。受 SQL 版本 2016-03-23 和更新版本支援。

`get_dynamodb()` 接受下列參數：

tableName

要查詢的 DynamoDB 資料表名稱。

partitionKeyName

分割區索引鍵的名稱。如需詳細資訊，請參閱 DynamoDB 金鑰。

partitionKeyValue

用來識別記錄的分割區索引鍵值。如需詳細資訊，請參閱 DynamoDB 金鑰。

sortKeyName

選用。排序索引鍵的名稱。只有在查詢的 DynamoDB 資料表使用複合索引鍵時，才需要此參數。如需詳細資訊，請參閱 DynamoDB 金鑰。

sortKeyValue

選用。排序索引鍵的值。只有在查詢的 DynamoDB 資料表使用複合索引鍵時，才需要此參數。如需詳細資訊，請參閱 DynamoDB 金鑰。

roleArn

授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。規則引擎會假設此角色代表您存取 DynamoDB 資料表。請避免使用過多許可的角色。僅授與角色規則所需的許可。以下是授與一個 DynamoDB 資料表存取權的範例政策。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            “Effect”: “Allow”,
            “Action”: “dynamodb:GetItem”,
            “Resource”: “arn:aws:dynamodb:<aws-region>:<account-id>:table/<table-name>”
        }
    ]
```

```
}}
```

舉例說明如何使用 `get_dynamodb()`，假設您有一個 DynamoDB 資料表，其中包含所有連接至 AWS IoT 之裝置的裝置 ID 和位置資訊。下列 SELECT 陳述式使用 `get_dynamodb()` 函數來擷取指定裝置 ID 的位置：

```
SELECT *, get_dynamodb("InServiceDevices", "deviceId", id,
"arn:aws:iam::12345678910:role/getdynamo").location AS location FROM 'some/
topic'
```

Note

- 每個 SQL 陳述式最多可以呼叫一次 `get_dynamodb()`。在單一 SQL 陳述式中多次呼叫 `get_dynamodb()` 會導致規則終止，而不會呼叫任何動作。
- 如果 `get_dynamodb()` 傳回的資料超過 8 KB，則可能不會叫用規則的動作。

get_thing_shadow(thingName, roleARN)

傳回指定物件的影子。受 SQL 版本 2016-03-23 和更新版本支援。

thingName

字串：想要擷取影子的物件名稱。

roleArn

字串：擁有 `iot:GetThingShadow`許可的角色 ARN。

範例：

```
SELECT * from 'a/b'

WHERE get_thing_shadow("MyThing","arn:aws:iam::123456789012:role/
AllowsThingShadowAccess") .state.reported.alarm = 'ON'
```

雜湊函數

AWS IoT 提供以下雜湊函數：

- md2
- md5
- sha1
- sha224
- sha256
- sha384
- sha512

所有雜湊函數都預期有一個字串引數。結果為該字串的雜湊值。標準字串轉換會套用至非字串的引數。SQL 版本 2015-10-8 和更新版本可支援所有雜湊函數。

範例：

`md2("hello")`= "a9046c73e00331af68917d3804f70655"

`md5("hello")`= "5d41402abc4b2a76b9719d911017c592"

indexof(String, String)

傳回第二個引數的第一個索引 (從 0 開始)，做為第一個引數的子字串。預期兩個引數均為字串。非字串的引數需遵守標準字串轉換規則。此函數不能套用在陣列上，僅可套用到字串上。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`indexof("abcd", "bc")` = 1

isNull()

如果引數是 `Null`值，則傳回 true受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`isNull(5)` = false。

`isNull(Null)` = true。

引數類型	結果
`Int`	false
`Decimal`	false
`Boolean`	false
`String`	false
`Array`	false
`Object`	false
`Null`	true
`Undefined`	false

isUndefined()

如果引數是 `Undefined`，則傳回 true。受 SQL 版本 2016-03-23 和更新版本支援。

範例：

`isUndefined(5)` = false。

`isUndefined(floor([1,2,3]))` = true。

引數類型	結果
`Int`	false
`Decimal`	false
`Boolean`	false
`String`	false
`Array`	false

引數類型	結果
`Object`	false
`Null`	false
`Undefined`	true

length(String)

傳回所提供的字串內的字元數。標準轉換規則會套用至非 `String`的引數。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`length("hi")`= 2

`length(false)`= 5

ln(Decimal)

傳回引數的自然對數。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`ln(e)`= 1。

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的自然對數。
`Decimal`	`Decimal`(使用雙精度)，引數的自然對數。
`Boolean`	`Undefined`。
`String`	`Decimal`(使用雙精度)，引數的自然對數。如果字串無法轉換為 `Decimal`，則結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

log(Decimal)

傳回引數以10 為底的對數。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`log(100)`= 2.0。

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數以 10 為底的對數。
`Decimal`	`Decimal`(使用雙精度)，引數以 10 為底的對數。

引數類型	結果
Boolean	Undefined。
String	Decimal(使用雙精度)，引數以 10 為底的對數。如果 String 無法轉換為 Decimal，則結果為 Undefined。
Array	Undefined。
物件	Undefined。
Null	Undefined。
未定義	Undefined。

lower(String)

傳回小寫版本的特定 String。使用標準轉換規則將非字串引數轉換為字串。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

lower("HELLO") =「hello」。

lower(["HELLO"]) = "[\"hello\"]".

lpad(String, Int)

傳回 String引數，左側填入第二個引數所指定的空格數。Int引數必須介於 0 到 1000 之間。如果提供的值超出此有效範圍，則引數將設為最接近的有效值 (0 或 1000)。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

lpad("hello", 2) =「hello」。

lpad(1, 3)= "1"

引數類型 1	引數類型 2	結果
String	Int	String，所給的 Str 的空格。
String	Decimal	Decimal 引數將無條 會在左側填充指定的空
String	String	第二個引數會轉換為 I Int，而 String 會在 引數無法轉換為 Int，
其他值	Int/Decimal/String	第一個值會使用標準轉 數會套用至該 String Undefined。
任何值	其他值	Undefined。

ltrim(String)

移除所給的 String前方所有空格 (tab 與空格)。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`Ltrim(" h i ")`= "hi "。

引數類型	結果
`Int`	`String`移除前方所有空格的 `Int` 顯示方式。
`Decimal`	`String`移除前方所有空格的 `Decimal` 顯示方式。
`Boolean`	布林值 (「true」或「false」) 移除前方所有空格的 `String`顯示方式。
`String`	移除前方所有空格的引數。
Array	`String`(使用標準轉換規則) 移除前方所有空格的 `Array` 顯示方式。
物件	物件 (使用標準轉換規則) 移除前方所有空格的 `String`顯示方式。
Null	`Undefined`。
未定義	`Undefined`。

machinelearning_predict(modelId)

使用 `machinelearning_predict` 函數，運用來自 MQTT 訊息的資料，根據 Amazon Machine Learning (Amazon ML) 模型進行預測。受 SQL 版本 2015-10-8 和更新版本支援。`machinelearning_predict`函數的引數如下：

modelId

要對其執行預測的模型 ID。必須啟用該模型的即時端點。

roleArn

IAM 角色具有擁有 `machinelearning:Predict` 和 `machinelearning:GetMLModel` 許可的政策，且允許存取要對其執行預測的模型。

record

資料會傳送到 Amazon ML 預測 API。此應顯示為單層 JSON 物件。若該記錄為多層級 JSON 物件，則記錄會透過值的序列化來扁平化。例如，以下 JSON：

```
{ "key1": {"innerKey1": "value1"}, "key2": 0}
```

會變成：

```
{ "key1": "{\"innerKey1\": \"value1\"}", "key2": 0}
```

該函數會傳回具有以下欄位的 JSON 物件：

predictedLabel

根據模型的輸入分類。

details

包含下列屬性：

PredictiveModelType

模型類型。有效值為 REGRESSION、BINARY、MULTICLASS。

演算法

Amazon ML 進行預測所使用的演算法。該值必須為 SGD。

predictedScores

包含對應至每個標籤的原始分類分數。

predictedValue

Amazon ML 預測的值。

mod(Decimal, Decimal)

傳回第一個引數除以第二的引數的餘數。受 SQL 版本 2015-10-8 和更新版本支援。也可以使用「%」當做同樣的模除功能的 infix 運算子。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`mod(8, 3)`= 2。

左運算元	右運算元	輸出
`Int`	`Int`	`Int`，第一個引數模除
`Int/Decimal`	`Int/Decimal`	`Decimal`，第一個引數
`String/Int/Decimal`	`String/Int/Decimal`	如果所有的字串都轉換個引數為模。否則會傳
其他值	其他值	`Undefined`。

nanvl(AnyValue, AnyValue)

若第一個引數為有效的 `Decimal`，即傳回。否則便傳回第二個引數。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`Nanvl(8, 3)`= 8。

引數類型 1	引數類型 2	輸出
未定義	任何值	第二個參數。
Null	任何值	第二個參數。
`Decimal`(NaN)	任何值	第二個參數。
`Decimal`(非 NaN)	任何值	第一個參數。
其他值	任何值	第一個參數。

newuuid()

傳回隨機的 16 位元組 UUID。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`newuuid()` = `123a4567-b89c-12d3-e456-789012345000`

numbytes(String)

傳回所給字串 UTF-8 編碼中的位元組數。標準轉換規則會套用至非 `String`的引數。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`numbytes("hi")`= 2

`numbytes("€") `= 3

principal()

傳回裝置用於身分驗證的委託人。傳回的委託人取決於發佈觸發訊息的方式。下表說明為各發佈方法和通訊協定傳回的委託人。

訊息發佈方式	通訊協定	登入資料類型
MQTT 用戶端	MQTT	X.509 裝置憑證
AWS IoT 主控台 MQTT 用戶端	MQTT	IAM 使用者或角色
AWS CLI	HTTP	IAM 使用者或角色
AWS IoT裝置 SDK	MQTT	X.509 裝置憑證
AWS IoT裝置 SDK	經 WebSocket 的 MQTT	IAM 使用者或角色

以下範例顯示 `principal` 所傳回值的不同類型：

- X.509 憑證指紋：`ba67293af50bf2506f5f93469686da660c7c844e7b3950bfb16813e0d31e9373`
- IAM 角色 ID 和工作階段名稱：`ABCD1EFG3HIJK2LMNOP5:my-session-name`
- 傳回使用者 ID：`ABCD1EFG3HIJK2LMNOP5`

parse_time(String, Long, [String])

使用 `parse_time`函數來設定時間戳記格式，變成人類易懂的日期/時間格式。受 SQL 版本 2016-03-23 和更新版本支援。`parse_time`函數的引數如下：

pattern

(字串) 符合 ISO 8601 標準格式的日期/時間。(該函數特別支援 Joda-Time 格式。)

timestamp

(Long) 自 Unix epoch 起以毫秒單位設定格式的時間。請參閱函數 timestamp() (p. 321)。

timezone

(字串) 選用。格式化日期/時間的時區。預設值為「UTC」。該函數支援 Joda-Time 時區。

例如：

在此訊息發佈至主題「A/B」時，酬載 `{"ts": "1970.01.01 AD at 21:46:40 CST"}` 會傳送至 S3 儲存貯體：

```
{
```

```
    "ruleArn": "arn:aws:iot:us-east-2:ACCOUNT_ID:rule/RULE_NAME",
    "topicRulePayload": {
        "sql": "SELECT parse_time("yyyy.MM.dd G 'at' HH:mm:ss z", 100000000, "America/
Belize" ) as ts FROM 'A/B'",

        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23",
        "actions": [
            {
                "s3": {
                    "roleArn": "arn:aws:iam::ACCOUNT_ID:rule:role/ROLE_NAME",
                    "bucketName": "BUCKET_NAME",
                    "key": "KEY_NAME"
                }
            }
        ],
        "ruleName": "RULE_NAME"
    }
}
```

在此訊息發佈至主題「A/B」時，與 `{"ts": "2017.06.09 AD at 17:19:46 UTC"}` 相似，但包含目前日期/時間的酬載會傳送至 S3 儲存貯體：

```
{
    "ruleArn": "arn:aws:iot:us-east-2:ACCOUNT_ID:rule/RULE_NAME",
    "topicRulePayload": {
        "sql": "SELECT parse_time("yyyy.MM.dd G 'at' HH:mm:ss z", timestamp() ) as ts FROM
 'A/B'",
        "awsIotSqlVersion": "2016-03-23",
        "ruleDisabled": false,
        "actions": [
            {
                "s3": {
                    "roleArn": "arn:aws:iam::ACCOUNT_ID:rule:role/ROLE_NAME",
                    "bucketName": "BUCKET_NAME",
                    "key": "KEY_NAME"
                }
            }
        ],
        "ruleName": "RULE_NAME"
    }
}
```

`parse_time()`也可以用來做為替代範本。例如，當此訊息發佈至主題「A/B」時，該酬載會傳送至金鑰 =「2017」的 S3 儲存貯體：

```
{
    "ruleArn": "arn:aws:iot:us-east-2:ACCOUNT_ID:rule/RULE_NAME",
    "topicRulePayload": {
        "sql": "SELECT * FROM 'A/B'",
        "awsIotSqlVersion": "2016-03-23",
        "ruleDisabled": false,
        "actions": [
            {
                "s3": {
                    "roleArn": "arn:aws:iam::ACCOUNT_ID:rule:role/ROLE_NAME",
                    "bucketName": BUCKET_NAME,
                    "key": "${parse_time("yyyy", timestamp(), "UTC")}"
                }
            }
        ],
        "ruleName": "RULE_NAME"
    }
```

```
}
```

power(Decimal, Decimal)

傳回第一個引數次方的第二個引數值。`Decimal` 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`power(2, 5)`= 32.0。

引數類型 1	引數類型 2	輸出
`Int/Decimal`	`Int/Decimal`	`Decimal`(使用雙精度
`Int/Decimal/String`	`Int/Decimal/String`	`Decimal`(使用雙精度 任何轉換為小數的字串 `Decimal`，則結果為
其他值	其他值	`Undefined`。

rand()

傳回虛擬亂數，平均分散在 0.0 和 1.0。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`rand()` = 0.8231909191640703

regexp_matches(String, String)

如果字串 (第一個引數) 包含規則表達式的相符項目 (第二個引數)，則傳回 true。

範例：

`Regexp_matches("aaaa", "a{2,}")` = true。

`Regexp_matches("aaaa", "b")`= false。

第一個引數：

引數類型	結果
`Int`	`String`的 `Int` 顯示方式。
`Decimal`	`String`的 `Decimal` 顯示方式。
`Boolean`	布林值 (「true」或「false」) 的 `String`顯示方式。
`String`	`String`。
Array	`String`(使用標準轉換規則) 的 `Array` 顯示方式。
物件	使用標準轉換規則之物件的 `String`顯示方式。
Null	`Undefined`。
未定義	`Undefined`。

第二個引數：

必須為有效的 regex 表達式。非字串類型都會使用標準轉換規則轉換為 `String`。根據類型，結果字串可能不是有效的正規運算式。如果 (轉換的) 引數並非有效的 regex 值，結果會為 `Undefined`。

第三個引數：

必須為有效的 regex 替換字串。(可以參考擷取群組)。 非字串類型都會使用標準轉換規則轉換為 `String`。如果 (轉換的) 引數並非有效的 regex 替換字串，則結果為 `Undefined`。

regexp_replace(String, String, String)

以第三個引數取代所有在第一個引數中出現的第二個參數 (一般表達式)。請以「$」參考擷取群組。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`Regexp_replace("abcd", "bc", "x")`= "axd"。

`Regexp_replace("abcd", "b(.*)d", "$1")`= "ac"。

第一個引數：

引數類型	結果
`Int`	`String`的 `Int` 顯示方式。
`Decimal`	`String`的 `Decimal` 顯示方式。
`Boolean`	布林值 (「true」或「false」) 的 `String`顯示方式。
`String`	來源值。
Array	`String`(使用標準轉換規則) 的 `Array` 顯示方式。
物件	使用標準轉換規則之物件的 `String`顯示方式。
Null	`Undefined`。
未定義	`Undefined`。

第二個引數：

必須為有效的 regex 表達式。非字串類型都會使用標準轉換規則轉換為 `String`。根據類型，結果字串可能不是有效的正規運算式。如果 (轉換的) 引數並非有效的 regex 表達式，則結果為 `Undefined`。

第三個引數：

必須為有效的 regex 替換字串。(可以參考擷取群組)。 非字串類型都會使用標準轉換規則轉換為 `String`。如果 (轉換的) 引數並非有效的 regex 替換字串，則結果為 `Undefined`。

regexp_substr(String, String)

在第一個參數中尋找第一個符合第二個參數 (regex) 的值。請以「$」參考擷取群組。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

`regexp_substr("hihihello", "hi")`"hi"

`regexp_substr("hihihello", "(hi)*")`。"hihi"

第一個引數：

引數類型	結果
Int	String的 Int 顯示方式。
Decimal	String的 Decimal 顯示方式。
Boolean	布林值 (「true」或「false」) 的 String顯示方式。
String	String引數。
Array	String(使用標準轉換規則) 的 Array 顯示方式。
物件	使用標準轉換規則之物件的 String顯示方式。
Null	Undefined。
未定義	Undefined。

第二個引數：

必須為有效的 regex 表達式。非字串類型都會使用標準轉換規則轉換為 String。根據類型，結果字串可能不是有效的正規運算式。如果 (轉換的) 引數並非有效的 regex 表達式，則結果為 Undefined。

第三個引數：

必須為有效的 regex 替換字串。(可以參考擷取群組)。 非字串類型都會使用標準轉換規則轉換為 String。如果引數並非有效的 regex 替換字串，則結果為 Undefined。

rpad(String, Int)

傳回字串引數，右側填入第二個引數所指定的空格數。Int引數必須介於 0 到 1000 之間。如果提供的值超出此有效範圍，則引數將設為最接近的有效值 (0 或 1000)。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

rpad("hello", 2) =「hello」。

rpad(1, 3)= "1"。

引數類型 1	引數類型 2	結果
String	Int	String右側填入的空格數量等同所給的 Int。
String	Decimal	Decimal引數會無條件捨去至最接近的 Int，

引數類型 1	引數類型 2	結果
		並且該字串將以提供的 `Int` 相同的空格數填入右側。
`String`	`String`	第二個引數將轉換為 `Decimal`，並無條件捨去至最接近的 `Int`。`String`右側填入的空格數量等同 `Int` 的值。
其他值	`Int`/`Decimal`/`String`	第一個值會使用標準轉換來轉換為 `String`，而 rpad 函數將套用到該 `String` 上。如果其無法轉換，則結果為 `Undefined`。
任何值	其他值	`Undefined`。

round(Decimal)

將指定的 `Decimal`無條件進位至最接近的 `Int`。如果 `Decimal`與兩個 `Int` 值 (例如，0.5) 等距，則 `Decimal` 會無條件進位。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`Round(1.2)`= 1。

`Round(1.5)` = 2。

`Round(1.7)` = 2。

`Round(-1.1)` = -1。

`Round(-1.5)` = -2。

引數類型	結果
`Int`	引數。
`Decimal`	`Decimal`是要向下捨入到最接近的 `Int`。
`String`	`Decimal`是要向下捨入到最接近的 `Int`。如果字串無法轉換為 `Decimal`，則結果為 `Undefined`。
其他值	`Undefined`。

rtrim(String)

移除所給的 `String`後方所有空格 (tab 與空格)。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`rtrim(" h i ")`= " h i"

引數類型	結果
`Int`	`String`的 `Int` 顯示方式。
`Decimal`	`String`的 `Decimal` 顯示方式。
`Boolean`	布林值 (「true」或「false」) 的 `String`顯示方式。
Array	`String`(使用標準轉換規則) 的 `Array` 顯示方式。
物件	使用標準轉換規則之物件的 `String`顯示方式。
Null	`Undefined`。
未定義	`Undefined`

sign(Decimal)

傳回所給數字的符號。當引數的符號為正值時，傳回 1。當引數的符號為負值時，傳回 -1。如果引數為 0，傳回 0。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`sign(-7)` = -1。

`sign(0)` = 0。

`sign(13)` = 1。

引數類型	結果
`Int`	`Int`，`Int` 值的符號。

引數類型	結果
Decimal	Int，Decimal 值的符號。
String	Int，Decimal 值的符號。此字串會轉換為 Decimal值，並傳回 Decimal 值的符號。如果 String無法轉換為 Decimal，則結果為 Undefined。受 SQL 版本 2015-10-8 和更新版本支援。
其他值	Undefined。

sin(Decimal)

以弧度傳回數字的正弦值。Decimal 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：sin(0)= 0.0

引數類型	結果
Int	Decimal(使用雙精度)，引數的正弦值。
Decimal	Decimal(使用雙精度)，引數的正弦值。
Boolean	Undefined。
String	Decimal(使用雙精度)，引數的正弦值。如果字串無法轉換為 Decimal，則結果為 Undefined。
Array	Undefined。
物件	Undefined。
Null	Undefined。
Undefined	Undefined。

sinh(Decimal)

以弧度傳回數字的雙曲正弦值。Decimal 值在套用函數前會四捨五入至雙精度。結果為雙精度的 Decimal值。受 SQL 版本 2015-10-8 和更新版本支援。

範例：sinh(2.3)= 4.936961805545957

引數類型	結果
Int	Decimal(使用雙精度)，引數的雙曲正弦值。
Decimal	Decimal(使用雙精度)，引數的雙曲正弦值。
Boolean	Undefined。
String	Decimal(使用雙精度)，引數的雙曲正弦值。如果字串無法轉換為 Decimal，則結果為 Undefined。
Array	Undefined。

引數類型	結果
物件	Undefined。
Null	Undefined。
未定義	Undefined。

substring(String, Int [, Int])

預期 String後有一或兩個 Int 值。若為 String和單個 Int 引數，此函數會從所給的 String 索引 (從 0 開始，包含 0) 到 Int 的結尾傳回所給的 String 的子字串。若為 String和兩個 Int 引數，該函數會傳回從第一個 String 索引引數 (從 0 開始，包含 0) 到第二個 Int 索引引數 (從 0 開始，不含 0) 所提供的所給 Int 的子字串。少於零的索引將設為零。比 String 長度還長的索引會設為 String 長度。至於第三個引數版本，如果第一個索引大於 (或等於) 第二個索引，則結果為空的 String。

如果提供的引數不是 (##、##) 或 (##、##、##)，則標準轉換會套用至該引數，以嘗試將那些引數轉換為正確類型。如果類型無法轉換，函數的結果即為 Undefined。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

substring("012345", 0)= "012345"。

substring("012345", 2)= "2345"。

substring("012345", 2.745)= "2345"。

substring(123, 2)= "3"。

substring("012345", -1)= "012345"。

substring(true, 1.2)= "rue"。

substring(false, -2.411E247)= "false"。

substring("012345", 1, 3)= "12"。

substring("012345", -50, 50)= "012345"。

substring("012345", 3, 1)= ""。

sqrt(Decimal)

傳回數字的平方根。Decimal 引數在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：sqrt(9)= 3.0。

引數類型	結果
Int	引數的平方根。
Decimal	引數的平方根。
Boolean	Undefined。

引數類型	結果
String	引數的平方根。如果字串無法轉換為 Decimal，則結果為 Undefined。
Array	Undefined。
物件	Undefined。
Null	Undefined。
未定義	Undefined。

startswith(String, String)

傳回 Boolean，無論第一個字串引數的開頭是否為第二個引數。如果引數為 Null或 Undefined，則結果為 Undefined。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

startswith("ranger","ran")= true

引數類型 1	引數類型 2	結果
String	String	無論第一個字串的開頭
其他值	其他值	所有引數都將使用標準 串的開頭是第二個字串 Undefined，則結果

tan(Decimal)

以弧度傳回數字的正切值。Decimal 值在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：tan(3)= -0.1425465430742778

引數類型	結果
Int	Decimal(使用雙精度)，引數的正切值。
Decimal	Decimal(使用雙精度)，引數的正切值。
Boolean	Undefined。
String	Decimal(使用雙精度)，引數的正切值。如果字串無法轉換為 Decimal，則結果為 Undefined。
Array	Undefined。
物件	Undefined。
Null	Undefined。
未定義	Undefined。

tanh(Decimal)

以弧度傳回數字的雙曲正切值。`Decimal` 值在套用函數前會四捨五入至雙精度。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`tanh(2.3)`= 0.9800963962661914

引數類型	結果
`Int`	`Decimal`(使用雙精度)，引數的雙曲正切值。
`Decimal`	`Decimal`(使用雙精度)，引數的雙曲正切值。
`Boolean`	`Undefined`。
`String`	`Decimal`(使用雙精度)，引數的雙曲正切值。如果字串無法轉換為 `Decimal`，則結果為 `Undefined`。
Array	`Undefined`。
物件	`Undefined`。
Null	`Undefined`。
未定義	`Undefined`。

timestamp()

以毫秒為單位，傳回自國際標準時間 (UTC) 1970 年 1 月 1 日星期四 00:00:00 起的時間戳記，如 AWS IoT 規則引擎的觀察。受 SQL 版本 2015-10-8 和更新版本支援。

範例：`timestamp()` = `1481825251155`

topic(Decimal)

傳回觸發該規則的訊息要傳送至的主題。如果未指定參數，則傳回整個主題。`Decimal` 參數用於指定特定主題區段，並會使用 1 指定第一個區段。對於 topic `foo/bar/baz`，topic(1) 會傳回 `foo`，topic(2) 會傳回 `bar`，依此類推。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

`topic()`= "things/myThings/thingOne"

`topic(1)`= "things"

使用 基本擷取 (p. 331) 時，topic() 函數無法使用主題的初始字首 (`$aws/rules/`*`rule-name`*)。例如，假定主題為：

`$aws/rules/BuildingManager/Buildings/Building5/Floor2/Room201/Lights`

`topic()` = "Buildings/Building5/Floor2/Room201/Lights"

`topic(3)` = "Floor2"

traceid()

傳回 MQTT 的追蹤 ID (UUID)，如果訊息不是透過 MQTT 傳送，則為 `Undefined`。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`traceid()` = "12345678-1234-1234-1234-123456789012"

trunc(Decimal, Int)

將第一個引數截為第二的引數所指定的 `Decimal`位數。若第二個引數少於零，則它會設為零。若第二個引數大於 34，則它會設為 34。結果的結尾去掉了零。受 SQL 版本 2015-10-8 和更新版本支援。

例如：

`trunc(2.3, 0)` = 2。

`trunc(2.3123, 2)`= 2.31。

`trunc(2.888, 2)` = 2.88。

`trunc(2.00, 5)`= 2。

引數類型 1	引數類型 2	結果
`Int`	`Int`	來源值。
`Int/Decimal`	`Int/Decimal`	第一個引數會截短為第 如果不是 `Int`，則會轉
`Int/Decimal/String`	`Int/Decimal`	第一個引數會截短為第 如果不是 `Int`，則會轉 會轉換為 `Decimal` 值 `Undefined`。
其他值		`Undefined`。

trim(String)

移除所給的 `String`後方所有空格。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`Trim(" hi ")` = "hi"

引數類型	結果
`Int`	`String`移除前後方所有空格的 `Int` 顯示方式。
`Decimal`	`String`移除前後方所有空格的 `Decimal` 顯示方式。
`Boolean`	`String`(「true」或「false」) 移除前後方所有空格的 `Boolean` 顯示方式。
`String`	`String`移除了前後方的所有空格。
Array	`String`(使用標準轉換規則) 的 `Array` 顯示方式。
物件	使用標準轉換規則之物件的 `String`顯示方式。
Null	`Undefined`。

引數類型	結果
未定義	`Undefined`。

upper(String)

傳回大寫版本的特定 `String`。非 `String`引數都會使用標準轉換規則轉換為 `String`。受 SQL 版本 2015-10-8 和更新版本支援。

範例：

`upper("hello")`= "HELLO"

`upper(["hello"])`= "[\"HELLO\"]"

SELECT 子句

AWS IoT SELECT 子句基本上與 ANSI SQL SELECT 子句是大同小異。

您可以使用 SELECT 子句來擷取傳入的 MQTT 訊息內的資訊。`SELECT *` 可用於擷取整個傳入訊息的承載。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL statement: SELECT * FROM 'a/b'
Outgoing payload: {"color":"red", "temperature":50}
```

如果承載為 JSON 物件，您可以參考物件中的鍵。您的傳出酬載會包含鍵值組。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL statement: SELECT color FROM 'a/b'
Outgoing payload: {"color":"red"}
```

您可以使用 AS 關鍵字重新命名索引鍵。例如：

```
Incoming payload published on topic 'a/b':{"color":"red", "temperature":50}
SQL:SELECT color AS my_color FROM 'a/b'
Outgoing payload: {"my_color":"red"}
```

您可以選擇多個項目，以逗號分隔即可。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL: SELECT color as my_color, temperature as farenheit FROM 'a/b'
Outgoing payload: {"my_color":"red","farenheit":50}
```

您可以選擇多個項目，包括以「*」新增項目至來年的承載。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL: SELECT *, 15 as speed FROM 'a/b'
Outgoing payload: {"color":"red", "temperature":50, "speed":15}
```

您可以使用 `"VALUE"`關鍵字產生傳出的承載不 JSON 物件。您可以只選擇一個項目。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL: SELECT VALUE color FROM 'a/b'
```

```
Outgoing payload: "red"
```

您可以使用 `'.'`語法來深入巢狀 JSON 物件傳入的承載。例如：

```
Incoming payload published on topic 'a/b': {"color":{"red":255,"green":0,"blue":0},
 "temperature":50}
SQL: SELECT color.red as red_value FROM 'a/b'
Outgoing payload: {"red_value":255}
```

您可以使用函數 (請參閱 函數 (p. 292))，將傳入的承載。括弧可用於分組。例如：

```
Incoming payload published on topic 'a/b': {"color":"red", "temperature":50}
SQL: SELECT (temperature - 32) * 5 / 9 AS celsius, upper(color) as my_color FROM 'a/b'
Outgoing payload: {"celsius":10,"my_color":"RED"}
```

使用二進位 Payloads

當訊息承載應該處理原始二進位資料 (而不是 JSON 物件)，您可以使用 * 營運，請參閱以 `SELECT`子句。

這些規則必須依照使用 *，請參閱訊息承載做為原始二進位資料：

1. SQL 陳述式和範本不得參考 * 之外的 JSON 名稱。
2. The SELECT 陳述式必須擁有 * 是唯一可歸為領導者的項目，或者必須擁有唯一的功能，例如：

```
SELECT * FROM 'a/b'
```

```
SELECT encode(*, 'base64') AS data, timestamp() AS ts FROM 'a/b'
```

二進位承載範例

您可以使用以下 `SELECT`陳述式使用二進位承載，因為它不參考任何 JSON 名稱。

```
SELECT * FROM 'a/b'
```

以下 `SELECT` 不得和二進位酬載搭配使用，因為它是引用 WHERE 子句中的 `device_type`。

```
SELECT * FROM 'a/b' WHERE device_type = 'thermostat'
```

以下 `SELECT` 不得和二進位酬載搭配使用，因為這違反了規則 2。

```
SELECT *, timestamp() AS timestamp FROM 'a/b'
```

以下 `SELECT` 可以和二進位酬載搭配使用，因為它不違反規則 1 或 2。

```
SELECT * FROM 'a/b' WHERE timestamp() % 12 = 0
```

以下 AWS IoT 規則不得和二進位酬載搭配使用，因為這違反了規則 1。

```
{
    "sql": "SELECT * FROM 'a/b'"
    "actions": [{
        "republish": {
```

```
            "topic":"device/${device_id}"
        }
    }]
}
```

FROM 子句

FROM 子句可讓主題或主題篩選器訂閱規則。此主題或主題篩選必須括在單引號 (') 內。系統會為每則傳送到符合此處指定的主題篩選條件之 MQTT 主題的訊息觸發該規則。主題篩選器可訂閱整群類似的主題。

範例：

發佈在主題 `'a/b'`的傳入承載：`{temperature: 50}`

發佈在主題 `'a/c'`的傳入承載：`{temperature: 50}`

SQL：`"SELECT temperature AS t FROM 'a/b'"`。

`'a/b'`已訂閱該規則，因此傳入的負載會傳送給該規則，而傳出承載 (傳送至規則動作) 為：`{t: 50}`。規則並未給 `'a/c'`訂閱，因此發佈在 `'a/c'` 的訊息不會觸發該規則。

萬用字元範例：

您可以使用 '#'(多層級) 萬用字元，以比對一或多個特定路徑元素：

發佈在主題 `'a/b'`的傳入承載：`{temperature: 50}`。

發佈在主題 `'a/c'`的傳入承載：`{temperature: 60}`。

發佈在主題 `'a/e/f'`的傳入承載：`{temperature: 70}`。

發佈在主題 `'b/x'`的傳入承載：`{temperature: 80}`。

SQL：`"SELECT temperature AS t FROM 'a/#'"`。

已將該規則訂閱至開頭為 `'a'` 的任何主題，因此該規則會執行三次，將傳出的 `{t: 50}` (為 a/b)、`{t: 60}` (為 a/c) 和 `{t: 70}` (為 a/e/f) 的承載傳送給其動作。它沒有訂閱 `'b/x'`，因此不會觸發 `{temperature: 80}` 訊息的規則。

+ 萬用字元範例：

您可以使用 '+'(單層級) 萬用字元，以比對任何一個特定路徑元素：

發佈在主題 `'a/b'`的傳入承載：`{temperature: 50}`。

發佈在主題 `'a/c'`的傳入承載：`{temperature: 60}`。

發佈在主題 `'a/e/f'`的傳入承載：`{temperature: 70}`。

發佈在主題 `'b/x'`的傳入承載：`{temperature: 80}`。

SQL：`"SELECT temperature AS t FROM 'a/+'"`。

所有主題訂閱的規則有兩個路徑元素，第一個為：`'a'`。規則會因傳送至 `'a/b'`和 `'a/c'` 的訊息而執行，但 `'a/e/f'` 或 `'b/x'` 不會。

WHERE 子句

WHERE 子句會判斷是否執行某規則指定的動作。如果 WHERE 子句判斷值為 true，則規則動作已執行。否則，規則動作就不會執行。

例如：

發佈在 a/b的傳入承載：`{"color":"red", "temperature":40}`。

SQL：`SELECT color AS my_color FROM 'a/b' WHERE temperature > 50 AND color <> 'red'`。

在此情況下，該規則會觸發，但由規則指定的動作則不會執行。不會出現傳出酬載。

您可以在 WHERE 子句中使用函數和運算子。不過您無法參考任何在 SELECT 中以 AS 關鍵字建立的別名。(WHERE 子句會優先進行評估，以判定 SELECT 是否受到評估。)

文字

可以直接在規則 SQL 的 SELECT 和 WHERE 子句中指定文字物件，很適合用於傳遞資訊。

> **Note**
>
> 只有在使用 2016-03-23 版本或更高版本的 SQL 時才能使用文字。

會使用 JSON 物件語法 (金鑰/值對，以逗號分隔，索引鍵為字串而值為 JSON 值，以大括號 {} 括起)。例如：

發佈在主題 a/b的傳入承載：`{"lat_long": [47.606,-122.332]}`

SQL 陳述式：`SELECT {'latitude': get(lat_long, 0),'longitude':get(lat_long, 1)} as lat_long FROM 'a/b'`

產生的傳出承載為：`{"lat_long":{"latitude":47.606,"longitude":-122.332}}`。

您也可以直接在規則 SQL 的 SELECT 和 WHERE 子句中指定陣列，即可將資訊分組。會使用 JSON 語法 (將逗號分隔項目用方括號 [] 括起，以建立陣列文字)。例如：

發佈在主題 a/b的傳入承載：`{"lat": 47.696, "long": -122.332}`

SQL 陳述式：`SELECT [lat,long] as lat_long FROM 'a/b'`

產生的輸出承載為：`{"lat_long": [47.606,-122.332]}`。

案例陳述式

案例陳述式可用於分支執行，例如切換陳述式或 if/else 陳述式。

語法：

```
CASE v WHEN t[1] THEN r[1]
  WHEN t[2] THEN r[2] ...
  WHEN t[n] THEN r[n]
  ELSE r[e] END
```

表達式 v會受到評估並對每個 t[i] 表達式配對是否相等。如果找到匹配的結果，相對應的 r[i]表達式會成為該案例陳述式的結果。如果可能的配對超過一個以上，則會選擇地一個配對。如果沒有相符的配對，則會將 else 陳述式的 re當成結果。如果沒有相符的配對，也沒有 else 陳述式，則案例陳述式的結果為 Undefined。例如：

發佈在主題 a/b的傳入承載：`{"color":"yellow"}`

SQL 陳述式：`SELECT CASE color WHEN 'green' THEN 'go' WHEN 'yellow' THEN 'caution' WHEN 'red' THEN 'stop' ELSE 'you are not at a stop light' END as instructions FROM 'a/b'`

產生的輸出承載為：`{"instructions":"caution"}`。

案例陳述式會要求至少一個 WHEN 子句。不需要 ELSE 子句。

> **Note**
>
> 如果 v 是 `Undefined`，則案例陳述式的結果為 `Undefined`。

JSON Extensions

您可以使用下列 ANSI SQL 的延伸模組，更輕鬆地使用巢狀 JSON 物件。

"." 運算子

運算子會存取嵌入式 JSON 物件的成員，功能與 ANSI SQL 和 JavaScript 完全相同。例如：

```
SELECT foo.bar AS bar.baz FROM 'a/b'
```

* 運算子

運作方式與 ANSI SQL 的 *萬用字元相同。此僅會使用在 SELECT 子句，且會建立包含訊息資料的新 JSON 物件。如果訊息承載並非 JSON 格式，*會以原始位元組的形式傳回整個訊息承載。例如：

```
SELECT * FROM 'a/b'
```

將函數套用在屬性值上

以下為可能為由裝置發佈的範例 JSON 酬載：

```
{
    "deviceid" : "iot123",
    "temp" : 54.98,
    "humidity" : 32.43,
    "coords" : {
        "latitude" : 47.615694,
        "longitude" : -122.3359976
    }
}
```

以下範例會將函數套用在 JSON 承載內的屬性值上：

```
SELECT temp, md5(deviceid) AS hashed_id FROM topic/#
```

此次查詢的結果為下列 JSON 物件：

```
{
   "temp": 54.98,
   "hashed_id": "e37f81fb397e595c4aeb5645b8cbbbd1"
}
```

替代範本

您可以使用替代範本，來強化在觸發規則和 AWS IoT 執行動作時，所傳回的 JSON 資料。替代範本的語法為 `${表達式}`，表達式可以是任何 AWS IoT 在 SELECT 或 WHERE 子句中所支援的任何表達式。這包含了在原始訊息中呈現的函數、運算子和資訊。因為替換範本中的運算式與「SELECT ...」陳述式是分開

計算的，所以不能參考使用 AS 子句建立的別名。如需支援的表達式的詳細資訊，請參閱 AWS IoTSQL 參考 (p. 283)。

替代範本顯示在規則中的 SELECT 子句：

```
{
    "sql": "SELECT *, topic() AS topic FROM 'my/iot/topic'",
    "ruleDisabled": false,
    "actions": [{
        "republish": {
            "topic": "${topic()}/republish",
            "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
        }
    }]
}
```

如果這個規則是由下列 JSON 所觸發：

```
{
    "deviceid" : "iot123",
    "temp" : 54.98,
    "humidity" : 32.43,
    "coords" : {
        "latitude" : 47.615694,
        "longitude" : -122.3359976
    }
}
```

以下是該規則的輸出：

```
{
    "coords":{
        "longitude":-122.3359976,
        "latitude":47.615694
    },
    "humidity":32.43,
    "temp":54.98,
    "deviceid":"iot123",
    "topic":"my/iot/topic"
}
```

SQL 版本

AWS IoT 規則引擎使用類似 SQL 的語法來挑選來自 MQTT 訊息的資料。SQL 陳述式會以指定的 SQL 版本來解釋，版本由描述規則的 JSON 文件的 `awsIotSqlVersion`屬性指定。如需更多關於 JSON 規則文件結構的資訊，請參閱建立規則的相關文章 (p. 256)。`awsIotSqlVersion` 屬性允許您指定要使用哪一個版本的 AWS IoT SQL 規則引擎。部署新版本時，可以繼續使用舊版本，或變更規則以使用新版本。目前的規則會繼續使用建立時使用的版本。

下列 JSON 範例顯示了如何使用 `awsIotSqlVersion`屬性指定 SQL 版本：

```
{
  "sql": "expression",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [{
      "republish": {
          "topic": "my-mqtt-topic",
```

```
            "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
        }
    }]
}
```

目前支援的版本：

- `2015-10-08`，建立於 2015 年 10 月 8 日的原始 SQL 版本。
- `2016-03-23`，建立於 2016 年 3 月 23 日的 SQL 版本。
- `beta`，最新的 SQL 版本。使用此版本可能導致規則受到破壞性的變更。

2016 年 3 月 23 日 SQL 規則引擎版本的新功能

- 修正巢狀 JSON 物件的選取問題。
- 修正陣列查詢的問題。
- 支援內部物件查詢。
- 支援將陣列輸出為最上層物件。
- 增加的編碼 (value、encodingScheme) 函數，可套用於 JSON 和非 JSON 格式的資料。

內部物件查詢

此功能可讓您查詢 JSON 物件的屬性。舉例而言，若為以下的 MQTT 訊息：

```
{
    "e": [
       { "n": "temperature", "u": "Cel", "t": 1234, "v":22.5 },
       { "n": "light", "u": "lm", "t": 1235, "v":135 },
       { "n": "acidity", "u": "pH", "t": 1235, "v":7 }
    ]
}
```

此外以下規則：

```
SELECT (SELECT v FROM e WHERE n = 'temperature') as temperature FROM 'my/topic'
```

規則會產生以下輸出：

```
{"temperature": [{"v":22.5}]}
```

若使用相同的 MQTT 訊息，而指定更加複雜的規則，例如：

```
SELECT get((SELECT v FROM e WHERE n = 'temperature'),1).v as temperature FROM 'topic'
```

規則會產生以下輸出：

```
{"temperature":22.5}
```

輸出一個 `Array`做為最上層物件

此功能可讓規則將陣列做為最上層物件傳回。舉例而言，若為以下的 MQTT 訊息：

```
{
    "a": {"b":"c"},
    "arr":[1,2,3,4]
}
```

此外以下規則：

```
SELECT VALUE arr FROM 'topic'
```

規則會產生以下輸出：

```
[1,2,3,4]
```

編碼函式

根據指定的編碼機制，將承載 (可能並非 JSON 資料) 編碼為字串表現形式。

基本擷取

基本擷取可讓您安全地將裝置資料傳送至 AWS IoT規則動作 (p. 260) 支援的 AWS 服務，而不會發生訊息費用。基本擷取透過從擷取路徑移除發佈/訂閱訊息中介裝置來將資料流程最佳化，因此更具成本效益。

若要使用基本擷取，您從裝置或應用程式傳送的訊息需要以 `$aws/rules/rule-name` 做為首三個層級主題名稱的開頭，而 *`rule-name`* 是您要觸發的 AWS IoT 規則名稱。

您只需透過將基本擷取字首 (`$aws/rules/rule-name`) 新增至您通常觸發規則的訊息主題，即可搭配基本擷取使用現有的規則。例如，如果您有一個名為 `BuildingManager` 規則會透過 `Buildings/Building5/Floor2/Room201/Lights` (`"sql": "SELECT * FROM 'Buildings/#'"`) 之類的訊息主題觸發，您可以透過傳送主題為 `$aws/rules/BuildingManager/Buildings/Building5/Floor2/Room201/Lights` 的訊息，使用基本擷取觸發相同的規則。

請注意：

- 您的裝置和規則無法訂閱基本擷取預留主題。如需更多詳細資訊，請參閱 預留主題 (p. 243)。
- 如果您需要發佈/訂閱中介裝置以將訊息分發至多個訂閱者 (例如，將訊息傳遞至其他裝置和規則引擎)，您應該繼續使用 AWS IoT 訊息中介裝置以處理訊息分發。只需使用基本擷取主題以外的主題來發佈您的訊息。

若要使用基本擷取

請確認您的裝置或應用程式正在使用政策 (p. 147)，且已在 `$aws/rules/*` 上發佈許可。或者，您可以使用政策中的 `$aws/rules/rule-name/*` 為個別規則指定許可。除此之外，您的裝置和應用程式可以繼續使用他們與 AWS IoT 核心的現有連線。

當訊息抵達規則引擎時，從基本擷取觸發的規則和透過訊息中介裝置訂閱觸發的規則，在執行或處理錯誤上就沒有差別。

當然，您可以建立用於基本擷取的規則。請謹記以下幾點：

- 基本擷取主題的初始字首 (`$aws/rules/rule-name`) 無法用於 topic(Decimal) (p. 321) 函數。
- 如果您定義了只能由基本擷取觸發的規則，`rule` 定義中 `sql` 欄位的 `FROM` 子句則為選用。如果規則也需要由必須透過訊息中介裝置傳送的其他訊息觸發 (例如，因為那些其他訊息必須分發到多個訂閱者)，則仍然會需要此項。如需詳細資訊，請參閱 AWS IoTSQL 參考 (p. 283)。
- 首三個層級的基本擷取主題 (`$aws/rules/rule-name`) 不會算在主題的八區段長度或 256 個總字元限制中。否則，相同的限制套用如 AWS IoT 限制 所述。
- 如果接收的訊息包含指定非作用中或不存在規則的基本擷取主題，則會在 Amazon CloudWatch 日誌中建立錯誤日誌，以協助您偵錯。如需詳細資訊，請參閱 Rules Engine 日誌 (p. 221)。系統會指出 `RuleNotFound` 指標，讓您可以為此指標建立警示。如需詳細資訊，請參閱 AWS IoT 指標 (p. 202) 中的規則指標。
- 您仍可使用 QoS 1 發佈基本擷取主題。訊息成功交付至規則引擎後，您就會收到 PUBACK。收到 PUBACK 不表示您的規則已經成功完成動作。您可以設定錯誤動作，以在動作執行時處理錯誤。請參閱 錯誤處理 (錯誤動作) (p. 280)。

AWS IoT 適用的 Device Shadow 服務

裝置的「影子」是一份 JSON 文件，用於存放及擷取裝置目前的狀態資訊。Device Shadow 服務會為每個連線至 AWS IoT 的裝置，維護一個影子。您可以透過 MQTT 或 HTTP，使用影子來取得及設定裝置的狀態 (無論該裝置是否連線至網際網路)。對應的物件名稱能識別出獨一的每項裝置之影子。

內容

- Device Shadow 服務資料流程 (p. 332)
- Device Shadow 服務文件 (p. 339)
- 使用影子 (p. 342)
- Device Shadow RESTful API (p. 350)
- 影子 MQTT 主題 (p. 353)
- 影子文件語法 (p. 359)
- 影子錯誤訊息 (p. 362)

Device Shadow 服務資料流程

Device Shadow 服務是一種媒介，可以讓裝置與應用程式擷取及更新裝置的影子。

為說明裝置與應用程式如何與 Device Shadow 服務進行通訊，本節會演練使用 AWS IoT MQTT 用戶端與 AWS CLI，模擬已與網際網路連線的燈泡、應用程式以及 Device Shadow 服務之間的通訊。

Device Shadow 服務使用 MQTT 主題，協助進行應用程式與裝置之間的通訊。若要了解其運作方式，請使用 AWS IoT MQTT 用戶端訂閱下列 QoS 1 的 MQTT 主題：

$aws/things/myLightBulb/shadow/update/accepted

當更新裝置影子成功時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/update/rejected

當更新裝置影子被拒絕時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/update/delta

當偵測到裝置影子的回報區段與所需區段之間出現差異時，Device Shadow 服務會傳送訊息至此主題。如需更多詳細資訊，請參閱 /update/delta (p. 356)。

$aws/things/myLightBulb/shadow/get/accepted

當要求裝置影子成功時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/get/rejected

當要求裝置影子被拒絕時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/delete/accepted

當裝置影子刪除時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/delete/rejected

當要求刪除裝置影子被拒絕時，Device Shadow 服務會傳送訊息至此主題。

$aws/things/myLightBulb/shadow/update/documents

每次成功執行更新裝置影子時，Device Shadow 服務就會發佈狀態文件至此主題。

若要進一步了解 Device Shadow 服務使用的所有 MQTT 主題，請參閱「影子 MQTT 主題 (p. 353)」。

> **Note**
>
> 若想查看 Device Shadow 服務傳送的所有錯誤，建議您訂閱 `.../rejected`主題。

當燈泡上線時，就會傳送 MQTT 訊息至 `$aws/things/myLightBulb/shadow/update`主題，將其目前的狀態傳送至 Device Shadow 服務。

> **Note**
>
> 系統會在第一次嘗試更新裝置影子時將其建立。裝置影子服務會偵測到該影子不存在，而且會建立一個。如果影子存在，就會更新該影子。

若要模擬此情況，請使用 AWS IoT MQTT 用戶端將下列訊息發佈至 `$aws/things/myLightBulb/shadow/update` 主題：

```
{
    "state": {
        "reported": {
            "color": "red"
        }
    }
}
```

此訊息會將燈泡的顏色設為「紅色」。

Device Shadow 服務會傳送下列訊息至 `$aws/things/myLightBulb/shadow/update/accepted`主題，予以回應：

```
{
  "messageNumber": 4,
  "payload": {
    "state": {
      "reported": {
        "color": "red"
      }
    },
    "metadata": {
      "reported": {
        "color": {
          "timestamp": 1469564492
        }
      }
    },
    "version": 1,
    "timestamp": 1469564492
  },
  "qos": 0,
  "timestamp": 1469564492848,
  "topic": "$aws/things/myLightBulb/shadow/update/accepted"
}
```

此訊息表示 Device Shadow 服務已收到了 UPDATE 要求，且已更新了裝置的影子。如果影子不存在，就會建立影子。如果存在，則會以訊息中的資料更新該影子。如果您沒看到訊息發佈至 `$aws/things/myLightBulb/shadow/update/accepted`，請查看 `$aws/things/myLightBulb/shadow/update/rejected` 的訂閱是否有任何錯誤訊息。

此外，Device Shadow 服務會將下列訊息發佈至 `$aws/things/myLightBulb/shadow/update/documents`主題。

```
{
    "previous":null,
    "current":{
```

```
        "state":{
            "reported":{
                "color":"red"
            }
         },
         "metadata":{
            "reported":{
                "color":{
                    "timestamp":1483467764
                }
            }
         },
         "version":1
     },
     "timestamp":1483467764
}
```

每次成功執行更新裝置影子時，就會發佈訊息至 `/update/documents`主題。欲深入了解發佈至此主題的訊息內容，請參閱 影子 MQTT 主題 (p. 353)。

應用程式會與連上線的燈泡互動並請求獲得燈泡的目前狀態。應用程式會傳送空訊息至 `$aws/things/myLightBulb/shadow/get`主題。若要模擬此情況，請使用 AWS IoT MQTT 用戶端將空訊息 ("") 發佈至 `$aws/things/myLightBulb/shadow/get` 主題。

Device Shadow 服務會將要求的影子發佈至 `$aws/things/myLightBulb/shadow/get/accepted`主題，予以回應：

```
{
  "messageNumber": 1,
  "payload": {
    "state": {
      "reported": {
        "color": "red"
      }
    },
    "metadata": {
      "reported": {
        "color": {
          "timestamp": 1469564492
        }
      }
    },
    "version": 1,
    "timestamp": 1469564571
  },
  "qos": 0,
  "timestamp": 1469564571533,
  "topic": "$aws/things/myLightBulb/shadow/get/accepted"
}
```

如果您在 `$aws/things/myLightBulb/shadow/get/accepted`主題沒看到訊息，請檢查 `$aws/things/myLightBulb/shadow/get/rejected` 主題是否有任何錯誤訊息。

應用程式會向使用者顯示此資訊，使用者則請求變更燈泡的顏色 (從紅色到綠色)。為此，應用程式會發佈訊息至 `$aws/things/myLightBulb/shadow/update`主題：

```
{
    "state": {
        "desired": {
            "color": "green"
        }
    }
```

```
}
```

欲模擬此情況，請使用 AWS IoT MQTT 用戶端將前述訊息發佈至 `$aws/things/myLightBulb/shadow/update` 主題。

Device Shadow 服務會傳送訊息至 `$aws/things/myLightBulb/shadow/update/accepted`主題，予以回應：

```
{
  "messageNumber": 5,
  "payload": {
    "state": {
      "desired": {
        "color": "green"
      }
    },
    "metadata": {
      "desired": {
        "color": {
          "timestamp": 1469564658
        }
      }
    },
    "version": 2,
    "timestamp": 1469564658
  },
  "qos": 0,
  "timestamp": 1469564658286,
  "topic": "$aws/things/myLightBulb/shadow/update/accepted"
}
```

以及 `$aws/things/myLightBulb/shadow/update/delta`主題：

```
{
  "messageNumber": 1,
  "payload": {
    "version": 2,
    "timestamp": 1469564658,
    "state": {
      "color": "green"
    },
    "metadata": {
      "color": {
        "timestamp": 1469564658
      }
    }
  },
  "qos": 0,
  "timestamp": 1469564658309,
  "topic": "$aws/things/myLightBulb/shadow/update/delta"
}
```

當 Device Shadow 服務接受影子更新，但產生的影子內所含之所需狀態與回報狀態的值不同時，其會發佈訊息至此主題。

Device Shadow 服務也會發佈訊息至 `$aws/things/myLightBulb/shadow/update/documents`主題：

```
{
    "previous":{
      "state":{
        "reported":{
```

```
          "color":"red"
        }
      },
      "metadata":{
        "reported":{
          "color":{
            "timestamp":1483467764
          }
        }
      },
      "version":1
    },
    "current":{
      "state":{
        "desired":{
          "color":"green"
        },
        "reported":{
          "color":"red"
        }
      },
      "metadata":{
        "desired":{
          "color":{
            "timestamp":1483468612
          }
        },
        "reported":{
          "color":{
            "timestamp":1483467764
          }
        }
      },
      "version":2
    },
    "timestamp":1483468612
}
```

由於燈泡是訂閱 `$aws/things/myLightBulb/shadow/update/delta`主題，所以會接收到訊息，變更其顏色並發佈新的狀態。若要模擬此情況，請使用 AWS IoT MQTT 用戶端將下列訊息發佈至 `$aws/things/myLightBulb/shadow/update` 主題，以更新影子狀態：

```
{
    "state":{
        "reported":{
            "color":"green"
        },
        "desired":null}
    }
}
```

Device Shadow 服務會傳送訊息至 `$aws/things/myLightBulb/shadow/update/accepted`主題，予以回應：

```
{
  "messageNumber": 6,
  "payload": {
    "state": {
      "reported": {
        "color": "green"
      },
      "desired": null
    },
```

```
    "metadata": {
      "reported": {
        "color": {
          "timestamp": 1469564801
        }
      },
      "desired": {
        "timestamp": 1469564801
      }
    },
    "version": 3,
    "timestamp": 1469564801
  },
  "qos": 0,
  "timestamp": 1469564801673,
  "topic": "$aws/things/myLightBulb/shadow/update/accepted"
}
```

以及 `$aws/things/myLightBulb/shadow/update/documents`主題：

```
{
    "previous":{
    "state":{
      "reported":{
        "color":"red"
      }
    },
    "metadata":{
         "reported":{
              "color":{
                  "timestamp":1483470355
              }
          }
      },
      "version":3
    },
    "current":{
      "state":{
        "reported":{
          "color":"green"
        }
      },
      "metadata":{
        "reported":{
          "color":{
            "timestamp":1483470364
          }
        }
      },
      "version":4
    },
    "timestamp":1483470364
}
```

應用程式會從 Device Shadow 服務要求目前的狀態，並會顯示最新的狀態資料。若要模擬此情況，請執行下列命令：

```
aws iot-data get-thing-shadow --thing-name "myLightBulb" "output.txt" && cat "output.txt"
```

> **Note**
>
> 在 Windows 中，省略 `&& cat "output.txt"`，該命令會向主控台會顯示 output.txt 的內容。您可以在記事本或任何文字編輯器中開啟檔案，查看影子的內容。

Device Shadow 服務會回傳影子文件：

```
{
    "state":{
      "reported":{
        "color":"green"
      }
    },
    "metadata":{
      "reported":{
        "color":{
          "timestamp":1469564801
        }
      }
    },
    "version":3,
    "timestamp":1469564864}
```

若要刪除裝置的陰影，空的訊息發佈到 `$aws/things/myLightBulb/shadow/delete` 主題。 AWS IoT 藉由將訊息發佈到 `$aws/things/myLightBulb/shadow/delete/accepted` 主題來回應：

```
{
  "version" : 1,
  "timestamp" : 1488565234
}
```

偵測物件是否連網

若要判斷裝置目前是否連線，請在影子中加入連線設定，並使用 MQTT Last Will and Testament (LWT) 訊息，此訊息會在裝置因錯誤而中斷連線時，將連線設定設置為 `false`。

> **Note**
>
> AWS IoT Device Shadow 服務目前會忽略傳送至 AWS IoT 預留主題 (以 $ 開頭的主題) 的 LWT 訊息，但訂閱用戶端與 AWS IoT 規則引擎仍會處理該訊息。如果您希望 Device Shadow 服務能收到 LWT 訊息，請對非預留主題登錄 LWT 訊息，並建立對預留主題重新發佈訊息的規則。以下為範例，顯示如何從 `my/things/myLightBulb/update`主題建立重新發佈規則來聽取訊息並將訊息重新發佈至 `$aws/things/myLightBulb/shadow/update`。
>
> ```
> {
>  "rule": {
>  "ruleDisabled": false,
>  "sql": "SELECT * FROM 'my/things/myLightBulb/update'",
>  "description": "Turn my/things/ into $aws/things/",
>  "actions": [{
>  "republish": {
>  "topic": "$$aws/things/myLightBulb/shadow/update",
>  "roleArn": "arn:aws:iam::123456789012:role/aws_iot_republish"
>  }
>  }]
>  }
> }
> ```

裝置連線時，就會註冊一則 LWT，將連線設定設置為 `false`：

```
{
    "state": {
        "reported": {
            "connected":"false"
```

```
        }
    }
}
```

此外也會在其更新主題 (`$aws/things/myLightBulb/shadow/update`) 發佈訊息，將連線狀態設置為 `true`：

```
{
     "state": {
        "reported": {
            "connected":"true"
        }
    }
}
```

裝置正常斷線時，就會在其更新主題發佈訊息，並將其連線狀態設置為 `false`：

```
{
    "state": {
        "reported": {
            "connected":"false"
        }
    }
}
```

如果裝置因為錯誤而斷線，其 LWT 訊息就會自動張貼至更新主題。

Device Shadow 服務文件

Device Shadow 服務遵守 JSON 規格的所有規則。值、物件與陣列均存放於裝置的影子文件中。

內容

- 文件屬性 (p. 339)
- Device Shadow 的版本控制 (p. 340)
- 用戶端符記 (p. 340)
- 範例文件 (p. 340)
- 空白部分 (p. 341)
- 陣列 (p. 341)

文件屬性

裝置的影子文件具有下列屬性：

`state`

`desired`

物件所需的狀態。應用程式可將此部分編寫至文件，在不必直接連線至物件的情況下更新物件狀態。

`reported`

物件回報的狀態。物件會在文件中編寫此部分以回報其新狀態。應用程式會讀取文件中的此部分，以判斷物件的狀態。

`metadata`

存放於文件 `state`部分之資料的相關資訊。其中包括針對 `state`部分中各屬性的時間戳記 (Epoch time)，可讓您確定資料是在何時更新的。

> **Note**
>
> 中繼資料不會影響服務限制或定價的文件大小。如需詳細資訊，請參閱 AWS IoT 服務限制。

`timestamp`

指明 AWS IoT 傳輸訊息的時間。物件可使用訊息中的時間戳記以及 `desired`或 `reported` 部分中個別屬性的時間戳記，藉此判斷已更新項目的存在時間，即使該項目不具有內部時鐘亦可。

`clientToken`

裝置特有的字串能夠讓您在 MQTT 環境中將回應與請求建立關聯。

`version`

文件版本。每次文件更新時，版本編號便會自動增加。如此可確保更新的文件為最新版本。

如需更多詳細資訊，請參閱 影子文件語法 (p. 359)。

Device Shadow 的版本控制

Device Shadow 服務支援每則更新訊息 (包括要求與回應) 的版本控制，也就是說每次更新裝置影子時，JSON 文件的版本就會增加。如此可確保兩種情況：

- 如果用戶端嘗試以較舊的版本編號覆寫影子，就會收到錯誤訊息。通知用戶端必須先重新同步裝置的影子，才可進行更新。
- 如果此訊息的版本低於存放於用戶端的訊息版本，則用戶端可以決定不採取行動。

在某些情況下，用戶端可以不提交版本，藉此繞過版本配對。

用戶端符記

您可以使用用戶端符記收發以 MQTT 為基礎的訊息，藉此驗證請求與針對請求的回應中是否包含了同樣的用戶端符記。如此可確保回應與請求之間具有關聯。

> **Note**
>
> 用戶端字符不能超過 64 位元組。長於 64 位元組的用戶端字符將導致 400 (Bad Request) 回應以及一個 Invalid clientToken 錯誤訊息。

範例文件

以下為影子文件的範例：

```
{
    "state" : {
        "desired" : {
          "color" : "RED",
          "sequence" : [ "RED", "GREEN", "BLUE" ]
        },
        "reported" : {
          "color" : "GREEN"
        }
```

```
    },
    "metadata" : {
        "desired" : {
            "color" : {
                "timestamp" : 12345
            },
            "sequence" : {
                "timestamp" : 12345
            }
        },
        "reported" : {
            "color" : {
                "timestamp" : 12345
            }
        }
    },
    "version" : 10,
    "clientToken" : "UniqueClientToken",
    "timestamp": 123456789
}
```

空白部分

影子文件只有在具備所需的狀態時，才會包含 desired區段。例如，以下為不具有 desired部分的有效狀態文件：

```
{
    "reported" : { "temp": 55 }
}
```

reported部分也可以空白：

```
{
    "desired" : { "color" : "RED" }
}
```

如果更新導致 desired或 reported 部分成為 null，則該部分會從文件中移除。欲移除的文件的 desired部分 (例如要回應正在更新狀態的裝置)，請將所需部分設為 null：

```
{
    "state": {
        "reported": {
            "color": "red"
        },
        "desired": null
    }
}
```

影子文件也可能不包含 desired或 reported 區段。在此情況下，影子的文件會空白。例如，以下也是有效的文件：

```
{
}
```

陣列

影子支援陣列，但會將其視為一般值，即更新某陣列會取代整組陣列。您無法只更新陣列中的某部分。

初始狀態：

```
{
    "desired" : { "colors" : ["RED", "GREEN", "BLUE" ] }
}
```

更新：

```
{
    "desired" : { "colors" : ["RED"] }
}
```

最終狀態：

```
{
    "desired" : { "colors" : ["RED"] }
}
```

陣列不能有 null 值。例如，以下陣列無效並且會被拒絕。

```
{
    "desired" : {
        "colors" : [ null, "RED", "GREEN" ]
    }
}
```

使用影子

AWS IoT 提供三種操作裝置影子的方法：

`UPDATE`

建立裝置影子 (若其不存在)，或以要求中所提供的資料更新裝置影子的內容。存放的資料會具有時間戳記資訊，以指明最近何時更新。訊息會傳送至所有訂閱者，告知在 `desired`或 `reported` 狀態 (delta) 的差異。收到訊息的物件或應用程式，可根據 `desired`或 `reported` 間的差異執行動作。例如，裝置可以更新其狀態至所需的狀態，或者應用程式可以更新 UI 以顯示裝置狀態的改變。

`GET`

擷取存放於裝置影子中的最新狀態 (例如，在裝置啟動期間擷取組態及最新的操作狀態)。這個方式會回傳完整的 JSON 文件，包括中繼資料。

`DELETE`

刪除裝置影子，包括其所有內容。這會從資料存放區移除 JSON 文件。您無法還原已刪除的裝置影子，但可以使用相同的名稱建立新的影子。

通訊協定支援

MQTT 與透過 HTTPS 的 RESTful API 都支援這些操作。由於 MQTT 是一種發佈/訂閱的通訊模式，因此 AWS IoT 會實行一組預留主題。為了實行請求–回應行為，在發佈請求主題之前，物件或應用程式應先訂閱這些主題。如需更多詳細資訊，請參閱 影子 MQTT 主題 (p. 353) 及 Device Shadow RESTful API (p. 350)。

更新影子

您可以使用 UpdateThingShadow (p. 352)RESTful API 或發佈至/update (p. 353)主題，來更新裝置的影子。更新只會影響請求中所指定的欄位。

初始狀態：

```
{
    "state": {
        "reported" : {
            "color" : { "r" :255, "g": 255, "b": 0 }
        }
    }
}
```

更新訊息傳送：

```
{
    "state": {
        "desired" : {
            "color" : { "r" : 10 },
            "engine" : "ON"
        }
    }
}
```

裝置接收到由先前 `desired`訊息所觸發關於 `/update/delta` 主題的 `/update` 狀態，接著執行所需的變更。完成後，裝置應會透過影子 JSON 文件中的 `reported`區段，確認其更新狀態。

最終狀態：

```
{
    "state": {
        "reported" : {
            "color" : {  "r" : 10, "g" : 255, "b": 0 },
            "engine" : "ON"
        }
    }
}
```

擷取影子文件

您可以使用 GetThingShadow (p. 351)RESTful API 或訂閱及發佈至/get (p. 356)主題，來擷取裝置的影子。這會擷取整份文件以及 `desired`或 `reported` 狀態之間的差量。

範例文件：

```
{
    "state": {
        "desired": {
            "lights": {
                "color": "RED"
            },
            "engine": "ON"
        },
        "reported": {
            "lights": {
                "color": "GREEN"
```

```
            },
            "engine": "ON"
        }
    },
    "metadata": {
        "desired": {
            "lights": {
                "color": {
                    "timestamp": 123456
                },
                "engine": {
                    "timestamp": 123456
                }
            }
        },
        "reported": {
            "lights": {
                "color": {
                    "timestamp": 789012
                }
            },
            "engine": {
                "timestamp": 789012
            }
        },
        "version": 10,
        "timestamp": 123456789
    }
}
```

回應：

```
{
    "state": {
        "desired": {
            "lights": {
                "color": "RED"
            },
            "engine": "ON"
        },
        "reported": {
            "lights": {
                "color": "GREEN"
            },
            "engine": "ON"
        },
        "delta": {
            "lights": {
                "color": "RED"
            }
        }
    },
    "metadata": {
        "desired": {
            "lights": {
                "color": {
                    "timestamp": 123456
                },
             },
             "engine": {
                "timestamp": 123456
            }
        },
        "reported": {
            "lights": {
```

```
                "color": {
                    "timestamp": 789012
                }
            },
            "engine": {
                "timestamp": 789012
            }
        },
        "delta": {
            "lights": {
                "color": {
                    "timestamp": 123456
                }
            }
        }
    },
    "version": 10,
    "timestamp": 123456789
}
```

樂觀鎖定

您可以使用狀態文件版本，確認您正在更新的裝置影子文件為最新版本。當您為更新請求提供版本，若狀態文件的目前版本與提供的版本不符，則服務會拒絕請求並顯示 HTTP 409 衝突回應代碼。

例如：

初始文件：

```
{
    "state" : {
        "desired" : { "colors" : ["RED", "GREEN", "BLUE" ] }
    },
    "version" : 10
}
```

更新：(版本不符合；將拒絕請求)

```
{
    "state": {
        "desired": {
            "colors": [
                "BLUE"
            ]
        }
    },
    "version": 9
}
```

結果：

```
409 Conflict
```

更新：(版本符合；將接受請求)

```
{
    "state": {
        "desired": {
            "colors": [
                "BLUE"
```

```
            ]
        }
    },
    "version": 10
}
```

最終狀態：

```
{
    "state": {
        "desired": {
            "colors": [
                "BLUE"
            ]
        }
    },
    "version": 11
}
```

刪除資料

您可以發佈至 /update (p. 353)主題，將要刪除的欄位設定為 null，藉此刪除裝置影子中的資料。只要有 `null`值的欄位，就會從文件中刪除。

初始狀態：

```
{
    "state": {
        "desired" : {
            "lights": { "color": "RED" },
            "engine" : "ON"
        },
        "reported" : {
            "lights" : { "color": "GREEN"  },
            "engine" : "OFF"
        }
    }
}
```

更新訊息傳送：

```
{
    "state": {
        "desired": null,
        "reported": {
            "engine": null
        }
    }
}
```

最終狀態：

```
{
    "state": {
        "reported" : {
            "lights" : { "color" : "GREEN" }
        }
    }
}
```

您可以將裝置影子的狀態設定為 `null`，藉此刪除裝置影子中的所有資料。例如，傳送下列訊息會刪除所有狀態資料，但仍會保留裝置的影子。

```
{
    "state": null
}
```

即使裝置影子的狀態為 `null`，其仍會存在。下次進行更新時，影子的版本就會增加。

刪除影子

您可以使用 DeleteThingShadow (p. 352)RESTful API 或發佈至 /delete (p. 358) 主題，來刪除裝置的影子文件。

> **Note**
>
> 刪除裝置的影子並不會刪除物件。刪除物件也不會刪除對應的裝置影子。

初始狀態：

```
{
    "state": {
        "desired" : {
            "lights": { "color": "RED" },
            "engine" : "ON"
        },
        "reported" : {
            "lights" : { "color": "GREEN"  },
            "engine" : "OFF"
        }
    }
}
```

空訊息已發佈至 /delete 主題。

最終狀態：

```
HTTP 404 - resource not found
```

差量狀態

差量 (delta) 狀態是一種虛擬類型的狀態，包含了 `desired`和 `reported` 之間的差異。在 `desired`部分的欄位若未出現於 `reported` 部分，則會納入差量之中。在 `reported`部分的欄位若未出現於 `desired` 部分，則不會納入差量之中。差量包含中繼資料，且其值與 `desired`欄位裡的中繼資料相同。例如：

```
{
    "state": {
        "desired": {
            "color": "RED",
            "state": "STOP"
        },
        "reported": {
            "color": "GREEN",
            "engine": "ON"
        },
        "delta": {
            "color": "RED",
            "state": "STOP"
        }
```

```
    },
    "metadata": {
        "desired": {
            "color": {
                "timestamp": 12345
            },
            "state": {
                "timestamp": 12345
            },
            "reported": {
                "color": {
                    "timestamp": 12345
                },
                "engine": {
                    "timestamp": 12345
                }
            },
            "delta": {
                "color": {
                    "timestamp": 12345
                },
                "state": {
                    "timestamp": 12345
                }
            }
        },
        "version": 17,
        "timestamp": 123456789
    }
}
```

當巢狀物件不同，差量則會包含根路徑。

```
{
    "state": {
        "desired": {
            "lights": {
                "color": {
                    "r": 255,
                    "g": 255,
                    "b": 255
                }
            }
        },
        "reported": {
            "lights": {
                "color": {
                    "r": 255,
                    "g": 0,
                    "b": 255
                }
            }
        },
        "delta": {
            "lights": {
                "color": {
                    "g": 255
                }
            }
        }
    },
    "version": 18,
    "timestamp": 123456789
}
```

Device Shadow 服務會重複查看 `desired`狀態內的所有欄位，然後將其與 `reported` 狀態相比較，藉此計算出差量。

陣列的處理和值相同。如果 `desired`部分的某個陣列與 `reported` 部分的陣列不符，則整組所需陣列都會複製到差量中。

觀察狀態變更

當更新裝置影子時，訊息會發佈於兩個 MQTT 主題：

- $aws/things/*thing-name*/shadow/update/accepted
- $aws/things/*thing-name*/shadow/update/delta

傳送至 `update/delta`主題的訊息是要用於狀態正在更新的物件。此訊息僅包含裝置影子文件之 `desired`和 `reported` 區段間的差異。收到此訊息時，裝置應決定是否進行要求的變更。如果裝置的狀態已變更，則該裝置應將其目前的新狀態發佈至 `$aws/things/thing-name/shadow/update`主題。

裝置和應用程式可以訂閱任一主題，以在文件狀態變更時獲得通知。

以下為該流程的範例：

1. 裝置回報其狀態。
2. 系統在其持久性資料存放區更新狀態文件。
3. 系統發佈差量訊息，其中僅包含差量並以訂閱的裝置為目標。裝置應訂閱此主題以接收更新。
4. 裝置影子發佈已接受訊息，內含整個接收的文件，包括中繼資料。應用程式應訂閱此主題以接收更新。

訊息順序

從 AWS IoT 服務發出的訊息，不保證能夠以任何特定順序送達裝置。

初始狀態文件：

```
{
    "state" : {
        "reported" : { "color" : "blue" }
    },
    "version" : 10,
    "timestamp": 123456777
}
```

更新 1：

```
{
    "state": { "desired" : { "color" : "RED" } },
    "version": 10,
    "timestamp": 123456777
}
```

更新 2：

```
{
    "state": { "desired" : { "color" : "GREEN" } },
    "version": 11,
    "timestamp": 123456778
}
```

最終狀態文件：

```
{
    "state": {
        "reported": { "color" : "GREEN" }
    },
    "version": 12,
    "timestamp": 123456779
}
```

這樣會產生兩個差量訊息：

```
{
    "state": {
        "color": "RED"
    },
    "version": 11,
    "timestamp": 123456778
}
```

```
{
    "state": { "color" : "GREEN" },
    "version": 12,
    "timestamp": 123456779
}
```

裝置可能不會依此順序收到這些訊息。由於這些訊息中的狀態是累積的，因此裝置可以安全捨棄比追蹤中版本編號更舊的所有訊息。如果裝置在接收到版本 11 的差量之前收到了版本 12 的差量，則可安全捨棄版本 11 的訊息。

裁剪影子訊息

若要減少傳送至您裝置的影子訊息大小，請定義只選擇了裝置所需之欄位的規則，然後將訊息重新發佈至裝置目前監聽的 MQTT 主題。

規則是在 JSON 中指定，並應如下所示：

```
{
    "sql": "SELECT state, version FROM '$aws/things/+/shadow/update/delta'",
    "ruleDisabled": false,
    "actions": [{
        "republish": {
            "topic": "${topic(2)}/delta",
            "roleArn": "arn:aws:iam::123456789012:role/my-iot-role"
        }
    }]
}
```

SELECT 陳述會決定將訊息的哪些欄位重新發佈至指定主題。使用 "+" 萬用字元表示所有影子名稱。規則會指定所有符合的訊息都應重新發佈至指定主題。在這種情況下，該 `"topic()"` 函數是用於指定要重新發佈的主題。`topic(2)` 評估的原始主題中的實物名稱。更多關於建立規則的資訊，請參閱規則。

Device Shadow RESTful API

影子會公開下列 URI 以更新狀態資訊：

```
https://endpoint/things/thingName/shadow
```

您的 AWS 帳戶具有特定的端點。若要擷取您的端點，請使用 describe-endpoint 命令。端點的格式如下：

```
identifier.iot.region.amazonaws.com
```

陰影 RESTful API 會遵循相同的 HTTPS 通訊協定/連接埠映射，如 AWS IoT 通訊協定中所述。

API 動作

- GetThingShadow (p. 351)
- UpdateThingShadow (p. 352)
- DeleteThingShadow (p. 352)

> **Note**
>
> 當您使用這些 API 時，務必使用通訊協定、連接埠映射和身分驗證 (p. 242)中所述的連接埠 8443

GetThingShadow

取得特定物件的影子。

回應狀態文件包括 `desired`和 `reported` 狀態之間的差量。

要求

此請求包括標準 HTTP 標頭加上以下 URI：

```
HTTP GET https://endpoint/things/thingName/shadow
```

回應

成功時，回應會包括標準 HTTP 標頭以及下列程式碼與本文：

```
HTTP 200
BODY: response state document
```

如需詳細資訊，請參閱回應狀態文件範例 (p. 360)。

授權

擷取影子需要一項允許發起人可執行 `iot:GetThingShadow`動作的政策。Device Shadow 服務接受兩種形式的身分驗證：使用 IAM 登入資料的簽章版本 4，或使用用戶端憑證的 TLS 交互身分驗證。

下列為允許發起人擷取裝置影子的政策範例：

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "iot:GetThingShadow",
        "Resource": ["arn:aws:iot:region:account:thing/thing"]
    }]
}
```

UpdateThingShadow

更新指定之物件的影子。

更新僅會影響請求狀態文件中所指定的欄位。值為 `null`的任何欄位，皆會從裝置影子中移除。

要求

此請求包含標準 HTTP 標頭和內文加上以下 URI 與本文：

```
HTTP POST https://endpoint/things/thingName/shadow
BODY: request state document
```

如需詳細資訊，請參閱請求狀態文件範例 (p. 360)。

回應

成功時，回應會包括標準 HTTP 標頭以及下列程式碼與本文：

```
HTTP 200
BODY: response state document
```

如需詳細資訊，請參閱回應狀態文件範例 (p. 360)。

授權

更新影子需要一項允許發起人執行 `iot:UpdateThingShadow`動作的政策。Device Shadow 服務接受兩種形式的身分驗證：使用 IAM 登入資料的簽章版本 4，或使用用戶端憑證的 TLS 交互身分驗證。

下列為允許發起人更新裝置影子的政策範例：

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "iot:UpdateThingShadow",
        "Resource": ["arn:aws:iot:region:account:thing/thing"]
    }]
}
```

DeleteThingShadow

刪除特定物件的影子。

要求

此請求包括標準 HTTP 標頭加上以下 URI：

```
HTTP DELETE https://endpoint/things/thingName/shadow
```

回應

成功時，回應會包括標準 HTTP 標頭以及下列程式碼與本文：

```
HTTP 200
BODY: Empty response state document
```

授權

刪除裝置影子需要一項允許發起人執行 `iot:DeleteThingShadow`動作的政策。Device Shadow 服務接受兩種形式的身分驗證：使用 IAM 登入資料的簽章版本 4，或使用用戶端憑證的 TLS 交互身分驗證。

下列為允許發起人刪除裝置影子的政策範例：

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "iot:DeleteThingShadow",
        "Resource": ["arn:aws:iot:region:account:thing/thing"]
    }]
}
```

影子 MQTT 主題

Device Shadow 服務使用預留的 MQTT 主題，讓應用程式與裝置可取得、更新或刪除裝置 (影子) 的狀態資訊。這些主題的名稱以 $aws/things/*thingName*/shadow 開頭。在陰影主題上發佈和訂閱需要主題為基礎的授權。AWS IoT 有權新增主題到現有的主題結構。因此，我們建議您避免使用萬用字元訂閱影子主題。例如，請勿訂閱像是 `$aws/things/thingName/shadow/#` 的主題篩選條件，原因是符合此主題篩選條件的主題數量可能會因為 AWS IoT 引入新的影子主題而增加。關於發佈至這些主題的訊息範例，請參閱 Device Shadow 服務資料流程 (p. 332)。

下列為用來與影子互動的 MQTT 主題。

主題

- /update (p. 353)
- /update/accepted (p. 354)
- /update/documents (p. 355)
- /update/rejected (p. 355)
- /update/delta (p. 356)
- /get (p. 356)
- /get/accepted (p. 357)
- /get/rejected (p. 357)
- /delete (p. 358)
- /delete/accepted (p. 358)
- /delete/rejected (p. 359)

/update

對此主題發佈要求狀態文件，以更新裝置的影子：

```
$aws/things/thingName/shadow/update
```

嘗試更新物件影子狀態的用戶端，會傳送如下的 JSON 請求狀態文件：

```
{
    "state" : {
        "desired" : {
```

```
            "color" : "red",
            "power" : "on"
        }
    }
}
```

更新其影子的物件，會傳送如下的 JSON 要求狀態文件：

```
{
    "state" : {
        "reported" : {
            "color" : "red",
            "power" : "on"
        }
    }
}
```

AWS IoT 會透過發佈至 /update/accepted (p. 354) 或 /update/rejected (p. 355) 予以回應。

如需更多詳細資訊，請參閱 請求狀態文件 (p. 360)。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": ["iot:Publish"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/
update"]
    }]
}
```

/update/accepted

當 AWS IoT 接受裝置的影子變更時，就會發佈回應狀態文件至此主題：

```
$aws/things/thingName/shadow/update/accepted
```

如需更多詳細資訊，請參閱 回應狀態文件 (p. 360)。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
update/accepted"]
      },
      {
        "Effect": "Allow",
```

```
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/update/
accepted"]
      }
    ]
}
```

/update/documents

每當影子成功執行更新，AWS IoT 就會發佈狀態文件至此主題：

```
$aws/things/thingName/shadow/update/documents
```

JSON 文件會包含兩個主要節點：`previous`和 `current`。`previous`節點包含執行更新前的完整影子文件容，而 `current` 則包含成功套用更新後的完整影子文件內容。首次更新 (建立) 影子時，`previous`節點會包含 `null`。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
update/documents"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/update/
accepted"]
      }
    ]
}
```

/update/rejected

當 AWS IoT 拒絕變更裝置的影子時，就會發佈錯誤回應文件至此主題：

```
$aws/things/thingName/shadow/update/rejected
```

如需更多詳細資訊，請參閱 錯誤回應文件 (p. 361)。

範例政策

以下為所需政策的範例：

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

```
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
update/rejected"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/update/
rejected"]
      }
    ]
}
```

/update/delta

當 AWS IoT 接受變更裝置的影子時，就會發佈回應狀態文件至此主題，且該文件內 desired 與 reported 狀態的值會不同：

```
$aws/things/thingName/shadow/update/delta
```

如需更多詳細資訊，請參閱 回應狀態文件 (p. 360)。

發佈詳細資訊

- 發佈於 update/delta的訊息，僅包括 desired 和 reported 部分之間不同的所需屬性。這些屬性全都會包含在其中，無論這些屬性是否原本就包含於目前的更新訊息中或是已經存放於 AWS IoT。在 desired和 reported 部分之間沒有差異的屬性則不會包含在內。
- 如果某個屬性出現在 reported部分，但在 desired 部分卻沒有，則此屬性不會包含在內。
- 如果某個屬性出現在 desired部分，但在 reported 部分卻沒有，則此屬性會包含在內。
- 如果某個屬性在 reported部分被刪除，但仍存在於 desired 部分，則此屬性會包含在內。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
update/delta"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/update/
delta"]
      }
    ]
}
```

/get

發佈空白訊息至此主題可取得裝置的影子：

```
$aws/things/thingName/shadow/get
```

AWS IoT 會透過發佈至 /get/accepted (p. 357) 或 /get/rejected (p. 357) 予以回應。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [ "iot:Publish" ],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/get"]
      }
    ]
}
```

/get/accepted

當 AWS IoT 傳回裝置的影子時，就會發佈回應狀態文件至此主題：

```
$aws/things/thingName/shadow/get/accepted
```

如需更多詳細資訊，請參閱 回應狀態文件 (p. 360)。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
get/accepted"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/get/
accepted"]
      }
    ]
}
```

/get/rejected

當 AWS IoT 無法傳回裝置的影子時，就會發佈錯誤回應文件至此主題：

```
$aws/things/thingName/shadow/get/rejected
```

如需更多詳細資訊，請參閱 錯誤回應文件 (p. 361)。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
get/rejected"]
      },
      {
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/get/
rejected"]
      }
    ]
}
```

/delete

若要刪除裝置的影子，發佈空白訊息即可刪除主題：

```
$aws/things/thingName/shadow/delete
```

訊息的內容會被忽略。

AWS IoT 會透過發佈至 /delete/accepted (p. 358) 或 /delete/rejected (p. 359) 予以回應。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
delete"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/
delete"]
      }
    ]
}
```

/delete/accepted

當刪除裝置的影子時，AWS IoT 就會發佈訊息至此主題：

```
$aws/things/thingName/shadow/delete/accepted
```

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
delete/accepted"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/delete/
accepted"]
      }
    ]
}
```

/delete/rejected

當 AWS IoT 無法刪除裝置的影子時，就會發佈錯誤回應文件至此主題：

```
$aws/things/thingName/shadow/delete/rejected
```

如需更多詳細資訊，請參閱 錯誤回應文件 (p. 361)。

範例政策

以下為所需政策的範例：

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": ["iot:Subscribe"],
        "Resource": ["arn:aws:iot:region:account:topicfilter/$aws/things/thingName/shadow/
delete/rejected"]
      },
      {
        "Effect": "Allow",
        "Action": ["iot:Receive"],
        "Resource": ["arn:aws:iot:region:account:topic/$aws/things/thingName/shadow/delete/
rejected"]
      }
    ]
}
```

影子文件語法

在使用 RESTful API (p. 350) 或 MQTT 發佈/訂閱訊息 (p. 353)的 UPDATE、GET、DELETE 操作中，Device Shadow 服務會使用下列文件。如需詳細資訊，請參閱 Device Shadow 服務文件 (p. 339)。

範例

- 請求狀態文件 (p. 360)
- 回應狀態文件 (p. 360)
- 錯誤回應文件 (p. 361)

請求狀態文件

請求狀態文件的格式如下：

```
{
    "state": {
        "desired": {
            "attribute1": integer2,
            "attribute2": "string2",
            ...
            "attributeN": boolean2
        },
        "reported": {
            "attribute1": integer1,
            "attribute2": "string1",
            ...
            "attributeN": boolean1
        }
    },
    "clientToken": "token",
    "version": version
}
```

- `state`— 更新僅會影響所指定的欄位。
- `clientToken`— 使用後，您可以驗證請求與回應是否包含相同的用戶端符記。
- `version`— 若使用此項目，Device Shadow 服務只有在指定版本與其擁有的最新版本相符時，才會處理更新。

回應狀態文件

回應狀態文件的格式如下：

```
{
    "state": {
        "desired": {
            "attribute1": integer2,
            "attribute2": "string2",
            ...
            "attributeN": boolean2
        },
        "reported": {
            "attribute1": integer1,
            "attribute2": "string1",
            ...
            "attributeN": boolean1
        },
        "delta": {
            "attribute3": integerX,
            "attribute5": "stringY"
        }
    },
    "metadata": {
```

```
        "desired": {
            "attribute1": {
                "timestamp": timestamp
            },
            "attribute2": {
                "timestamp": timestamp
            },
            ...
            "attributeN": {
                "timestamp": timestamp
            }
        },
        "reported": {
            "attribute1": {
                "timestamp": timestamp
            },
            "attribute2": {
                "timestamp": timestamp
            },
            ...
            "attributeN": {
                "timestamp": timestamp
            }
        }
    },
    "timestamp": timestamp,
    "clientToken": "token",
    "version": version
}
```

- `state`
  - `reported`— 只有當物件在 `reported` 部分回報任何資料，且包含僅位於請求狀態文件中的欄位時才出現。
  - `desired`— 只有當物件在 `desired` 部分回報任何資料，且包含僅位於請求狀態文件中的欄位時才出現。
  - `delta` — 只有在物件報告物件影子文件的 `reported` 和 `desired` 小節中有不同資料時才會出現。
- `metadata`— 包含 `desired` 和 `reported` 部分中每個屬性的時間戳記，因此您可確定狀態的更新時間。
- `timestamp`— 由 AWS IoT 產生回應的 Epoch 日期與時間。
- `clientToken`— 僅在發佈有效 JSON 至 `/update` 主題並使用用戶端符記時才會出現。
- `version`—AWS IoT 中目前共用之裝置影子的文件版本。此版本會在先前的文件版本編號加一。

錯誤回應文件

錯誤回應文件的格式如下：

```
{
    "code": error-code,
    "message": "error-message",
    "timestamp": timestamp,
    "clientToken": "token"
}
```

- `code`— 用於指明錯誤類型的 HTTP 回應碼。
- `message`— 可提供額外資訊的文字訊息。
- `timestamp`—AWS IoT 產生回應的日期與時間。
- `clientToken`— 僅在發佈有效 JSON 至 `/update` 主題並使用用戶端符記時才會出現。

如需更多詳細資訊，請參閱 影子錯誤訊息 (p. 362)。

影子錯誤訊息

當嘗試變更狀態文件失敗時，Device Shadow 服務會透過 MQTT 發佈訊息至錯誤主題。只有在回應針對其中一個預留 $aws 主題的發佈請求時，此訊息才會發送。如果用戶端使用 REST API 更新文件，則會收到 HTTP 錯誤代碼做為部分回應，而 MQTT 錯誤訊息也不會發出。

HTTP 錯誤代碼	錯誤訊息
400 (錯誤的請求)	• 無效的 JSON • 缺少必要的節點：狀態 • 狀態節點必須是物件 • 所需節點必須是物件 • 報告節點必須是物件 • 版本無效 • clientToken 無效 Note 用戶端字符超過 64 位元組將會導致這個回應。 • JSON 包含太多巢狀層級；最大數量為 6 • 狀態包含無效的節點
401 (未經授權)	• 未經授權
403 (禁止)	• 禁止
404 (未找到)	• 找不到物件
409 (衝突)	• 版本衝突
413 (承載過大)	• 承載超過允許的最大值
415 (不支援的媒體類型)	• 不支援的文件編碼；支援的編碼為 UTF-8
429 (太多請求)	• 當傳送中的要求超過 10 則時，Device Shadow 服務就會產生此錯訊息。
500 (內部伺服器錯誤)	• 內部服務故障

任務

AWS IoT 任務可用來定義一組遠端操作，這組操作會傳送到連接 AWS IoT 的一個或多個裝置並且執行。

Jobs 的重要概念

job

任務即遠端操作，其會傳送到同時在連線到 AWS IoT 的一或多項裝置上執行。例如，您可以定義一個工作，指示一組裝置下載並安裝應用程式或韌體更新、重新啟動、輪換憑證，或者執行遠端故障排除操作。

任務文件

若要建立任務，您必須先建立任務文件，而其為裝置要執行之遠端操作的描述。

任務文件為 UTF-8 編碼的 JSON 文件，應會包含您裝置執行工作所需的資訊。任務文件包含一個或多個 URL，讓裝置可以下載更新或其他資料。任務文件可以存放於 Amazon S3 儲存貯體，或者內嵌於建立任務的命令。

目標

當您建立任務時，要指定目標的清單，也就是要執行操作的裝置。目標可以是物件或 物件群組 (p. 112)，或同時為兩者。AWS IoT Jobs 服務會傳送訊息到每個目標，通知該目標已有工作可執行。

工作執行

工作執行即目標裝置上工作的執行個體。目標會下載工作文件，藉此開始執行工作。接著會執行文件中指定的操作，並對 AWS IoT 回報其進度。執行編號是特定目標上不會重複的工作執行之識別符。任務服務提供命令來追蹤目標上某工作執行的進度和所有目標上的工作進度。

快照工作

在預設情況下，當您建立工作，該工作會傳送至您指定的所有目標。這些目標完成工作之後 (或者回報無法完成工作)，工作即視為完成。

持續性工作

當您建立工作，持續任務會傳送至您指定的所有目標。它會繼續執行並傳送任何新的裝置 (物件) 到已新增的目標群組。例如：持續工作加入群組時，可用於部署或更新裝置。您可以在建立工作時設定選用參數，藉此讓工作可以持續。

推展

您可以指定何時通知目標有一項待執行的任務。這樣可以建立一個分階段推展的任務，讓您更輕易管理更新、重新啟動以及其他操作。

下列欄位可新增到 `CreateJob` 請求，以指定每分鐘通知任務目標的數目上限。此範例設定靜態推展速率。

```
"jobExecutionRolloutConfig": {
    "maximumPerMinute": "integer"
}
```

您也可以使用 `exponentialRate` 欄位設定變數推展速率。以下範例建立具有指數速率的推展。

```
  "jobExecutionsRolloutConfig": {
      "exponentialRate": {
         "baseRatePerMinute": integer,
         "incrementFactor": integer,
         "rateIncreaseCriteria": {
            "numberOfNotifiedThings": integer, // Set one or the other
            "numberOfSucceededThings": integer // of these two values.
         },
         "maximumPerMinute": integer
    }
}
```

如需設定任務推展的詳細資訊，請參閱任務推展和中止組態。

中止

您可以建立一組條件，當滿足您指定的特定條件後中止推展。如需詳細資訊，請參閱 任務推展和中止組態。

預先簽章的 URL

若要讓裝置以安全、限時的方式存取任務文件以外的資料，您可以使用預先簽章的 Amazon S3 URL。您可以將資料置於 Amazon S3儲存貯體，並在任務作文件中的資料新增一個預留位置連結。任務服務收到對於任務文件的請求時，就會剖析任務文件，尋找預留位置連結並以預先簽章的 Amazon S3 URL 取代。

預留位置連結的格式如下：

```
${aws:iot:s3-presigned-url:https://s3.amazonaws.com/bucket/key}
```

其中，*bucket* 是您的儲存貯體名稱，而 *key* 是您所連結在儲存貯體中的物件。

逾時

> **Note**
>
> 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

當任務執行長時間非預期卡在 `IN_PROGRESS` 狀態時，任務逾時會嘗試通知。計時器有兩種類型：進行中計時器和步驟計時器。

當您建立任務時，您可以設定選用 TimeoutConfig 物件的 `inProgressTimeoutInMinutes` 屬性值。進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 `IN_PROGRESS` 狀態的時間超過此間隔時，任務執行就會失敗，並切換到結束 `TIMED_OUT` 狀態。AWS IoT 也會發佈 MQTT 通知。

您也可以透過在呼叫 UpdateJobExecution 時設定 `stepTimeoutInMinutes` 的值，來為任務執行設定步驟計時器。步驟計時器僅適用於您更新的任務執行。您可以在每次更新任務執行時，為此計時器設定新值。當您呼叫 StartNextPendingJobExecution 時，您也可以建立步驟計時器。如果任務執行維持在 `IN_PROGRESS` 狀態的時間超過步驟計時器隔時，任務執行就會失敗，並切換到終止的 `TIMED_OUT` 狀態。當您建立任務時，步驟計時器不會影響您設定的進行中計時器。

下圖和描述說明進行中逾時和步驟逾時的互動方式。

任務建立： `CreateJob` 會設定二十分鐘後逾時的進行中計時器。此計時器適用於所有任務執行，並且無法更新。

12:00 PM：任務執行開始，並切換到 `IN_PROGRESS` 狀態。進行中計時器開始執行。

12:05 PM： `UpdateJobExecution`建立步驟計時器，值為 7 分鐘。如未建立新的步驟計時器，則任務執行在 12:12 PM 過期。

12:10 PM： `UpdateJobExecution`建立新步驟計時器，值為 5 分鐘。會捨棄上述步驟計時器。如未建立新的步驟計時器，則任務執行在 12:15 PM 過期。

12:13 PM： `UpdateJobExecution`建立新步驟計時器，值為 9 分鐘。任務執行會在 12:20 逾期，因為進行中計時器在 12:20 逾期。步驟計時器無法超過進行中計時器建立的絕對界限。

`UpdateJobExecution` 也可以透過建立值為 -1 的新步驟計時器，來捨棄已建立的步驟計時器。

管理工作

您可以使用 AWS IoT 主控台、任務 HTTPS API、AWS Command Line Interface 或 AWS 開發套件來建立和管理任務。如需詳細資訊，請參閱 工作管理和控制 API (p. 385)、AWS CLI 命令列參考：iot 或 AWS 開發套件與工具。

任務的主要目的是通知裝置軟體或韌體更新。當將程式碼傳送到裝置時，最佳實務是簽署程式碼檔案。這可讓裝置偵測程式碼是否在傳輸過程中遭到修改。以下部分中的說明假設您要對要傳送到裝置的軟體更新進行程式碼簽署。

如需詳細資訊，請參閱 什麼是 AWS IoT 程式碼簽署？

建立任務之前，您必須建立任務文件。如果您使用適用於 AWS IoT 的程式碼簽署，您必須將任務文件上傳到已建立版本的 Amazon S3 儲存貯體。如需建立 Amazon S3 儲存貯體和將檔案上傳至其中的詳細資訊，請參閱 Amazon S3 入門指南中的 Amazon Simple Storage Service 入門。

您的任務文件可以包含指向程式碼檔 (或其他檔案) 的預先簽章 Amazon S3 URL。預先簽章的 Amazon S3 URL 在有限的時間內有效，因此在裝置請求任務文件之後才會產生。由於在建立任務文件時尚未建立預先簽章的 URL，因此請在任務文件中新增預留位置 URL。預留位置 URL 如下所示：`${aws:iot:s3-presigned-url:https://s3.region.amazonaws.com/<bucket>/<code file>}`，其中`####`是包含程式碼檔的 Amazon S3 儲存貯體，而`####`是程式碼檔的 Amazon S3 金鑰。

當裝置請求任務文件時，AWS IoT 會產生預先簽章的 URL，並以預先簽章的 URL 取代預留位置 URL。您的任務文件接著會傳送到裝置。

當您建立使用預先簽章 Amazon S3 URL 的任務時，您必須提供授予許可的 IAM 角色，以從存放資料或更新的 Amazon S3 儲存貯體下載檔案。此角色也必須授與 AWS IoT 許可來擔任此角色。

您可指定預先簽章 URL 的選擇性逾時值。如需更多詳細資訊，請參閱 CreateJob (p. 399)。

若要授與任務服務權限到您假設的角色

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. 在導覽窗格中，選擇 Roles (角色)。
3. 搜尋並選取您的角色。
4. 在 Trust Relationships (信任關係) 標籤上，選擇 Edit Trust Relationship (編輯信任關係)。
5. 在 Edit Trust Relationship (編輯信任關係) 頁面，以下列 JSON 取代政策文件：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "iot.amazonaws.com"
        ]
      },
```

```
        "Action": "sts:AssumeRole"
      }
    ]
}
```

6. 選擇 Update Trust Policy (更新信任政策)。
7. 如果您的任務使用的任務文件是 Amazon S3 物件，請選擇 Permissions (許可) 和下列 JSON，並新增授予許可的政策，以從您的 Amazon S3 儲存貯體下載檔案：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::your_S3_bucket/*"
        }
    ]
}
```

建立和管理任務 (主控台)

如果您使用適用於 AWS IoT 的程式碼簽署，您必須在任務文件中新增兩個預留位置 URL：

程式碼檔的預留位置應如下所示：`${aws:iot:s3-presigned-url:https://s3.amazonaws.com/<my-s3-bucket>/<my-code-file>}`。

> **Note**
>
> 目前，任務的預先簽章 URL 預留位置並不支援版本。如果您更新程式碼檔，並將其複製到相同的 Amazon S3 位置，您必須建立新的簽章，然後在任務文件中參考新的簽章版本。

簽章的預留位置應如下所示：`${aws:iot:code-sign-signature:s3://<region>.<my-s3-bucket>/<my-code-file>@<code-file-version-id>}`。

建立任務

1. 瀏覽至 AWS IoT 主控台。
2. 在導覽窗格中，選擇 Manage (管理)，然後選擇 Jobs (任務)。
3. 選擇 Create a job (建立任務)。
4. 選擇 Create a custom job (建立自訂任務)。
5. 輸入任務的英數字元 ID 和選擇性描述。

   > **Note**
   >
   > 我們不建議在任務 ID 或描述中使用個人識別資訊。
6. 選取您要更新的裝置或裝置群組。
7. 在 Add a job file (新增任務檔案) 下，選擇 Select (選取)，然後選取您的任務文件。
8. 選擇 Sign image for me (幫我簽署映像)。如果您不是對更新進行程式碼簽署，則可以略過此步驟。
9. 建立或選擇程式碼簽署設定檔。如果您不是對更新進行程式碼簽署，則可以略過此步驟。
10. 在 Pre-sign resource URLs (預先簽章的資源 URL) 下，選擇 I want to pre-sign my URLs and have configured my job file (我想要預先簽署我的 URL，並已設定任務檔案。)。如果您不是對更新進行程式碼簽署，則可以略過此步驟。
11. 選擇角色和預先簽章 URL 的到期時間。
12. 在 Job type (任務類型) 下，選擇適當的更新選項，然後選擇 Next (下一步)。
13. 指定任何進階組態的值，然後選擇 Create (建立)。

建立任務後，主控台會產生 JSON 簽章，並將其放置在您的任務文件中。

您可以使用 AWS IoT 主控台來檢視狀態，或取消、刪除任務。

1. 瀏覽至 AWS IoT 主控台。
2. 在導覽窗格中，選擇 Manage (管理)，然後選擇 Jobs (任務)。

建立和管理任務 (CLI)

本節說明如何建立和管理任務。

建立工作

您可以使用 CreateJob 命令來建立 AWS IoT 任務。工作會排入您所指定目標 (物件或物件群組) 的執行佇列。若要建立 AWS IoT 任務，您需要可以加入請求本文中，或做為 Amazon S3 文件連結的任務文件。如果任務包含使用預先簽章的 Amazon S3 URL 下載檔案，您就需要 IAM 角色 ARN 以得到許可下載檔案，並授予 AWS IoT 任務服務許可來擔任此角色。

使用任務進行程式碼簽署

如果您使用適用於 AWS IoT 的程式碼簽署，您必須開始程式碼簽署任務，並將輸出包含在任務文件中。使用 start-signing-job 命令來建立程式碼簽署任務。`start-signing-job` 會傳回任務 ID。使用 describe-signing-job 命令以取得簽章的 Amazon S3 存放位置。然後，您可以從 Amazon S3 下載簽章。如需程式碼簽署任務的詳細資訊，請參閱適用於 AWS IoT 的程式碼簽署。

您的任務文件必須包含程式碼檔的預先簽章 URL 預留位置，以及使用在 `codesign` 元素中包含之 start-signing-job 命令放置在 Amazon S3 儲存貯體的 JSON 簽章輸出：

```
{
    "presign": "${aws:iot:s3-presigned-url:https://s3.region.amazonaws.com/bucket/image}",
    "codesign": {
        "rawPayloadSize": <image-file-size>,
        "signature": <signature>,
        "signatureAlgorithm": <signature-algorithm>,
        "payloadLocation": {
            "s3": {
                "bucketName": <my-s3-bucket>,
                "key": <my-code-file>,
                "version": <code-file-version-id>
            }
        }
    }
}
```

使用任務文件建立任務

以下命令說明如何使用在 Amazon S3 儲存貯體 (*jobBucket*) 中存放的任務文件 (*job-document.json*) 建立任務，並使用具有許可的角色來下載 Amazon S3 (*S3DownloadRole*) 中的檔案。

```
aws iot create-job  \
      --job-id 010  \
      --targets arn:aws:iot:us-east-1:123456789012:thing/thingOne  \
      --document-source https://s3.amazonaws.com/my-s3-bucket/job-document.json  \
      --timeout-config inProgressTimeoutInMinutes=100 \
      --job-executions-rollout-config "{ \"exponentialRate\": { \"baseRatePerMinute\": 50,
 \"incrementFactor\": 2, \"rateIncreaseCriteria\": { \"numberOfNotifiedThings\": 1000,
 \"numberOfSucceededThings\": 1000}, \"maximumPerMinute\": 1000}}" \
```

```
      --abort-config "{ \"criteriaList\": [ { \"action\": \"CANCEL\", \"failureType\":
 \"FAILED\", \"minNumberOfExecutedThings\": 100, \"thresholdPercentage\": 20}, { \"action
\": \"CANCEL\", \"failureType\": \"TIMED_OUT\", \"minNumberOfExecutedThings\": 200,
 \"thresholdPercentage\": 50}]}" \
      --presigned-url-config "{\"roleArn\":\"arn:aws:iam::123456789012:role/
S3DownloadRole\", \"expiresInSec\":3600}"
```

該任務在 *thingOne* 上執行。

選用的 `timeout-config` 參數，會指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態 ，就會將其設為 `TIMED_OUT`。

> **Note**
>
> 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 `IN_PROGRESS` 狀態的時間超過此間隔時，任務執行就會失敗，並切換到結束 `TIMED_OUT` 狀態。AWS IoT 也會發佈 MQTT 通知。

如需建立任務推展和中止組態的詳細資訊，請參閱任務推展和中止組態。

> **Note**
>
> 指定為 Amazon S3 檔案的任務文件會在您建立任務時擷取。在您建立任務之後，若變更做為任務文件來源的 Amazon S3 檔案內容，並不會變更傳送至任務目標的內容。

更新任務

您可以使用 UpdateJob 命令來更新任務。您可以更新任務的 `description`、`presignedUrlConfig`、`jobExecutionsRolloutConfig`、`abortConfig` 和 `timeoutConfig` 欄位。

```
aws iot update-job  \
  --job-id 010  \
  --description "updated description" \
  --timeout-config inProgressTimeoutInMinutes=100 \
  --job-executions-rollout-config "{ \"exponentialRate\": { \"baseRatePerMinute\": 50,
 \"incrementFactor\": 2, \"rateIncreaseCriteria\": { \"numberOfNotifiedThings\": 1000,
 \"numberOfSucceededThings\": 1000}, \"maximumPerMinute\": 1000}}" \
  --abort-config "{ \"criteriaList\": [ { \"action\": \"CANCEL\", \"failureType\":
 \"FAILED\", \"minNumberOfExecutedThings\": 100, \"thresholdPercentage\": 20}, { \"action
\": \"CANCEL\", \"failureType\": \"TIMED_OUT\", \"minNumberOfExecutedThings\": 200,
 \"thresholdPercentage\": 50}]}" \
  --presigned-url-config "{\"roleArn\":\"arn:aws:iam::123456789012:role/S3DownloadRole\",
 \"expiresInSec\":3600}"
```

如需詳細資訊，請參閱 任務推展和中止組態。

取消任務

您可以使用 CancelJob 命令來取消任務。取消任務會使 AWS IoT 停止為任務部署新的任務執行。它還會取消狀態為 `QUEUED` 的任何任務執行。AWS IoT 會使結束狀態為 的任務執行維持不變，由於裝置已完成任務。如果任務執行的狀態為 `IN_PROGRESS`，則該任務執行也將維持不變，除非您使用選用的 `--force` 參數。

以下命令說明如何使用 ID 010 來取消任務。

```
aws iot cancel-job --job-id 010
```

該命令會顯示下列輸出：

```
{
    "jobArn": "string",
    "jobId": "string",
    "description": "string"
}
```

當您取消任務時，狀態為 QUEUED 的任務執行將會遭到取消。若您指定選用的 --force 參數，狀態為 IN_PROGRESS 的任務執行將被取消。處於結束狀態的任務執行不會取消。

> **Warning**
>
> 取消狀態為 IN_PROGRESS (透過設定 --force 參數) 的任務將取消任何進行中的任務執行，並導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可以復原至有效狀態的。

取消的任務或其中一項任務執行的狀態具有最終一致性，AWS IoT 會盡快停止安排新的任務執行，並且不將該任務的 QUEUED 任務執行提供給裝置。將任務執行的狀態變更為 CANCELED (取消) 可能會需要一些時間，視裝置數量及其他因素而定。

如果任務因符合 AbortConfig 物件定義的條件而被取消，則服務新增自動填入 comment 和 reasonCode 欄位的值。當任務取消是由使用者驅動時，您可以建立自己的 reasonCode 值。

取消任務執行

您可以使用 CancelJobExecution 命令來取消裝置上的任務執行。這麼做會取消狀態為 QUEUED 的任務執行。如果您想要取消進行中的任務執行，您必須使用 --force 參數。

以下命令說明如何從 myThing 上執行的任務 010 取消任務執行。

```
aws iot cancel-job-execution --job-id 010 --thing-name myThing
```

此命令不會顯示輸出。

系統會取消狀態為 QUEUED 的任務執行。若您指定選用的 --force 參數，狀態為 IN_PROGRESS 的任務執行將被取消。處於結束狀態的任務執行無法被取消。

> **Warning**
>
> 取消狀態為 IN_PROGRESS 的任務執行時，將導致裝置無法更新任務執行狀態。請謹慎使用並確認裝置可以還原為有效狀態。

若任務執行為結束狀態或任務執行為 IN_PROGRESS 狀態，且 --force 參數未設定為 true，此命令將導致 InvalidStateTransitionException。

取消的任務執行之狀態最終會保持一致。將任務執行的狀態變更為 CANCELED (取消) 可能會需要一些時間，依不同因素而定。

刪除任務

您可以使用 DeleteJob 命令來刪除任務及其任務執行。依預設，您只能刪除處於結束狀態 (SUCCEEDED 或 CANCELED) 的任務。否則會發生例外狀況。如果 force 參數設為「true」時，您可以刪除狀態為「IN_PROGRESS」的任務。

執行下列命令以刪除任務：

```
aws iot delete-job --job-id 010 --force|--no-force
```

此命令不會顯示輸出。

> **Warning**
>
> 當您刪除狀態為 `IN_PROGRESS` 的任務時，正在執行任務的裝置將無法存取任務資訊或更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已刪除任務是可以復原至有效狀態的。

刪除任務可能會需要一些時間，根據為任務建立的任務執行數量與其他因素而定。在刪除任務期間，任務狀態會顯示為「`DELETION_IN_PROGRESS`」。嘗試刪除或取消狀態已為「`DELETION_IN_PROGRESS`」的任務將會導致錯誤。

只有 10 個任務可以同時具有 `DELETION_IN_PROGRESS` 狀態。否則會發生 `LimitExceededException`。

取得任務文件

您可以使用 GetJobDocument命令從工作中擷取任務文件。任務文件為由裝置執行遠端操作的描述。

執行下列命令來取得任務文件：

```
aws iot get-job-document --job-id 010
```

此命令會傳回指定工作的工作文件：

```
{
    "document": "{\n\t\"operation\":\"install\",\n\t\"url\":\"http://amazon.com/
firmWareUpate-01\",\n\t\"data\":\"${aws:iot:s3-presigned-url:https://s3.amazonaws.com/job-
test-bucket/datafile}\"\n}"
}
```

> **Note**
>
> 當您使用此命令擷取任務文件時，預留位置 URL 並不會被預先簽章的 Amazon S3 URL 取代。當裝置呼叫 GetPendingJobExecutions (p. 439) MQTT API 時，預留位置 URL 才會被任務文件中的預先簽章 Amazon S3 URL 取代。

列出任務

您可以使用 ListJobs命令取得您在 AWS 帳戶中所有工作的清單。任務資料及任務執行資料會在 90 天後清除。執行下列命令以列出 AWS 帳戶中的所有任務：

```
aws iot list-jobs
```

此命令會傳回您帳戶中所有任務，並依任務狀態排序：

```
{
    "jobs": [
        {
            "status": "IN_PROGRESS",
            "lastUpdatedAt": 1486687079.743,
            "jobArn": "arn:aws:iot:us-east-1:123456789012:job/013",
            "createdAt": 1486687079.743,
            "targetSelection": "SNAPSHOT",
            "jobId": "013"
        },
        {
            "status": "SUCCEEDED",
            "lastUpdatedAt": 1486685868.444,
            "jobArn": "arn:aws:iot:us-east-1:123456789012:job/012",
```

```
            "createdAt": 1486685868.444,
            "completedAt": 148668789.690,
            "targetSelection": "SNAPSHOT",
            "jobId": "012"
        },
        {
            "status": "CANCELED",
            "lastUpdatedAt": 1486678850.575,
            "jobArn": "arn:aws:iot:us-east-1:123456789012:job/011",
            "createdAt": 1486678850.575,
            "targetSelection": "SNAPSHOT",
            "jobId": "011"
        }
    ]
}
```

描述任務

執行 DescribeJob 命令以取得任務的狀態。以下命令說明如何描述任務：

```
$ aws iot describe-job --job-id 010
```

此命令會傳回指定任務的狀態。例如：

```
{
    "documentSource": "https://s3.amazonaws.com/job-test-bucket/job-document.json",
    "job": {
        "status": "IN_PROGRESS",
        "jobArn": "arn:aws:iot:us-east-1:123456789012:job/010",
        "targets": [
            "arn:aws:iot:us-east-1:123456789012:thing/myThing"
        ],
        "jobProcessDetails": {
            "numberOfCanceledThings": 0,
            "numberOfFailedThings": 0,
            "numberOfInProgressThings": 0,
            "numberOfQueuedThings": 0,
            "numberOfRejectedThings": 0,
            "numberOfRemovedThings": 0,
            "numberOfSucceededThings": 0,
            "numberOfTimedOutThings": 0,
            "processingTargets": [
                arn:aws:iot:us-east-1:123456789012:thing/thingOne,
                arn:aws:iot:us-east-1:123456789012:thinggroup/thinggroupOne,
                arn:aws:iot:us-east-1:123456789012:thing/thingTwo,
                arn:aws:iot:us-east-1:123456789012:thinggroup/thinggroupTwo
            ]
        },
        "presignedUrlConfig": {
            "expiresInSec": 60,
            "roleArn": "arn:aws:iam::123456789012:role/S3DownloadRole"
        },
        "jobId": "010",
        "lastUpdatedAt": 1486593195.006,
        "createdAt": 1486593195.006,
        "targetSelection": "SNAPSHOT",
        "jobExecutionsRolloutConfig": {
            "exponentialRate": {
                "baseRatePerMinute": integer,
                "incrementFactor": integer,
                "rateIncreaseCriteria": {
                    "numberOfNotifiedThings": integer, // Set one or the other
                    "numberOfSucceededThings": integer // of these two values.
```

```
                },
            "maximumPerMinute": integer
         }
        },
        "abortConfig": {
            "criteriaList": [
                {
                    "action": "string",
                    "failureType": "string",
                    "minNumberOfExecutedThings": integer,
                    "thresholdPercentage": integer
                }
            ]
        },
        "timeoutConfig": {
           "inProgressTimeoutInMinutes": number
          }
    }
}
```

列出工作的執行

工作執行物件代表在特定裝置上運作的工作。執行 ListJobExecutionsForJob 命令以列出任務中的所有任務執行。以下顯示如何列出任務執行：

```
aws iot list-job-executions-for-job --job-id 010
```

此命令會傳回一份工作執行清單：

```
{
    "executionSummaries": [
    {
        "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/thingOne",
        "jobExecutionSummary": {
            "status": "QUEUED",
            "lastUpdatedAt": 1486593196.378,
            "queuedAt": 1486593196.378,
            "executionNumber": 1234567890
        }
    },
    {
        "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/thingTwo",
        "jobExecutionSummary": {
            "status": "IN_PROGRESS",
            "lastUpdatedAt": 1486593345.659,
            "queuedAt": 1486593196.378,
            "startedAt": 1486593345.659,
            "executionNumber": 4567890123
        }
    }
    ]
}
```

列出物件的工作執行

執行 ListJobExecutionsForThing 命令以列出在物件上運作的所有工作執行。以下顯示如何列出物件的任務執行：

```
aws iot list-job-executions-for-thing --thing-name thingOne
```

此命令會傳回一份清單，列出特定物件上正在運作或已經運作的工作執行：

```
{
    "executionSummaries": [
    {
        "jobExecutionSummary": {
            "status": "QUEUED",
            "lastUpdatedAt": 1486687082.071,
            "queuedAt": 1486687082.071,
            "executionNumber": 9876543210
        },
        "jobId": "013"
    },
    {
        "jobExecutionSummary": {
            "status": "IN_PROGRESS",
            "startAt": 1486685870.729,
            "lastUpdatedAt": 1486685870.729,
            "queuedAt": 1486685870.729,
            "executionNumber": 1357924680
        },
        "jobId": "012"
    },
    {
        "jobExecutionSummary": {
            "status": "SUCCEEDED",
            "startAt": 1486678853.415,
            "lastUpdatedAt": 1486678853.415,
            "queuedAt": 1486678853.415,
            "executionNumber": 4357680912
        },
        "jobId": "011"
    },
    {
        "jobExecutionSummary": {
            "status": "CANCELED",
            "startAt": 1486593196.378,
            "lastUpdatedAt": 1486593196.378,
            "queuedAt": 1486593196.378,
            "executionNumber": 2143174250
        },
        "jobId": "010"
    }
    ]
}
```

描述工作執行

執行 DescribeJobExecution 命令以取得任務執行的狀態。你必須指定任務 ID 及物件名稱，或者可選擇是否指定執行編號來識別任務執行。以下顯示如何描述任務執行：

```
aws iot describe-job-execution --job-id 017 --thing-name thingOne
```

此命令會傳回 JobExecution (p. 389)。例如：

```
{
    "execution": {
        "jobId": "017",
        "executionNumber": 4516820379,
        "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/thingOne",
        "versionNumber": 123,
        "createdAt": 1489084805.285,
```

```
        "lastUpdatedAt": 1489086279.937,
        "startedAt": 1489086279.937,
        "status": "IN_PROGRESS",
        "approximateSecondsBeforeTimedOut": 100,
        "statusDetails": {
            "status": "IN_PROGRESS",
            "detailsMap": {
                "percentComplete": "10"
            }
        }
    }
}
```

刪除任務執行

執行 DeleteJobExecution 命令以刪除任務執行。你必須指定任務 ID (物件名稱) 和執行編號來識別任務執行。以下顯示如何刪除任務執行：

```
aws iot delete-job-execution --job-id 017 --thing-name thingOne --execution-number
 1234567890 --force|--no-force
```

此命令不會顯示輸出。

在預設情況下，任務執行的狀態必須為 `QUEUED` 或結束狀態 (`SUCCEEDED`、`FAILED`、`REJECTED`、`TIMED_OUT`、`REMOVED` 或 `CANCELED`)。否則會發生錯誤。若要刪除狀態為 `IN_PROGRESS` 的任務執行，您可以將 `force` 參數設為 `true`。

> **Warning**
>
> 當您刪除狀態為 `IN_PROGRESS` 的任務執行時，正在執行任務的裝置將無法存取任務資訊或更新任務執行狀態。請謹慎使用並確保裝置可以還原為有效狀態。

裝置和任務

Device Communication with Jobs

裝置可以透過下列方法與 AWS IoT 任務服務通訊：

- MQTT
- HTTP Signature 第 4 版
- HTTP TLS

Using the MQTT Protocol

AWS IoT 任務服務與您的裝置之間可透過 MQTT 通訊協定來通訊。裝置訂閱 MQTT 主題後，即可收到新任務的通知以及來自 AWS IoT 任務服務的回應。裝置發佈於 MQTT 主題以查詢或更新工作執行的狀態。每項裝置皆有其一般性 MQTT 主題。如需發佈與訂閱 MQTT 主題的詳細資訊，請參閱 AWS IoT 的訊息中介裝置 (p. 242)。

> **Note**
>
> 您必須使用正確的端點，透過 MQTT 與 AWS IoT 任務服務進行通訊。使用 DescribeEndpoint 命令來加以尋找。例如，如果您執行此命令：
>
> ```
> aws iot describe-endpoint --endpoint-type iot:Data
> ```
>
> 您會得到類似以下的結果：

```
{
    "endpointAddress": "a1b2c3d4e5f6g7.iot.us-west-2.amazonaws.com"
}
```

運用此方法，您的裝置可以使用其裝置專屬的憑證與私有金鑰，進行 AWS IoT 任務服務的驗證。

裝置可以：

- 藉由訂閱 `$aws/things/thing-name/jobs/notify` MQTT 主題 (其中 `thing-name` 是與裝置關聯的物件名稱)，在待定任務執行清單新增或移除任務執行時獲得通知。
- 藉由訂閱 `$aws/things/thing-name/jobs/notify-next`MQTT 主題 (其中 `thing-name` 是與裝置關聯的物件名稱)，在下一個待定工作執行發生變更時獲得通知。
- 藉由呼叫 UpdateJobExecution (p. 451)API 以更新工作執行的狀態。
- 藉由呼叫 DescribeJobExecution (p. 447)API 以查詢工作執行的狀態。
- 藉由呼叫 GetPendingJobExecutions (p. 439)API 以擷取待定工作執行的清單。
- 以 jobId DescribeJobExecution (p. 447)呼叫 `$next` API 來擷取下一個待定工作執行。
- 呼叫 StartNextPendingJobExecution (p. 442)API 以取得並開始下一個待定工作執行。

AWS IoT 任務服務會在 MQTT 主題上發佈成功和失敗訊息。主題是透過將 `accepted` 或 `rejected` 附加到用於發出請求的主題來形成。例如，如果在 `$aws/things/myThing/jobs/get` 主題上發佈請求訊息，AWS IoT 任務服務會將接受訊息發佈在 `$aws/things/myThing/jobs/get/accepted` 主題，將拒絕訊息發佈在 `$aws/things/myThing/jobs/get/rejected` 主題。

Using HTTP Signature Version 4

AWS IoT 任務服務與您的裝置之間，可在連接埠 443 上透過 HTTP Signature 第 4 版進行通訊。此為 AWS 開發套件與 CLI 使用的方法。如需這些工具的詳細資訊，請參閱 AWS CLI 命令參考：iot-jobs-data 或 AWS 開發套件與工具，並參閱您慣用語言的 IotJobsDataPlane 章節。

> **Note**
>
> 透過 HTTP Signature 第 4 版或使用 AWS 開發套件或 CLI IotJobsDataPlane 命令與 AWS IoT 任務服務進行通訊時，必須使用正確的端點。使用 DescribeEndpoint 命令來加以尋找。例如，如果您執行此命令：
>
> ```
> aws iot describe-endpoint --endpoint-type iot:Jobs
> ```
>
> 您會得到類似以下的結果：
>
> ```
> {
>  "endpointAddress": "a1b2c3d4e5f6g7.jobs.iot.us-west-2.amazonaws.com"
> }
> ```
>
> 若使用此通訊方法，您的裝置會使用 IAM 登入資料，進行 AWS IoT 任務服務的驗證。

使用此方法可運用下列命令：

- DescribeJobExecution

  ```
  aws iot-jobs-data describe-job-execution ...
  ```
- GetPendingJobExecutions

  ```
  aws iot-jobs-data get-pending-job-executions ...
  ```
- StartNextPendingJobExecution

  ```
  aws iot-jobs-data start-next-pending-job-execution ...
  ```

- UpdateJobExecution

  ```
  aws iot-jobs-data update-job-execution ...
  ```

Using HTTP TLS

AWS IoT 任務服務與您裝置之間可使用支援此協定的第三方軟體用戶端，在連接埠 8443 上透過 HTTP TLS 進行通訊。

> **Note**
>
> 您必須使用正確的端點，透過 HTTP TLS 與 AWS IoT 任務服務進行通訊。使用 DescribeEndpoint 命令來加以尋找。例如，如果您執行此命令：
>
> ```
> aws iot describe-endpoint --endpoint-type iot:Jobs
> ```
>
> 您會得到類似以下的結果：
>
> ```
> {
>  "endpointAddress": "a1b2c3d4e5f6g7.jobs.iot.us-west-2.amazonaws.com"
> }
> ```
>
> 若使用此方法，您的裝置會使用 X.509 憑證型身份驗證 (例如，其裝置專屬的憑證與私有金鑰)。

使用此方法可運用下列命令：

- DescribeJobExecution
- GetPendingJobExecutions
- StartNextPendingJobExecution
- UpdateJobExecution

設定裝置與 Jobs 合作

本節中的範例使用 MQTT 說明裝置與 AWS IoT 任務服務之間如何運作。或者，您也可以使用對應的 API 或 CLI 命令。在這些範例中，我們會假設名為 `MyThing`的裝置將訂閱下列 MQTT 主題：

- `$aws/things/`*`MyThing`*`/jobs/notify` (或 `$aws/things/`*`MyThing`*`/jobs/notify-next`)
- `$aws/things/`*`MyThing`*`/jobs/get/accepted`
- `$aws/things/`*`MyThing`*`/jobs/get/rejected`
- `$aws/things/`*`MyThing`*`/jobs/`*`jobId`*`/get/accepted`
- `$aws/things/`*`MyThing`*`/jobs/`*`jobId`*`/get/rejected`

如果您使用適用於 AWS IoT 的程式碼簽署，您的裝置程式碼必須驗證程式碼檔的簽章。該簽章是包含任務文件中的 `codesign` 屬性。如需驗證程式碼檔簽章的詳細資訊，請參閱裝置代理程式範例。

裝置工作流程

裝置可透過兩種方式處理該執行的工作。

Option A: Get the next job

1. 裝置首次上線時，應訂閱裝置的 `notify-next`主題。

2. 以 jobId `$next`呼叫 DescribeJobExecution (p. 447) MQTT API 來取得下一個工作、工作文件及其他詳細資訊，包括任何儲存於 `statusDetails` 的狀態。如果任務文件具有程式碼檔簽章，則必須先驗證簽章，然後再繼續處理任務請求。
3. 呼叫 UpdateJobExecution (p. 451)MQTT API 以更新工作狀態。或者，裝置可以呼叫 StartNextPendingJobExecution (p. 442)，將此步驟與前一步驟結合至同一個呼叫裡。
4. (選用) 當您呼叫 UpdateJobExecution (p. 451) 或 StartNextPendingJobExecution (p. 442) 時，您可以透過設定 `stepTimeoutInMinutes` 的值來新增步驟計時器。
5. 執行由使用 UpdateJobExecution (p. 451)MQTT API 之工作文件所指定的動作，以回報工作進度。
6. 使用此 jobId 來呼叫 DescribeJobExecution (p. 447) MQTT API 即可繼續監控任務執行。如果任務執行在裝置執行任務時被取消或刪除，裝置應可還原為有效狀態。
7. 任務完成時呼叫 UpdateJobExecution (p. 451)MQTT API 以更新任務狀態並回報成功或失敗。
8. 由於此任務的執行狀態已變更為結束狀態，因此下一個可執行的任務 (若有) 將會變更。狀置會收到下一個待定工作執行已變更的通知。此時，裝置應繼續如步驟 2 所述運作。

如果裝置維持在線上，只要裝置完成任務或新增待定任務執行，就會繼續收到下一個待定任務執行的通知，包括其任務執行資料。在此情況下，裝置應繼續如步驟 2 所述運作。

Option B: Pick from available jobs

1. 裝置首次上線時，應訂閱物件的 `notify`主題。
2. 呼叫 GetPendingJobExecutions (p. 439)MQTT API 以取得待定工作執行清單。
3. 如果清單包含了一個或多個工作執行，則從中選擇一個。
4. 呼叫 DescribeJobExecution (p. 447) MQTT API 以取得任務文件及其他詳細資訊，包括於 `statusDetails` 儲存的任何狀態。
5. 呼叫 UpdateJobExecution (p. 451)MQTT API 以更新工作狀態。如果 `includeJobDocument` 欄位在此命令中設定為 `true`，則裝置可以略過先前的步驟並擷取當前的任務文件。
6. 或者，當您呼叫 UpdateJobExecution (p. 451) 時，您可以透過設定 `stepTimeoutInMinutes` 的值來新增步驟計時器。
7. 執行由使用 UpdateJobExecution (p. 451)MQTT API 之工作文件所指定的動作，以回報工作進度。
8. 使用此 jobId 來呼叫 DescribeJobExecution (p. 447) MQTT API 即可繼續監控任務執行。如果任務執行在裝置執行任務時被取消或刪除，裝置應可還原為有效狀態。
9. 任務完成時呼叫 UpdateJobExecution (p. 451)MQTT API 以更新任務狀態並回報成功或失敗。

如果裝置維持在線上，只要有新的待定任務執行出現，裝置就會收到所有待定任務執行的通知。在此情況下，裝置可繼續如步驟 2 所述運作。

如果裝置無法執行工作，則應呼叫 UpdateJobExecution (p. 451)MQTT API 將工作狀態更新為 `REJECTED`。

開始新工作

new job notification

新任務建立時，AWS IoT 任務服務會針對每個目標裝置在 `$aws/things/`*`thing-name`*`/jobs/notify` 主題發佈訊息。

More Information(1)

訊息包含下列資訊：

```
{
```

```
    "timestamp":1476214217017,
    "jobs":{
        "QUEUED":[{
            "jobId":"0001",
            "queuedAt":1476214216981,
            "lastUpdatedAt":1476214216981,
            "versionNumber" : 1
        }]
    }
}
```

當任務執行排入佇列時，裝置就會收到這個以 `'$aws/things/thingName/jobs/notify'` 為主題的訊息。

get job information

若需取得更多有關任務執行的資訊，該裝置會呼叫 DescribeJobExecution (p. 447) MQTT API，且 `includeJobDocument` 欄位將設為 `true` (此為預設)。

More Information(2)

如果請求成功，AWS IoT 任務服務會以 `$aws/things/MyThing/jobs/0023/get/accepted` 主題發佈訊息：

```
{
    "clientToken" : "client-001",
    "timestamp" : 1489097434407,
    "execution" : {
        "approximateSecondsBeforeTimedOut": number,
        "jobId" : "023",
        "status" : "QUEUED",
        "queuedAt" : 1489097374841,
        "lastUpdatedAt" : 1489097374841,
        "versionNumber" : 1,
        "jobDocument" : {
            < contents of job document >
        }
    }
}
```

Note

如果請求失敗，AWS IoT 任務服務會以 `$aws/things/MyThing/jobs/0023/get/rejected` 主題發佈訊息。

裝置現在具有可用於執行任務遠端操作的任務文件。如果任務文件包含 Amazon S3 預先簽章的 URL，則裝置可以使用此 URL 下載任務需要的所有檔案。

報告工作執行狀態

Update Execution Status

裝置執行工作時，可以呼叫 UpdateJobExecution (p. 451)MQTT API 以更新工作執行的狀態。

More Information (3)

例如，裝置可以藉由以 `IN_PROGRESS`主題發佈下列訊息而將工作執行狀態更新為 `$aws/things/MyThing/jobs/0023/update`：

```
{
    "status":"IN_PROGRESS",
```

```
    "statusDetails": {
        "progress":"50%"
    },
    "expectedVersion":"1",
    "clientToken":"client001"
}
```

Jobs 會予以回應，將訊息發佈至 `$aws/things/MyThing/jobs/0023/update/accepted`或 `$aws/things/MyThing/jobs/0023/update/rejected` 主題：

```
{
    "clientToken":"client001",
    "timestamp":1476289222841
}
```

裝置可以透過呼叫 StartNextPendingJobExecution (p. 442) 來合併前兩個請求。這會取得並開始下一個待定任務執行，並能讓裝置更新任務執行狀態。此請求也會在工作執行待定時傳回工作文件。

如果任務包含 TimeoutConfig，則進行中計時器會開始執行。您也可以透過在呼叫 UpdateJobExecution 時設定 `stepTimeoutInMinutes` 的值，來為任務執行設定步驟計時器。步驟計時器僅適用於您更新的任務執行。您可以在每次更新任務執行時，為此計時器設定新值。當您呼叫 StartNextPendingJobExecution 時，您也可以建立步驟計時器。如果任務執行維持在 `IN_PROGRESS` 狀態的時間超過步驟計時器隔時，任務執行就會失敗，並切換到終止的 `TIMED_OUT` 狀態。當您建立任務時，步驟計時器不會影響您設定的進行中計時器。

> **Note**
>
> 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

`status` 欄位可以設定為 `IN_PROGRESS`、`SUCCEEDED` 或 `FAILED`。若工作執行已經處於結束狀態，您就無法更新其狀態。

Report Execution Completed

裝置完成執行工作時，會呼叫 UpdateJobExecution (p. 451)MQTT API。如果任務成功，則將 `status` 設定為 `SUCCEEDED`，並在訊息承載的 `statusDetails` 中以名稱值對的方式加入任務的其他相關資訊。任務執行完成時，執行中與步驟計時器會結束。

More Information(4)

例如：

```
{
    "status":"SUCCEEDED",
    "statusDetails": {
        "progress":"100%"
    },
    "expectedVersion":"2",
    "clientToken":"client-001"
}
```

如果工作不成功，則將 `status`設定為 `FAILED`，並在 `statusDetails` 加入所發生錯誤的資訊：

```
{
    "status":"FAILED",
    "statusDetails": {
        "errorCode":"101",
        "errorMsg":"Unable to install update"
    },
    "expectedVersion":"2",
    "clientToken":"client-001"
```

```
}
```

> **Note**
>
> `statusDetails` 屬性可以包含任意數量的名稱值對。

當 AWS IoT 任務服務收到此更新，就會以 `$aws/things/MyThing/jobs/notify` 主題發佈訊息，表示任務執行完成：

```
{
    "timestamp":1476290692776,
    "jobs":{}
}
```

額外工作

additional jobs

如果裝置還有其他待定的工作執行，則會包含在發佈至 `$aws/things/MyThing/jobs/notify`的訊息中。

More Information(5)

例如：

```
{
    "timestamp":1476290692776,
    "jobs":{
        "QUEUED":[{
            "jobId":"0002",
            "queuedAt":1476290646230,
            "lastUpdatedAt":1476290646230
        }],
        "IN_PROGRESS":[{
            "jobId":"0003",
            "queuedAt":1476290646230,
            "lastUpdatedAt":1476290646230
        }]
    }
}
```

任務通知

當任務待處理或者當清單中第一個任務執行變動時，AWS IoT 任務服務會將 MQTT 訊息發佈到預留主題。裝置可訂閱這些主題以隨時追蹤待定的任務。

任務通知會做為 JSON 承載發佈到 MQTT 主題。通知有兩種：

- `ListNotification` 包含不超過 10 個待定任務執行的清單。在這個清單中的任務執行具有 `IN_PROGRESS` 或 `QUEUED` 的狀態值。它們會依狀態 (`IN_PROGRESS` 任務執行前的 `QUEUED` 任務執行)，然後依其佇列的時間排序。

  每當符合以下其中一個條件，就發佈 `ListNotification`。
  - 新的任務執行會排入佇列或變更為非終結狀態 (`IN_PROGRESS` 或 `QUEUED`)。
  - 舊任務執行會變更為終止狀態 (`FAILED`、`SUCCEEDED`、`CANCELED`、`TIMED_OUT`、`REJECTED` 或 `REMOVED`)。
- `NextNotification` 包含佇列中下一個任務執行的摘要資訊。

當清單中第一個任務執行變更時，就發佈 NextNotification。

- 新的任務執行會新增到清單中做為 QUEUED，而且會是清單中的第一個。
- 清單中不是第一個的現有任務執行的狀態，從 QUEUED 變更 IN_PROGRESS，並且成為清單中的第一個。(清單中沒有其他 IN_PROGRESS 任務執行時，或者其狀態從 QUEUED 變更為 IN_PROGRESS 的任務執行比清單中任何其他 IN_PROGRESS 任務執行較早排入佇列時，就會發生這種情況)。
- 任務執行的狀態，即清單中第一個變更為終結狀態並從清單中移除。

如需發佈與訂閱 MQTT 主題的詳細資訊，請參閱 AWS IoT 的訊息中介裝置 (p. 242)。

Note

當您使用 HTTP Signature 第 4 版或 HTTP TLS 來與任務通訊時，通知無法使用。

job pending

當從物件的待定任務執行清單新增或移除任務，或者清單中的第一個任務變更時，AWS IoT 任務服務就會在 MQTT 主題上發佈訊息：

- $aws/things/*thingName*/jobs/notify
- $aws/things/*thingName*/jobs/notify-next

More Information(6)

訊息包含的承載範例如下：

$aws/things/*thingName*/jobs/notify：

```
{
  "timestamp" : 10011,
  "jobs" : {
    "IN_PROGRESS" : [ {
      "jobId" : "other-job",
      "queuedAt" : 10003,
      "lastUpdatedAt" : 10009,
      "executionNumber" : 1,
      "versionNumber" : 1
    } ],
    "QUEUED" : [ {
      "jobId" : "this-job",
      "queuedAt" : 10011,
      "lastUpdatedAt" : 10011,
      "executionNumber" : 1,
      "versionNumber" : 0
    } ]
  }
}
```

$aws/things/*thingName*/jobs/notify-next：

```
{
  "timestamp" : 10011,
  "execution" : {
    "jobId" : "other-job",
    "status" : "IN_PROGRESS",
    "queuedAt" : 10009,
    "lastUpdatedAt" : 10009,
    "versionNumber" : 1,
    "executionNumber" : 1,
    "jobDocument" : {"c":"d"}
```

```
  }
}
```

可能的任務執行狀態為 QUEUED、IN_PROGRESS、FAILED、SUCCEEDED、CANCELED、TIMED_OUT、REJECTED 及 REMOVED。

以下範例系列顯示任務執行建立、並從一個狀態變為另一個狀態時，發佈到各主題的通知。

首先，建立一個任務，稱為 job1。此通知會發佈到 jobs/notify 主題：

```
{
  "timestamp": 1517016948,
  "jobs": {
    "QUEUED": [
      {
        "jobId": "job1",
        "queuedAt": 1517016947,
        "lastUpdatedAt": 1517016947,
        "executionNumber": 1,
        "versionNumber": 1
      }
    ]
  }
}
```

此通知會發佈到 jobs/notify-next 主題：

```
{
  "timestamp": 1517016948,
  "execution": {
    "jobId": "job1",
    "status": "QUEUED",
    "queuedAt": 1517016947,
    "lastUpdatedAt": 1517016947,
    "versionNumber": 1,
    "executionNumber": 1,
    "jobDocument": {
      "operation": "test"
    }
  }
}
```

建立另一個任務 (job2) 時，此通知會發佈至 jobs/notify 主題：

```
{
  "timestamp": 1517017192,
  "jobs": {
    "QUEUED": [
      {
        "jobId": "job1",
        "queuedAt": 1517016947,
        "lastUpdatedAt": 1517016947,
        "executionNumber": 1,
        "versionNumber": 1
      },
      {
        "jobId": "job2",
        "queuedAt": 1517017191,
        "lastUpdatedAt": 1517017191,
        "executionNumber": 1,
        "versionNumber": 1
```

```
      }
    ]
  }
}
```

通知不會發佈至 jobs/notify-next 主題，因為佇列中的下一個任務 (job1) 尚未變更。當 job1 開始執行時，它的狀態變更為 IN_PROGRESS。不會發佈通知，因為任務清單與佇列中下一個任務尚未變更。

新增第三個任務 (job3) 時，此通知會發佈至 jobs/notify 主題：

```
{
  "timestamp": 1517017906,
  "jobs": {
    "IN_PROGRESS": [
      {
        "jobId": "job1",
        "queuedAt": 1517016947,
        "lastUpdatedAt": 1517017472,
        "startedAt": 1517017472,
        "executionNumber": 1,
        "versionNumber": 2
      }
    ],
    "QUEUED": [
      {
        "jobId": "job2",
        "queuedAt": 1517017191,
        "lastUpdatedAt": 1517017191,
        "executionNumber": 1,
        "versionNumber": 1
      },
      {
        "jobId": "job3",
        "queuedAt": 1517017905,
        "lastUpdatedAt": 1517017905,
        "executionNumber": 1,
        "versionNumber": 1
      }
    ]
  }
}
```

通知不會發佈至 jobs/notify-next 主題，因為佇列中的下一個任務仍然是 (job1)。

當 job1 完成時，它的狀態會變更為 SUCCEEDED，而且此通知會發佈至 jobs/notify 主題：

```
{
  "timestamp": 1517186269,
  "jobs": {
    "QUEUED": [
      {
        "jobId": "job2",
        "queuedAt": 1517017191,
        "lastUpdatedAt": 1517017191,
        "executionNumber": 1,
        "versionNumber": 1
      },
      {
        "jobId": "job3",
        "queuedAt": 1517017905,
        "lastUpdatedAt": 1517017905,
        "executionNumber": 1,
```

```
        "versionNumber": 1
      }
    ]
  }
}
```

此時，系統會從佇列中移除 job1，而下一個要執行的任務為 job2。此通知會發佈到 jobs/notify-next 主題：

```
{
  "timestamp": 1517186269,
  "execution": {
    "jobId": "job2",
    "status": "QUEUED",
    "queuedAt": 1517017191,
    "lastUpdatedAt": 1517017191,
    "versionNumber": 1,
    "executionNumber": 1,
    "jobDocument": {
      "operation": "test"
    }
  }
}
```

如果 job3 必須在 job2 前開始執行 (不建議這麼做)，可將 job3 的狀態變更為 IN_PROGRESS。這種情況下，job2 將不會是佇列中的下一個任務，而此通知會發佈至 jobs/notify-next 主題：

```
{
  "timestamp": 1517186779,
  "execution": {
    "jobId": "job3",
    "status": "IN_PROGRESS",
    "queuedAt": 1517017905,
    "startedAt": 1517186779,
    "lastUpdatedAt": 1517186779,
    "versionNumber": 2,
    "executionNumber": 1,
    "jobDocument": {
      "operation": "test"
    }
  }
}
```

不會將通知發佈到 jobs/notify 主題，因為未新增或移除任何任務。

如果裝置拒絕 job2 並將其狀態更新為 REJECTED，此通知會發佈至 jobs/notify 主題：

```
{
  "timestamp": 1517189392,
  "jobs": {
    "IN_PROGRESS": [
      {
        "jobId": "job3",
        "queuedAt": 1517017905,
        "lastUpdatedAt": 1517186779,
        "startedAt": 1517186779,
        "executionNumber": 1,
        "versionNumber": 2
      }
    ]
  }
}
```

如果 `job3` (仍在進行中) 是強制刪除，此通知會發佈到 `jobs/notify` 主題：

```
{
  "timestamp": 1517189551,
  "jobs": {}
}
```

此時，佇列是空的。此通知會發佈到 `jobs/notify-next` 主題：

```
{
  "timestamp": 1517189551
}
```

使用 AWS IoT 任務 API

AWS IoT 任務服務使用的 API 分為兩類：

- 用於管理與控制工作的 API。
- 由裝置使用以執行工作的 API。

一般而言，工作管理與控制會使用 HTTPS 通訊協定的 API。裝置可以使用 MQTT 或 HTTPS 通訊協定的 API。(HTTPS API 適用於少量呼叫，特別是在建立與追蹤工作時使用。其通常會為單一請求開啟連線，並在接收到回應之後關閉連線。MQTT API 可用於長輪詢。其適用於可擴展至數百萬個裝置的大量流量。)

Note

每個 AWS IoT 任務 HTTPS API 都有一項對應的命令，可讓您從 AWS CLI 呼叫 API。命令為小寫字母，且在組成 API 名稱的單字之間有連字號。例如，您可以輸入以下內容來呼叫 CLI 上的 `CreateJob`API：

```
aws iot create-job ...
```

工作管理和控制 API

工作管理和控制資料類型

管理和控制應用程式會使用下列資料類型與 AWS IoT 任務服務通訊。

任務

Job Data Type

`Job`物件包含了工作的詳細資訊。

Syntax (1)

```
{
    "jobArn": "string",
    "jobId": "string",
    "status": "IN_PROGRESS|CANCELED|SUCCEEDED",
    "forceCanceled": boolean,
    "targetSelection": "CONTINUOUS|SNAPSHOT",
    "comment": "string",
    "targets": ["string"],
    "description": "string",
    "createdAt": timestamp,
```

```
    "lastUpdatedAt": timestamp,
    "completedAt": timestamp,
    "jobProcessDetails": {
        "processingTargets": ["string"],
        "numberOfCanceledThings": long,
        "numberOfSucceededThings": long,
        "numberOfFailedThings": long,
        "numberOfRejectedThings": long,
        "numberOfQueuedThings": long,
        "numberOfInProgressThings": long,
        "numberOfRemovedThings": long,
        "numberOfTimedOutThings": long
    },
    "presignedUrlConfig": {
        "expiresInSec": number,
        "roleArn": "string"
    },
    "jobExecutionsRolloutConfig": {
        "exponentialRate": {
           "baseRatePerMinute": integer,
           "incrementFactor": integer,
           "rateIncreaseCriteria": {
              "numberOfNotifiedThings": integer, // Set one or the other
              "numberOfSucceededThings": integer // of these two values.
           },
           "maximumPerMinute": integer
      }
    },
    "abortConfig": {
       "criteriaList": [
          {
             "action": "string",
             "failureType": "string",
             "minNumberOfExecutedThings": integer,
             "thresholdPercentage": integer
          }
       ]
    },
    "timeoutConfig": {
        "inProgressTimeoutInMinutes": long
    }
}
```

Description (1)

`jobArn`

以 "arn:aws:iot:*region*:*account*:job/*jobId*" 格式識別工作的 ARN。

`jobId`

您在建立此工作時所指派的獨特識別符。

`status`

工作的狀態，為 `IN_PROGRESS`、`CANCELED` 或 `SUCCEEDED` 其中一項。

`targetSelection`

指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。

`comment`

如果工作已更新，則會說明更新原因。

`targets`

應收到任務的 AWS IoT 實物及實物群組的清單。

`description`

關於工作的簡短文字描述。

`createdAt`

任務建立的 epoch 時間，以秒為單位。

`lastUpdatedAt`

任務最近一次更新的 epoch 時間，以秒為單位。

`completedAt`

任務完成的 epoch 時間，以秒為單位。

`jobProcessDetails`

關於工作流程的詳細資訊：

`processingTargets`

目前正在執行任務的 AWS IoT 實物及實物群組清單。

`numberOfCanceledThings`

取消任務的 AWS IoT 實物數量。

`numberOfSucceededThings`

成功完成任務的 AWS IoT 實物數量。

`numberOfFailedThings`

無法完成任務 AWS IoT 實物數量。

`numberOfRejectedThings`

拒絕任務的 AWS IoT 實物數量。

`numberOfQueuedThings`

等待執行任務的 AWS IoT 實物數量。

`numberOfInProgressThings`

目前正在執行任務的 AWS IoT 實物數量。

`numberOfRemovedThings`

因為從當成任務目標的群組中被刪除或移除而不再排定執行任務的 AWS IoT 實物數量。

`numberOfTimedOutThings`

任務執行狀態為 `TIMED_OUT` 的實物數目。

`presignedUrlConfig`

預先簽章的 Amazon S3 URL 的組態資訊。

`expiresInSec`

預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。

`roleArn`

IAM 角色的 ARN，可授權從 Amazon S3 儲存貯體下載檔案。此角色也必須授予許可讓 AWS IoT 下載檔案。如需建立和設定角色的詳細資訊，請參閱 建立工作 (p. 367)。

`jobExecutionRolloutConfig`

選用。可讓您建立工作的分階段推展。

`maximumPerMinute`

每分鐘收到要執行的任務之物件 (裝置) 最大數量。

`exponentialRate`

可讓您為任務建立指數推展速率。

`baseRatePerMinute`

在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。

`incrementFactor`

提高任務推展速率的指數因素。

`rateIncreaseCriteria`

任務啟動增加推展速率的條件。您可以指定 `numberOfNotifiedThings` 或 `numberOfSucceededThing`，但不能同時指定兩者。

`numberOfNotifiedThings`

啟動增加推展速率通知物件數量的閾值。

`numberOfSucceededThings`

啟動增加推展速率成功物件數量的閾值。

`abortConfig`

選用。任務中止條件的詳細資訊。

`criteriaList`

用於定義中止任務規則的中止條件清單。

`action`

用於初始化任務中止的中止動作類型。

`failureType`

用於定義初始化任務中止的任務執行失敗類型。

`minNumberOfExecutedThings`

評估中止規則之前的已執行物件最低數量。

`thresholdPercentage`

以閥值做為啟動任務中止的已執行物件總數百分比。

`timeoutConfig`

選用。指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 `TIMED_OUT`。

> **Note**
>
> 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

`inProgressTimeoutInMinutes`

指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。只要此任務的執行狀態指定為 `IN_PROGRESS` 且已填入此欄位，就會啟動或重新啟動計時器。如果任務執行狀態未在計時器逾期之前設定為終止狀態，或者在填入此欄位的情況下傳送另一個任務執行狀態更新之前，會將狀態設定為 `TIMED_OUT`。

JobSummary

JobSummary Data Type

`JobSummary`物件包含了工作摘要。

Syntax (2)

```
{
    "jobArn": "string",
    "jobId": "string",
    "status": "IN_PROGRESS|CANCELED|SUCCEEDED",
    "targetSelection": "CONTINUOUS|SNAPSHOT",
    "thingGroupId": "string",
    "createdAt": timestamp,
    "lastUpdatedAt": timestamp,
    "completedAt": timestamp
}
```

Description (2)

`jobArn`

識別工作的 ARN。

`jobId`

您在建立此工作時所指派的獨特識別符。

`status`

工作狀態。可以是 `IN_PROGRESS`，`CANCELED` 或 `SUCCEEDED` 其中之一。

`targetSelection`

指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。

`thingGroupId`

物件群組的 ID。

`createdAt`

建立工作時的 UNIX 時間戳記。

`lastUpdatedAt`

工作最近一次更新時的 UNIX 時間戳記。

`completedAt`

工作完成時的 UNIX 時間戳記。

JobExecution

JobExection Data Type

`JobExecution` 物件代表在裝置上執行任務。

Syntax (3)

```
{
    "approximateSecondsBeforeTimedOut": 50,
    "executionNumber": 1234567890,
```

```
    "forceCanceled": true|false,
    "jobId": "string",
    "lastUpdatedAt": timestamp,
    "queuedAt": timestamp,
    "startedAt": timestamp,
    "status": "QUEUED|IN_PROGRESS|FAILED|SUCCEEDED|CANCELED|TIMED_OUT|REJECTED|
REMOVED",
    "forceCanceled": boolean,
    "statusDetails": {
        "detailsMap": {
            "string": "string" ...
        },
        "status": "string"
    },
    "thingArn": "string",
    "versionNumber": 123
}
```

Description (3)

approximateSecondsBeforeTimedOut

將任務執行狀態變更為 `TIMED_OUT` 之前要保留狀態的預估秒數。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。實際任務執行逾時可能會比預估持續時間最多晚 60 秒發生。

`jobId`

您在建立此工作時所指派的獨特識別符。

`executionNumber`

可識別在此裝置上之任務執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。

`thingArn`

AWS IoT 實物 ARN。

`queuedAt`

任務排入佇列的 epoch 時間，以秒為單位。

`lastUpdatedAt`

任務執行最近一次更新的 epoch 時間，以秒為單位。

`startedAt`

任務執行開始的 epoch 時間，以秒為單位。

`status`

工作執行的狀態。可以是 `QUEUED`、`IN_PROGRESS`、`FAILED`、`SUCCEEDED`、`CANCELED`、`TIMED_OUT`、`REJECTED` 或 `REMOVED`。

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。

JobExecutionSummary

JobExecutionSummary Data Type

`JobExecutionSummary`物件包含了工作執行摘要的資訊：

Syntax (4)

```
{
```

```
    "executionNumber": 1234567890,
    "queuedAt": timestamp,
    "lastUpdatedAt": timestamp,
    "startedAt": timestamp,
    "status": "QUEUED|IN_PROGRESS|FAILED|SUCCEEDED|CANCELED|TIMED_OUT|REJECTED|REMOVED"
}
```

Description (4)

`executionNumber`

可識別在裝置上之任務執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。

`queuedAt`

任務排入佇列的 epoch 時間，以秒為單位。

`lastUpdatedAt`

任務執行最近一次更新的 epoch 時間，以秒為單位。

`startAt`

任務執行開始的 epoch 時間，以秒為單位。

`status`

任務執行的狀態：`QUEUED`、`IN_PROGRESS`、`FAILED`、`SUCCEEDED`、`CANCELED`、`TIMED_OUT`、`REJECTED` 或 `REMOVED`。

JobExecutionSummaryForJob

JobExecutionSummaryForJob Data Type

`JobExecutionSummaryForJob`物件包含了關於特定工作之工作執行的資訊摘要。

Syntax (5)

```
{
    "executionSummaries": [
        {
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyThing",
            "jobExecutionSummary": {
                "status": "IN_PROGRESS",
                "lastUpdatedAt": 1549395301.389,
                "queuedAt": 1541526002.609,
                "executionNumber": 1
            }
        },
        ...
    ]
}
```

Description (5)

`thingArn`

AWS IoT 實物 ARN。

`jobExecutionSummary`

JobExecutionSummary (p. 390) 物件。

JobExecutionSummaryForThing

JobExecutionSummaryForThing Data Type

`JobExecutionSummaryForThing`物件包含了關於特定物件之工作執行的資訊摘要。

Syntax (6)

```
{
    "executionSummaries": [
        {
            "jobExecutionSummary": {
                "status": "IN_PROGRESS",
                "lastUpdatedAt": 1549395301.389,
                "queuedAt": 1541526002.609,
                "executionNumber": 1
            },
            "jobId": "MyThingJob"
        },
        ...
    ]
}
```

Description (6)

`jobId`

您在建立此工作時所指派的獨特識別符。

`jobExecutionSummary`

JobExecutionSummary (p. 390) 物件。

工作管理和控制 HTTPS 命令

以下命令可透過 HTTPS 通訊協定用於管理和控制應用程式。

AssociateTargetsWithJob

AssociateTargetsWithJob Command

將群組與持續的工作建立關聯。如需更多詳細資訊，請參閱 CreateJob (p. 399)。必須符合以下條件：

- 建立任務時必須將 `targetSelection` 欄位設定為「`CONTINUOUS`」。
- 任務狀態目前必須是「`IN_PROGRESS`」。
- 與工作關聯的目標總數不得超過 100 個。

HTTPS (1)

要求:

```
POST /jobs/jobId/targets

{
    "targets": [ "string" ],
    "comment": "string"
}
```

`jobId`

您在建立此工作時所指派的獨特識別符。

`targets`

定義工作之目標的物件群組 ARN 清單。

`comment`

選用。描述工作為何與目標建立關聯的註解字串。

回應：

```
{
    "jobArn": "string",
    "jobId": "string",
    "description": "string"
}
```

`jobArn`

識別工作的 ARN。

`jobId`

您在建立此工作時所指派的獨特識別符。

`description`

關於工作的簡短文字描述。

CLI (1)

概要：

```
aws iot  associate-targets-with-job \
    --targets <value> \
    --job-id <value> \
    [--comment <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "targets": [
    "string"
  ],
  "jobId": "string",
  "comment": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
targets	清單 成員：TargetArn	定義工作之目標的物件群組 ARN 清單。
TargetArn	字串	

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
comment	字串 長度上限：2028 模式：[^\\p{C}]+	描述任務為何與目標建立關聯的選用字串。

輸出：

```
{
  "jobArn": "string",
  "jobId": "string",
  "description": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
jobArn	字串	識別工作的 ARN。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
描述	字串 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。

MQTT (1)

無.

CancelJob

CancelJob Command

取消工作。

HTTPS (2)

要求:

```
PUT /jobs/jobId/cancel

{
    "force": boolean,
    "comment": "string",
```

```
    "reasonCode": "string"
}
```

`jobId`

您在建立此工作時所指派的獨特識別符。

`force`

[選用] 若為 `true`，則會取消狀態為 `IN_PROGRESS` 和 `QUEUED` 的任務執行。否則，系統只會取消狀態為 `QUEUED` 的任務執行。預設值為 `false`。

> **Warning**
>
> 取消狀態為 `IN_PROGRESS` 的任務，會導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可復原至有效狀態的。

`comment`

[選用] 描述任務取消原因的註解字串。

`reasonCode`

[選用] 說明任務取消原因的原因代碼。如果任務因符合 `abortConfig` 定義的條件而遭到取消，則系統會自動填入此欄位。

回應：

```
{
    "jobArn": "string",
    "jobId": "string",
    "description": "string"
}
```

`jobArn`

工作 ARN。

`jobId`

您在建立此工作時所指派的獨特識別符。

`description`

關於工作的簡短文字描述。

CLI (2)

概要：

```
aws iot  cancel-job \
    --job-id <value> \
    [--force <value>]  \
    [--comment <value>]  \
    [--reasonCode <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "force": boolean,
  "comment": "string"
```

```
}
```

cli-input-json 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
force	布林值	若為 true，則會取消狀態為 QUEUED 和 IN_PROGRESS 的任務。否則，系統只會取消狀態為 QUEUED 的任務。 Warning 取消狀態為 IN_PROGRESS 的任務，會導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可復原至有效狀態的。
comment	字串 長度上限：2028 模式：[^\\p{C}]+	描述任務取消原因的選用字串。
reasonCode	字串 長度上限：128 模式：[\p{Upper}\p{Digit}_]+	說明任務取消原因的選用字串。如果任務因符合 `abortConfig` 定義的條件而遭到取消，則系統會自動填入此欄位。

輸出：

```
{
  "jobArn": "string",
  "jobId": "string",
  "description": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
jobArn	字串	工作 ARN。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

名稱	類型	描述
描述	字串 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。

MQTT (2)

不可用。

CancelJobExecution

CancelJobExecution Command

取消裝置上的任務執行。

HTTPS (3)

要求:

```
PUT /things/thingName/jobs/jobId/cancel

{
    "force": boolean,
    "expectedVersion": "string",
    "statusDetails": {
        "string": "string"
        ...
    }
}
```

`thingName`

任務執行將被取消的物件名稱。

`jobId`

您在建立此工作時所指派的獨特識別符。

`force`

選用。若為 `true`，則可以取消狀態為 `IN_PROGRESS` 或 `QUEUED` 的任務執行。否則，系統只能取消狀態為 `QUEUED` 的任務執行。若您嘗試取消狀態為 `IN_PROGRESS` 的任務執行，且您未將 `force` 設為 `true`，則系統會擲回 `InvalidStateTransitionException`。預設值為 `false`。

> **Warning**
>
> 取消狀態為 `IN_PROGRESS` 的任務，會導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可復原至有效狀態的。

`expectedVersion`

選用。預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果於 AWS IoT 任務服務存放的工作執行版本不符，則更新會被 `VersionConflictException` 錯誤拒絕，並傳回包含目前任務執行狀態資料的 `ErrorResponse`。(這樣就不必另外執行 `DescribeJobExecution` 請求以獲得任務執行狀態資料。)

`statusDetails`

選用。名稱值對的集合，能夠描述任務執行的狀態。

回應：

```
{
}
```

CLI (3)

概要：

```
aws iot  cancel-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--force | --no-force] \
    [--expected-version <value>] \
    [--status-details <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "thingName": "string",
  "force": boolean,
  "expectedVersion": long,
  "statusDetails": {
    "string": "string"
  }
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	欲取消的任務。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	任務執行將被取消的物件名稱。
force	布林值	選用。若為 `true`，則系統會取消狀態為 IN_PROGRESS 或 QUEUED 的任務執行。否則，系統只會取消狀態為 QUEUED 的任務執行。但是，若您嘗試取消狀態為 IN_PROGRESS 的任務執行，且您未將 `--force` 設為 `true`，則系統會擲回 `InvalidStateTransitionException`。預設值為 `false`。 Warning 取消狀態為 IN_PROGRESS 的任

名稱	類型	描述
		務，會導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可復原至有效狀態的。
expectedVersion	長整數 Java 類別：java.lang.Long	選用。預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果於 AWS IoT 任務服務存放的工作執行版本不符，則更新會被 `VersionMismatch` 錯誤拒絕，並傳回包含目前任務執行狀態資料的 `ErrorResponse`。(這樣就不必另外執行 `DescribeJobExecution` 請求以獲得任務執行狀態資料。)
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 `statusDetails` 不會改變。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	

輸出：

無

MQTT (3)

不可用。

CreateJob

CreateJob Command

建立工作。您可以將 Amazon S3 儲存貯體 (`documentSource` 參數) 或請求之本文 (`document` 參數) 中的檔案連結提供給任務文件。

您可以將選用的 `targetSelection` 參數設定為「`CONTINUOUS`」，以讓任務連續。(預設為 `SNAPSHOT`.) 持續工作能夠用於部署或更新新增至群組的裝置，因為此工作會持續運作並在新加入的物件上執行，即使群組裡的物件在工作建立時就已經完成了工作。

任務可以有一個選用 TimeoutConfig，其會設定進行中計時器的值。進行中計時器無法更新，並會套用到任務的所有執行。

Note

目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

以下驗證會在 `CreateJob`API 的引數上執行：

- `targets`引數必須是有效物件或物件群組 ARN 的清單。所有物件和物件群組都必須在您的 AWS 帳戶中。
- `documentSource` 引數必須是連結至任務文件的有效 Amazon S3 URL。Amazon S3 URL 的格式為：`https://s3.amazonaws.com/`*`bucketName`*`/`*`objectName`*。
- 由 `documentSource`引數指定存放文件的 URL 必須是 UTF-8 編碼的 JSON 文件。
- 由於 MQTT 訊息大小限制 (128 KB) 與加密因素，因此工作文件的大小限制為 32 KB。
- `jobId` 在您的 AWS 帳戶中必須是唯一的。

HTTPS (4)

要求:

```
PUT /jobs/jobId

{
    "targets": [ "string" ],
    "document": "string",
    "documentSource": "string",
    "description": "string",
    "presignedUrlConfigData": {
        "roleArn": "string",
        "expiresInSec": "integer"
    },
    "targetSelection": "CONTINUOUS|SNAPSHOT",
    "jobExecutionsRolloutConfig": {
        "exponentialRate": {
           "baseRatePerMinute": integer,
           "incrementFactor": integer,
           "rateIncreaseCriteria": {
              "numberOfNotifiedThings": integer, // Set one or the other
              "numberOfSucceededThings": integer // of these two values.
           },
           "maximumPerMinute": integer
      }
    },
    "abortConfig": {
       "criteriaList": [
          {
             "action": "string",
             "failureType": "string",
             "minNumberOfExecutedThings": integer,
             "thresholdPercentage": integer
          }
       ]
    },
    "timeoutConfig": {
      "inProgressTimeoutInMinutes": long
   }
}
```

`jobId`

工作識別符，這在您 AWS 帳戶中必須是獨特的。我們建議使用 UUID。此處可使用英數字元「-」與「_」。

`targets`

物件或物件群組 ARN 的清單，可定義工作的目標。

`document`

選用。工作文件。

`documentSource`

選用。任務文件的 Amazon S3 連結。

`description`

選用。關於工作的簡短文字描述。

`presignedUrlConfigData`

選用。預先簽章的 Amazon S3 URL 的組態資訊。

`roleArn`

IAM 角色的 ARN，包含了存取 Amazon S3 儲存貯體的許可。此儲存貯體包含了裝置以預先簽章之 Amazon S3 URL 下載的資料。此角色也必須授與 AWS IoT 許可來擔任該角色。如需更多詳細資訊，請參閱 建立工作 (p. 367)。

`expiresInSec`

預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。

`targetSelection`

選用。指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會排定運作於物件上。例如，當物件新增至目標群組，工作就會排定在物件上運作，即使該工作已被原本就在群組中的所有物件完成。

`jobExecutionRolloutConfig`

選用。可讓您建立工作的分階段推展。

`maximumPerMinute`

每分鐘收到要執行的工作之物件最大數量。有效值為 1 到 1000。如果未指定，則預設值為 1000。物件收到任務的實際數量在任一分鐘間隔內可能會較少 (因為系統延遲的關係)，但不會超過指定值。

`exponentialRate`

可讓您為任務建立指數推展速率。

`baseRatePerMinute`

在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。

`incrementFactor`

提高任務推展速率的指數因素。

`rateIncreaseCriteria`

任務啟動增加推展速率的條件。設定 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 的值，但非兩者皆設定。

numberOfNotifiedThings

啟動增加推展速率通知物件數量的閾值。

numberOfSucceededThings

啟動增加推展速率成功物件數量的閾值。

abortConfig

選用。任務中止條件的詳細資訊。

criteriaList

用於定義中止任務規則的中止條件清單。

action

用於初始化任務中止的中止動作類型。

failureType

用於定義初始化任務中止的任務執行失敗類型。

minNumberOfExecutedThings

評估中止規則之前的已執行物件最低數量。

thresholdPercentage

以閾值做為啟動任務中止的已執行物件總數百分比。

timeoutConfig

選用。指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 IN_PROGRESS 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 TIMED_OUT。

Note

目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。

inProgressTimeoutInMinutes

指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。只要此任務的執行狀態指定為 IN_PROGRESS 且已填入此欄位，就會啟動或重新啟動計時器。如果任務執行狀態未在計時器逾期之前設定為終止狀態，或者在填入此欄位的情況下傳送另一個任務執行狀態更新之前，會將狀態設定為 TIMED_OUT。

回應：

```
{
    "jobArn": "string",
    "jobId": "string",
    "description": "string"
}
```

jobArn

工作的 ARN。

jobId

您指派給此工作的獨特識別符。

`description`

關於工作可選用的簡短文字描述。

CLI (4)

概要：

```
aws iot  create-job \
    --job-id <value> \
    --targets <value> \
    [--document-source <value>] \
    [--document <value>] \
    [--description <value>] \
    [--presigned-url-config <value>] \
    [--target-selection <value>] \
    [--job-executions-rollout-config <value>] \
    [--abort-config <value>] \
    [--timeout-config <value>] \
    [--document-parameters <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "targets": [
    "string"
  ],
  "documentSource": "string",
  "document": "string",
  "description": "string",
  "presignedUrlConfig": {
    "roleArn": "string",
    "expiresInSec": long
  },
  "targetSelection": "string",
  "jobExecutionsRolloutConfig": {
      "exponentialRate": {
         "baseRatePerMinute": integer,
          "incrementFactor": integer,
          "rateIncreaseCriteria": {
             "numberOfNotifiedThings": integer, // Set one or the other
             "numberOfSucceededThings": integer // of these two values.
          },
      "maximumPerMinute": integer
      }
    },
  "abortConfig": {
     "criteriaList": [
        {
           "action": "string",
            "failureType": "string",
            "minNumberOfExecutedThings": integer,
            "thresholdPercentage": integer
         }
      ]
    },
  "timeoutConfig": {
      "inProgressTimeoutInMinutes": long
   },
  "documentParameters": {
```

```
    "string": "string"
  }
}
```

cli-input-json 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	工作識別符，這在您 AWS 帳戶中必須是獨特的。我們建議使用 UUID。英數字元「-」和「_」於此處有效。
targets	清單 成員：TargetArn	應收到工作的物件及物件群組清單。
TargetArn	字串	
documentSource	字串 長度上限：1350；下限：1	工作文件的 S3 連結。
document	字串 長度上限：32768	工作文件。
description	字串 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
presignedUrlConfig	PresignedUrlConfig	預先簽章的 S3 URL 之組態資訊。
roleArn	字串 長度上限：2048；下限：20	IAM 角色的 ARN，其會授予自任務資料/更新儲存位置之 Amazon S3 儲存貯體中下載檔案的許可。此角色也必須授予許可讓 AWS IoT 下載檔案。
expiresInSec	長整數 Java 類別：java.lang.Long 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。
targetSelection	字串 列舉：CONTINUOUS | SNAPSHOT	指定任務是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成任務後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，任務也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。

名稱	類型	描述
jobExecutionsRolloutConfig	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。
maximumPerMinute	整數 Java 類別：java.lang.Integer 範圍 - 上限：1000；下限：1	每分鐘通知待定任務的物件之數量上限。此參數可讓您建立分階段推展的任務。
exponentialRate	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
baseRatePerMinute	Java 類別：java.lang.Integer	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。
incrementFactor	Java 類別：java.lang.Double	提高任務推展速率的指數因素。
rateIncreaseCriteria	RateIncreaseCriteria	可讓您定義條件以啟動任務推展的增加速率。擇一設定 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 的值，但非兩者皆設定。
numberOfNotifiedThings	Java 類別：java.lang.Double	啟動增加推展速率通知物件數量的閾值。
numberOfSucceededThings	Java 類別：java.lang.Double	啟動增加推展速率成功物件數量的閾值。
abortConfig	AbortConfig	可讓您建立中止任務的條件。
criteriaList	AbortCriteria	用於定義中止任務規則的中止條件清單。
action	java 類別：java.lang.String (CANCEL)	用於初始化任務中止的中止動作類型。
failureType	java 類別：java.lang.String (FAILED | REJECTED | TIMED_OUT | ALL)	用於定義初始化任務中止的任務執行失敗類型。
minNumberOfExecutedThings	Java 類別：java.lang.Integer)	評估中止規則之前的已執行物件最低數量。
thresholdPercentage	Java 類別：java.lang.Double)	以閾值做為啟動任務中止的已執行物件總數百分比。 AWS IoT 最多支援小數點後兩位數 (例如，10.9 和 10.99，但不支援 10.999)。

名稱	類型	描述
timeoutConfig	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 `TIMED_OUT`。 Note 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。只要此任務的執行狀態指定為 `IN_PROGRESS` 且已填入此欄位，就會啟動或重新啟動計時器。如果任務執行狀態未在計時器逾期之前設定為終止狀態，或者在填入此欄位的情況下傳送另一個任務執行狀態更新之前，會將狀態設定為 `TIMED_OUT`。
documentParameters	對應 金鑰：ParameterKey 值：ParameterValue	工作文件的參數。
ParameterKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
ParameterValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]+	

輸出：

```
{
  "jobArn": "string",
  "jobId": "string",
  "description": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
jobArn	字串	工作 ARN。

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您指派給此工作的獨特識別符。
description	字串 長度上限：2028 模式：[^\\p{C}]+	工作描述。

MQTT (4)

不可用。

DeleteJob

DeleteJob Command

刪除任務及其相關的任務執行。

刪除任務可能會需要時間，根據為任務與各種其他因素所建立的任務執行數量而定。雖然任務被刪除，任務狀態將顯示為「DELETION_IN_PROGRESS」。嘗試刪除或取消狀態已為「DELETION_IN_PROGRESS」的任務將會導致錯誤。

HTTPS (5)

請求語法：

```
DELETE /jobs/jobId?force=force
```

URI 請求參數：

名稱	類型	是否必要？	描述
jobId	JobId	是	欲刪除的任務 ID。
force	ForceFlag	否	(選用) 為 true 時，您可以刪除狀態為「IN_PROGRESS」的任務。否則，您只能刪除結束狀態 (SUCCEEDED 或 CANCELED) 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務，會導致正在執行任務的裝置無法存取任務資訊

名稱	類型	是否必要？	描述
			或更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已刪除任務是可復原至有效狀態的。

錯誤：

`InvalidRequestException`

請求的內容無效。例如，當 UpdateJobExecution 請求包含了無效的狀態詳細資訊，此代碼就會傳回。訊息包含錯誤的詳細資訊。

HTTP 回應代碼：400

`InvalidStateTransitionException`

由於任務執行的目前狀態，而使嘗試變更任務或任務執行狀態的更新無效 (例如，嘗試變更將 SUCCEEDED 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 executionState 欄位。

HTTP 回應代碼：409

`ResourceNotFoundException`

指定的資源不存在。

HTTP 回應代碼：404

`ThrottlingException`

請求頻率超過限制。

HTTP 回應代碼：429

`ServiceUnavailableException`

此服務暫時無法使用。

HTTP 回應代碼：503

CLI (5)

概要：

```
aws iot  delete-job \
    --job-id <value> \
    [--force | --no-force]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "force": boolean
```

```
}
```

cli-input-json 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	欲刪除的任務 ID。
force	布林值	(選用) 為 true 時，您可以刪除狀態為「IN_PROGRESS」的任務。否則，您只能刪除結束狀態 ("SUCCEEDED" 或 "CANCELED") 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務，會導致正在執行任務的裝置無法存取任務資訊或更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已刪除任務是可復原至有效狀態的。

輸出：

無

MQTT (5)

不可用。

DeleteJobExecution

DeleteJobExecution Command

刪除任務執行。

HTTPS (6)

請求語法：

```
DELETE /things/thingName/jobs/jobId/executionNumber/executionNumber?force=force
```

URI 請求參數：

名稱	類型	是否必要？	描述
jobId	JobId	是	執行將被刪除的任務 ID。

名稱	類型	是否必要？	描述
thingName	ThingName	是	任務執行將被刪除的物件名稱。
executionNumber	ExecutionNumber	是	欲刪除的任務執行 ID。
force	ForceFlag	否	(選用) 為 true 時，您可以刪除狀態為「IN_PROGRESS」的任務執行。否則，您只能刪除結束狀態 (SUCCEEDED、FAILED、TIMED_OU 或 CANCELED) 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務執行，會導致裝置無法存取任務資訊或更新任務執行狀態。請謹慎使用並確認裝置可還原為有效狀態。

錯誤：

`InvalidRequestException`

請求的內容無效。例如，當 UpdateJobExecution 請求包含了無效的狀態詳細資訊，此代碼就會傳回。訊息包含錯誤的詳細資訊。

HTTP 回應代碼：400

`InvalidStateTransitionException`

由於任務執行的目前狀態，而使嘗試變更任務執行狀態的更新無效 (例如，嘗試變更將 SUCCEEDED 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 `executionState`欄位。

HTTP 回應代碼：409

`ResourceNotFoundException`

指定的資源不存在。

HTTP 回應代碼：404

`ThrottlingException`

請求頻率超過限制。

HTTP 回應代碼：429

`ServiceUnavailableException`

此服務暫時無法使用。

HTTP 回應代碼：503

CLI (6)

概要：

```
aws iot  delete-job-execution \
    --job-id <value> \
    --thing-name <value> \
    --execution-number <value> \
    [--force | --no-force]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "thingName": "string",
  "executionNumber": long,
  "force": boolean
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	執行將被刪除的任務 ID。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	任務執行將被刪除的物件名稱。
executionNumber	長整數 Java 類別：java.lang.Long	欲刪除的任務執行 ID。
force	布林值	(選用) 為 true 時，您可以刪除狀態為「IN_PROGRESS」的任務執行。否則，您只能刪除結束狀態 (SUCCEEDED、FAILED、TIMED_OUT、REJ 或 CANCELED) 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務執行，會導致裝置無法存取任務資訊或更新任務執行狀態。請謹

名稱	類型	描述
		慎使用並確認裝置可還原為有效狀態。

輸出：

無

MQTT (6)

不可用。

DescribeJob

DescribeJob Command

取得指定工作的詳細資訊。

HTTPS (7)

要求:

```
GET /jobs/jobId
```

`jobId`

您在建立此工作時所指派的獨特識別符。

回應：

```
{
    "documentSource": "string",
    "job": Job
}
```

`documentSource`

任務文件的 Amazon S3 連結。

`job`

任務 (p. 385) 物件。

CLI (7)

概要：

```
aws iot  describe-job \
    --job-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string"
}
```

cli-input-json 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

輸出：

```
{
  "documentSource": "string",
  "job": {
    "jobArn": "string",
    "jobId": "string",
    "targetSelection": "string",
    "status": "string",
    "forceCanceled": boolean,
    "comment": "string",
    "targets": [
      "string"
    ],
    "description": "string",
    "presignedUrlConfig": {
      "roleArn": "string",
      "expiresInSec": long
    },
    "jobExecutionsRolloutConfig": {
        "exponentialRate": {
           "baseRatePerMinute": integer,
           "incrementFactor": integer,
           "rateIncreaseCriteria": {
              "numberOfNotifiedThings": integer, // Set one or the other
              "numberOfSucceededThings": integer // of these two values.
           },
           "maximumPerMinute": integer
      }
    },
    "abortConfig": {
       "criteriaList": [
          {
             "action": "string",
             "failureType": "string",
             "minNumberOfExecutedThings": integer,
             "thresholdPercentage": integer
          }
       ]
    },
    "createdAt": "timestamp",
    "lastUpdatedAt": "timestamp",
    "completedAt": "timestamp",
    "jobProcessDetails": {
      "processingTargets": [
        "string"
      ],
      "numberOfCanceledThings": "integer",
      "numberOfSucceededThings": "integer",
      "numberOfFailedThings": "integer",
      "numberOfRejectedThings": "integer",
      "numberOfQueuedThings": "integer",
      "numberOfInProgressThings": "integer",
```

```
      "numberOfRemovedThings": "integer",
      "numberOfTimedOutThings": "integer"
    },
    "documentParameters": {
      "string": "string"
    },
    "timeoutConfig": {
       "inProgressTimeoutInMinutes": number
    }
  }
}
```

CLI 輸出欄位：

名稱	類型	描述
documentSource	字串 長度上限：1350；下限：1	任務文件的 Amazon S3 連結。
job	任務	有關工作的資訊。
jobArn	字串	以 "arn:aws:iot:region:account:job/jobId" 格式識別任務的 ARN。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
targetSelection	字串 列舉：CONTINUOUS | SNAPSHOT	指定任務是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成任務後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，任務也可能會運作於物件上。例如，當代表裝置的物件加入目標群組，任務就會在裝置上運作，即使該任務已被原本就在群組中的所有物件完成。
status	字串 列舉：IN_PROGRESS | CANCELED | SUCCEEDED	工作的狀態，為 `IN_PROGRESS`、`CANCELED` 或 `SUCCEEDED` 其中一項。
forceCanceled	布林值 Java 類別：java.lang.Boolean	若任務取消且選用的 `force` 參數設為 `true`，將為 `true`。
comment	字串 長度上限：2028 模式：[^\\p{C}]+	如果工作已更新，則會說明更新原因。
targets	清單 成員：TargetArn	應收到任務的 AWS IoT 實物及實物群組的清單。

名稱	類型	描述
TargetArn	字串	
描述	字串 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
presignedUrlConfig	PresignedUrlConfig	預先簽章的 Amazon S3 URL 之組態。
roleArn	字串 長度上限：2048；下限：20	IAM 角色的 ARN，其會授予自任務資料/更新儲存位置之 Amazon S3 儲存貯體中下載檔案的許可。此角色也必須授予許可讓 AWS IoT 任務服務下載檔案。
expiresInSec	長整數 Java 類別：java.lang.Long 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。
jobExecutionsRolloutConfig	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。
maximumPerMinute	整數 Java 類別：java.lang.Integer 範圍 - 上限：1000；下限：1	每分鐘通知待定任務的物件之數量上限。此參數可讓您建立分階段推展的任務。
exponentialRate	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
baseRatePerMinute	Java 類別：java.lang.Integer	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。
incrementFactor	Java 類別：java.lang.Double	提高任務推展速率的指數因素。
rateIncreaseCriteria	RateIncreaseCriteria	可讓您定義條件以啟動任務推展的增加速率。擇一設定 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 的值，但非兩者皆設定。
numberOfNotifiedThings	Java 類別：java.lang.Double	啟動增加推展速率通知物件數量的閾值。
numberOfSucceededThings	Java 類別：java.lang.Double	啟動增加推展速率成功物件數量的閾值。
abortConfig	AbortConfig	可讓您建立中止任務的條件。

名稱	類型	描述
criteriaList	AbortCriteria	用於定義中止任務規則的中止條件清單。
action	java 類別：java.lang.String (CANCEL)	用於初始化任務中止的中止動作類型。
failureType	java 類別：java.lang.String (FAILED | REJECTED | TIMED_OUT | ALL)	用於定義初始化任務中止的任務執行失敗類型。
minNumberOfExecutedThings	Java 類別：java.lang.Integer)	評估中止規則之前的已執行物件最低數量。
thresholdPercentage	Java 類別：java.lang.Double)	以閾值做為啟動任務中止的已執行物件總數百分比。 AWS IoT 最多支援小數點後兩位數 (例如，10.9 和 10.99，但不支援 10.999)。
createdAt	timestamp	任務建立的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務最近一次更新的 epoch 時間，以秒為單位。
completedAt	timestamp	任務完成的 epoch 時間，以秒為單位。
jobProcessDetails	JobProcessDetails	關於工作流程的詳細資訊。
processingTargets	清單 成員：ProcessingTargetName Java 類別：java.util.List	執行工作的裝置。
ProcessingTargetName	字串	
numberOfCanceledThings	整數 Java 類別：java.lang.Integer	取消任務的物件數量。
numberOfSucceededThings	整數 Java 類別：java.lang.Integer	成功完成任務的物件數量。
numberOfFailedThings	整數 Java 類別：java.lang.Integer	執行工作失敗的物件數量。
numberOfRejectedThings	整數 Java 類別：java.lang.Integer	拒絕工作的物件數量。
numberOfQueuedThings	整數 Java 類別：java.lang.Integer	等待執行工作的物件數量。

名稱	類型	描述
numberOfInProgressThings	整數 Java 類別：java.lang.Integer	現正執行工作的物件數量。
numberOfRemovedThings	整數 Java 類別：java.lang.Integer	因為從當成工作目標的群組中被刪除或移除而不再排定執行工作的物件數量。
numberOfTimedOutThings	整數 Java 類別：java.lang.Integer	任務執行狀態為 `TIMED_OUT` 的實物數目。
documentParameters	對應 金鑰：ParameterKey 值：ParameterValue	工作文件的指定參數。
ParameterKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
ParameterValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]+	
timeoutConfig	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在計時器過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 `TIMED_OUT`。 Note 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 `IN_PROGRESS` 狀態的時間超過此間隔時，任務執行就會失敗，並切換到結束 `TIMED_OUT` 狀態。

MQTT (7)

不可用。

DescribeJobExecution

DescribeJobExecution Command

取得工作執行的詳細資訊。任務的執行狀態必須為 `SUCCEEDED`或 `FAILED`。

HTTPS (8)

要求:

```
GET /things/thingName/jobs/jobId?executionNumber=executionNumber
```

`jobId`

您在建立此工作時所指派的獨特識別符。

`thingName`

跟運作工作執行之裝置關聯的物件名稱。

`executionNumber`

選用。用於指定裝置執行之任務的編號。(請參閱 JobExecution (p. 389).) 如果未指定，則會傳回最新的工作執行。

回應：

```
{
    "execution": { JobExecution }
}
```

`execution`

JobExecution (p. 389) 物件。

CLI (8)

概要：

```
aws iot  describe-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "thingName": "string",
  "executionNumber": long
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串	您在建立此工作時所指派的獨特識別符。

名稱	類型	描述
	長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	運作工作執行的物件名稱。
executionNumber	長整數 Java 類別：java.lang.Long	用於指定特定裝置上特定任務執行的字串 (由數字「0」至「9」組成)。

輸出：

```
{
  "execution": {
    "approximateSecondsBeforeTimedOut": "number"
    "jobId": "string",
    "status": "string",
    "forceCanceled": boolean,
    "statusDetails": {
      "detailsMap": {
        "string": "string"
      }
    },
    "thingArn": "string",
    "queuedAt": "timestamp",
    "startedAt": "timestamp",
    "lastUpdatedAt": "timestamp",
    "executionNumber": long,
    "versionNumber": long
  }
}
```

CLI 輸出欄位：

名稱	類型	描述
execution	JobExecution	有關工作執行的資訊。
approximateSecondsBeforeTimedOut	長整數	將任務執行狀態變更為 `TIMED_OUT` 之前要保留狀態的預估秒數。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。實際任務執行逾時可能會比預估持續時間最多晚 60 秒發生。如果任務執行已經達到結束狀態，就不會包含這個值。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

名稱	類型	描述
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | | TIMED_OUT | REJECTED | REMOVED | CANCELED	任務執行的狀態 (IN_PROGRESS、QUEUED、FAILED、SUCC 或 REJECTED)。
forceCanceled	布林值 Java 類別：java.lang.Boolean	若任務執行已取消且選用的 `force` 參數設為 `true`，將為 `true`。
statusDetails	JobExecutionStatusDetails	名稱值對的集合，能夠描述任務執行的狀態。
detailsMap	對應 金鑰：DetailsKey 值：DetailsValue	工作執行狀態。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	
thingArn	字串	運作工作執行的物件 ARN。
queuedAt	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	timestamp	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
executionNumber	長整數 Java 類別：java.lang.Long	用於辨識在此裝置上任務執行的字串 (由數字「0」至「9」組成)。這可於命令中使用，以傳回或更新任務執行的資訊。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。

MQTT (8)

不可用。

GetJobDocument

GetJobDocument Command

取得工作的工作文件。

> **Note**
>
> 在傳回的文件中，預留位置 URL 不會被預先簽章的 Amazon S3 URL 取代。預先簽章的 URL 只有在 AWS IoT 任務服務透過 MQTT 收到請求時才會產生。

HTTPS (9)

要求:

```
GET /jobs/jobId/job-document
```

`jobId`

您在建立此工作時所指派的獨特識別符。

回應：

```
{
    "document": "string"
}
```

`document`

工作文件的內容。

CLI (9)

概要：

```
aws iot  get-job-document \
    --job-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

輸出：

```
{
  "document": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
document	字串 長度上限：32768	工作文件的內容。

MQTT (9)

無.

ListJobExecutionsForJob

ListExecutionsForJob Command

取得工作的工作執行清單。

HTTPS (10)

要求:

```
GET /jobs/jobId/things?status=status&maxResults=maxResults&nextToken=nextToken
```

`jobId`

您在建立此工作時所指派的獨特識別符。

`status`

選用。這是篩選條件，能讓您搜尋具有特定狀態的任務：QUEUED、IN_PROGRESS、SUCCEEDED、FAILED、TIMED_OUT、REJECTED、REMOVED 或 CANCELED。

`maxResults`

選用。每筆請求傳回結果的最大數量。

`nextToken`

選用。用來擷取下一組結果的符記。

回應：

```
{
    "executionSummaries": [ JobExecutionSummary ... ]
}
```

`executionSummaries`

與指定工作 ID 關聯的JobExecutionSummary (p. 390)物件清單。

CLI (10)

概要：

```
aws iot  list-job-executions-for-job \
    --job-id <value> \
    [--status <value>] \
    [--max-results <value>] \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "status": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	任務的狀態。
maxResults	整數 Java 類別：java.lang.Integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
nextToken	字串	用來擷取下一組結果的符記。

輸出：

```
{
  "executionSummaries": [
    {
      "thingArn": "string",
      "jobExecutionSummary": {
        "status": "string",
        "queuedAt": "timestamp",
        "startedAt": "timestamp",
        "lastUpdatedAt": "timestamp",
        "executionNumber": long
      }
    }
  ],
  "nextToken": "string"
```

```
}
```

CLI 輸出欄位：

名稱	類型	描述
executionSummaries	清單 成員：JobExecutionSummaryForJob Java 類別：java.util.List	工作執行摘要清單。
JobExecutionSummaryForJob	JobExecutionSummaryForJob	
thingArn	字串	運作工作執行的物件 ARN。
jobExecutionSummary	JobExecutionSummary	包含關於工作執行的部分資訊。
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	工作執行的狀態。
queuedAt	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	timestamp	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
executionNumber	長整數 Java 類別：java.lang.Long	用於辨識在此裝置上任務執行的字串 (由數字「0」至「9」組成)。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
nextToken	字串	用於下一組結果的符記，如果沒有其他結果則為 null。

MQTT (10)

不可用。

ListJobExecutionsForThing

ListJobExecutionsForThing Command

取得物件的工作執行清單。

HTTPS (11)

要求:

```
GET /things/thingName/jobs?status=status&maxResults=maxResults&nextToken=nextToken
```

`thingName`

列出 `JobExecutions`的物件名稱。

`status`

這是選用篩選條件，能讓您搜尋具有特定狀態的任務：QUEUED、IN_PROGRESS、SUCCEEDED、FAILED、TIMED_OUT、REJECTED、REMOVED 或 CANCELED。

`maxResults`

每筆請求傳回結果的最大數量。

`nextToken`

用於下一組結果的符記，如果沒有其他結果則為 `null`。

回應：

```
{
    "executionSummaries": [ JobExecutionSummary ... ]
}
```

`executionSummaries`

與指定物件關聯之工作執行的 JobExecutionSummary (p. 390)物件清單。

CLI (11)

概要：

```
aws iot  list-job-executions-for-thing \
    --thing-name <value> \
    [--status <value>] \
    [--max-results <value>] \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "thingName": "string",
  "status": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
thingName	字串	物件名稱。

名稱	類型	描述
	長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	讓您搜尋特定狀態之工作的選用篩選條件。
maxResults	整數 Java 類別：java.lang.Integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
nextToken	字串	用來擷取下一組結果的符記。

輸出：

```
{
  "executionSummaries": [
    {
      "jobId": "string",
      "jobExecutionSummary": {
        "status": "string",
        "queuedAt": "timestamp",
        "startedAt": "timestamp",
        "lastUpdatedAt": "timestamp",
        "executionNumber": long
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
executionSummaries	清單 成員：JobExecutionSummaryForThing Java 類別：java.util.List	工作執行摘要清單。
JobExecutionSummaryForThing	JobExecutionSummaryForThing	
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
jobExecutionSummary	JobExecutionSummary	包含關於工作執行的部分資訊。

名稱	類型	描述
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	工作執行的狀態。
queuedAt	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	timestamp	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
executionNumber	長整數 Java 類別：java.lang.Long	用於辨識在此裝置上任務執行的字串 (由數字「0」至「9」組成)。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
nextToken	字串	用於下一組結果的符記，如果沒有其他結果則為 null。

MQTT (11)

不可用。

ListJobs

ListJobs Command

取得您 AWS 帳戶中的工作清單。

HTTPS (12)

要求:

```
GET /jobs?
status=status&targetSelection=targetSelection&thingGroupName=thingGroupName&thingGroupId=thingGroup
```

`status`

選用。這是篩選條件，能讓您搜尋具有特定狀態的任務：IN_PROGRESS、CANCELED 或 SUCCEEDED。

`targetSelection`

選用。這是一種篩選條件，能讓您搜尋具有特定 `targetSelection`值的工作：「CONTINUOUS」或「SNAPSHOT」。

`thingGroupName`

選用。這是一種篩選條件，能讓您搜尋以特定物件群組名稱為目標的工作。

`thingGroupId`

選用。這是一種篩選條件，能讓您搜尋以特定物件群組 ID 為目標的工作。

`maxResults`

選用。每筆請求傳回結果的最大數量。

`nextToken`

選用。用來擷取下一組結果的符記。

回應：

```
{
    "jobs": [ JobSummary ... ],
}
```

`jobs`

JobSummary (p. 389)物件的清單，在您 AWS 帳戶中的每個工作都有一個物件，並且符合指定篩選條件。

CLI (12)

概要：

```
aws iot  list-jobs \
    [--status <value>] \
    [--target-selection <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--thing-group-name <value>] \
    [--thing-group-id <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "status": "string",
  "targetSelection": "string",
  "maxResults": "integer",
  "nextToken": "string",
  "thingGroupName": "string",
  "thingGroupId": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
status	字串 列舉：IN_PROGRESS | CANCELED | SUCCEEDED	讓您搜尋特定狀態之工作的選用篩選條件。
targetSelection	字串 列舉：CONTINUOUS | SNAPSHOT	指定任務是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成任務後視為完成 (SNAPSHOT)。如果是

名稱	類型	敘述
		CONTINUOUS，則在目標中偵測到變更時，任務也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。
maxResults	整數 Java 類別：java.lang.Integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
nextToken	字串	用來擷取下一組結果的符記。
thingGroupName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	限制僅回傳指定群組之工作的篩選條件。
thingGroupId	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	限制僅回傳指定群組之工作的篩選條件。

輸出：

```
{
  "jobs": [
    {
      "jobArn": "string",
      "jobId": "string",
      "thingGroupId": "string",
      "targetSelection": "string",
      "status": "string",
      "createdAt": "timestamp",
      "lastUpdatedAt": "timestamp",
      "completedAt": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
jobs	清單 成員：JobSummary Java 類別：java.util.List	工作清單。
JobSummary	JobSummary	
jobArn	字串	工作 ARN。

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
thingGroupId	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	物件群組的 ID。
targetSelection	字串 列舉：CONTINUOUS | SNAPSHOT	指定任務是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成任務後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，任務也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。
status	字串 列舉：IN_PROGRESS | CANCELED | SUCCEEDED	工作摘要狀態。
createdAt	timestamp	任務建立的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務最近一次更新的 epoch 時間，以秒為單位。
completedAt	timestamp	任務完成的 epoch 時間，以秒為單位。
nextToken	字串	用於下一組結果的符記，如果沒有其他結果則為 null。

MQTT (12)

不可用。

UpdateJob

UpdateJob Command

更新指定任務的支援欄位。`timeoutConfig` 的更新值僅對新進行中的執行生效。目前進行中的執行會繼續執行舊逾時組態。

HTTPS (13)

要求:

```
PATCH /jobs/jobId
{
```

```
   "description": "string",
   "presignedUrlConfig": {
      "expiresInSec": number,
      "roleArn": "string"
   },
   "jobExecutionsRolloutConfig": {
      "exponentialRate": {
         "baseRatePerMinute": number,
         "incrementFactor": number,
         "rateIncreaseCriteria": {
            "numberOfNotifiedThings": number,
            "numberOfSucceededThings": number
         },
      "maximumPerMinute": number
      },
   "abortConfig": {
      "criteriaList": [
         {
            "action": "string",
            "failureType": "string",
            "minNumberOfExecutedThings": number,
            "thresholdPercentage": number
         }
      ]
   },
   "timeoutConfig": {
      "inProgressTimeoutInMinutes": number
   }
}
```

`jobId`

任務識別符，這在您 AWS 帳戶中必須是唯一的。我們建議使用 UUID。此處可使用英數字元「-」與「_」。

`description`

選用。關於工作的簡短文字描述。

`presignedUrlConfigData`

選用。預先簽章的 Amazon S3 URL 的組態資訊。

`roleArn`

IAM 角色的 ARN，包含了存取 Amazon S3 儲存貯體的許可。此儲存貯體包含了裝置以預先簽章之 Amazon S3 URL 下載的資料。此角色也必須授與 AWS IoT 許可來擔任該角色。如需更多詳細資訊，請參閱 建立工作 (p. 367)。

`expiresInSec`

預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。

`jobExecutionRolloutConfig`

選用。可讓您建立工作的分階段推展。

`maximumPerMinute`

每分鐘收到要執行的工作之物件最大數量。有效值為 1 到 1000。如果未指定，則預設值為 1000。物件收到任務的實際數量在任一分鐘間隔內可能會較少 (因為系統延遲的關係)，但不會超過指定值。

`exponentialRate`

可讓您為任務建立指數推展速率。

baseRatePerMinute

在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。

incrementFactor

提高任務推展速率的指數因素。

rateIncreaseCriteria

任務啟動增加推展速率的條件。設定 numberOfNotifiedThings 或 numberOfSucceededThings 的值，但非兩者皆設定。

numberOfNotifiedThings

啟動增加推展速率通知物件數量的閾值。

numberOfSucceededThings

啟動增加推展速率成功物件數量的閾值。

abortConfig

選用。任務中止條件的詳細資訊。

criteriaList

用於定義中止任務規則的中止條件清單。

action

用於初始化任務中止的中止動作類型。

failureType

用於定義初始化任務中止的任務執行失敗類型。

minNumberOfExecutedThings

評估中止規則之前的已執行物件最低數量。

thresholdPercentage

以閥值做為啟動任務中止的已執行物件總數百分比。

timeoutConfig

選用。指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 IN_PROGRESS 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 TIMED_OUT。

inProgressTimeoutInMinutes

指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。只要此任務的執行狀態指定為 IN_PROGRESS 且已填入此欄位，就會啟動或重新啟動計時器。如果任務執行狀態未在計時器逾期之前設定為終止狀態，或者在填入此欄位的情況下傳送另一個任務執行狀態更新之前，會將狀態設定為 TIMED_OUT。

回應：

```
HTTP/1.1 200
```

如果動作成功，則服務會傳回具空 HTTP 內文的 HTTP 200 回應。

CLI (13)

概要：

```
aws iot  update-job \
```

```
    --job-id <value> \
    [--description <value>] \
    [--presigned-url-config <value>] \
    [--job-executions-rollout-config <value>] \
    [--abort-config <value>] \
    [--timeout-config <value>] \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
   "description": "string",
   "presignedUrlConfig": {
      "expiresInSec": number,
      "roleArn": "string"
   },
   "jobExecutionsRolloutConfig": {
      "exponentialRate": {
         "baseRatePerMinute": number,
         "incrementFactor": number,
         "rateIncreaseCriteria": {
            "numberOfNotifiedThings": number,
            "numberOfSucceededThings": number
         }
      },
      "maximumPerMinute": number
   },
   "abortConfig": {
      "criteriaList": [
         {
            "action": "string",
            "failureType": "string",
            "minNumberOfExecutedThings": number,
            "thresholdPercentage": number
         }
      ]
   },
   "timeoutConfig": {
      "inProgressTimeoutInMinutes": number
   }
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	任務識別符，這在您 AWS 帳戶中必須是唯一的。我們建議使用 UUID。英數字元「-」和「_」於此處有效。
描述	字串 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
presignedUrlConfig	PresignedUrlConfig	預先簽章的 S3 URL 之組態資訊。

名稱	類型	描述
roleArn	字串 長度上限：2048；下限：20	IAM 角色的 ARN，其會授予自任務資料/更新儲存位置之 S3 儲存貯體中下載檔案的許可。此角色也必須授予許可讓 AWS IoT 下載檔案。
expiresInSec	長整數 Java 類別：java.lang.Long 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 – 3600。預設值為 3600 秒。當 AWS IoT 任務服務收到任務文件的 MQTT 請求時，就會產生預先簽章的 URL。
jobExecutionsRolloutConfig	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。
maximumPerMinute	整數 Java 類別：java.lang.Integer 範圍 - 上限：1000；下限：1	每分鐘通知待定任務的物件之數量上限。此參數可讓您建立分階段推展的任務。
exponentialRate	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
baseRatePerMinute	Java 類別：java.lang.Integer	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。
incrementFactor	Java 類別：java.lang.Double	提高任務推展速率的指數因素。
rateIncreaseCriteria	RateIncreaseCriteria	可讓您定義條件以啟動任務推展的增加速率。擇一設定 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 的值，但非兩者皆設定。
numberOfNotifiedThings	Java 類別：java.lang.Double	啟動增加推展速率通知物件數量的閾值。
numberOfSucceededThings	Java 類別：java.lang.Double	啟動增加推展速率成功物件數量的閾值。
abortConfig	AbortConfig	可讓您建立中止任務的條件。
criteriaList	AbortCriteria	用於定義中止任務規則的中止條件清單。
action	java 類別：java.lang.String (CANCEL)	用於初始化任務中止的中止動作類型。
failureType	java 類別：java.lang.String (FAILED | REJECTED | TIMED_OUT | ALL)	用於定義初始化任務中止的任務執行失敗類型。
minNumberOfExecutedThings	Java 類別：java.lang.Integer)	評估中止規則之前的已執行物件最低數量。

名稱	類型	描述
thresholdPercentage	Java 類別：java.lang.Double)	以閾值做為啟動任務中止的已執行物件總數百分比。 AWS IoT 最多支援小數點後兩位數 (例如，10.9 和 10.99，但不支援 10.999)。
timeoutConfig	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會將其設為 `TIMED_OUT`。 Note 目前在 AWS GovCloud (US) 區域中不提供任務逾時功能。
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。只要此任務的執行狀態指定為 `IN_PROGRESS` 且已填入此欄位，就會啟動或重新啟動計時器。如果任務執行狀態未在計時器逾期之前設定為終止狀態，或者在填入此欄位的情況下傳送另一個任務執行狀態更新之前，會將狀態設定為 `TIMED_OUT`。
documentParameters	對應 金鑰：ParameterKey 值：ParameterValue	工作文件的參數。
ParameterKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
ParameterValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]+	

輸出：

```
HTTP/1.1 200
```

如果動作成功，則服務會傳回具空 HTTP 內文的 HTTP 200 回應。

MQTT (13)

不可用。

Jobs 裝置 MQTT 和 HTTPS API

裝置 MQTT 和 HTTPS 資料類型

與 AWS IoT 任務服務通訊時會使用以下資料類型搭配 MQTT 和 HTTPS 通訊協定。

JobExecution

JobExecution Data Type

包含關於工作執行的資料。

Syntax (7)

```
{
    "jobId" : "string",
    "thingName" : "string",
    "jobDocument" : "string",
    "status": "QUEUED|IN_PROGRESS|FAILED|SUCCEEDED|CANCELED|TIMED_OUT|REJECTED|
REMOVED",
    "statusDetails": {
        "string": "string"
    },
    "queuedAt" : "timestamp",
    "startedAt" : "timestamp",
    "lastUpdatedAt" : "timestamp",
    "versionNumber" : "number",
    "executionNumber": long
}
```

Description (7)

`jobId`

您在建立此工作時所指派的獨特識別符。

`thingName`

執行工作的物件名稱。

`jobDocument`

工作文件的內容。

`status`

工作執行的狀態。可以是這些狀態其中之一：QUEUED、IN_PROGRESS、FAILED、SUCCEEDED"、CANCELED、TIMED_OUT、REJECTED 或 REMOVED。

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。

`queuedAt`

任務排入佇列的 epoch 時間，以秒為單位。

`startedAt`

任務執行開始的 epoch 時間，以秒為單位。

`lastUpdatedAt`

任務執行最近一次更新的 epoch 時間，以秒為單位。

`versionNumber`

工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。

`executionNumber`

可識別在裝置上之任務執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。

JobExecutionState

JobExecutionState Data Type

包含關於工作執行狀態的資料。

Syntax (8)

```
{
    "status": "QUEUED|IN_PROGRESS|FAILED|SUCCEEDED|CANCELED|TIMED_OUT|REJECTED|
REMOVED",
    "statusDetails": {
        "string": "string"
        ...
    }
    "versionNumber": "number"
}
```

Description (8)

`status`

工作執行的狀態。可以是這些狀態其中之一：QUEUED、IN_PROGRESS、FAILED、SUCCEEDED"、CANCELED、TIMED_OUT、REJECTED 或 REMOVED。

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。

`versionNumber`

工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。

JobExecutionSummary

JobExecutionSummary Data Type

包含關於工作執行的部分資訊。

Syntax (9)

```
{
    "jobId": "string",
    "queuedAt": timestamp,
    "startedAt": timestamp,
    "lastUpdatedAt": timestamp,
    "versionNumber": "number",
    "executionNumber": long
}
```

Description (9)

`jobId`

您在建立此工作時所指派的獨特識別符。

`queuedAt`

任務排入佇列的 epoch 時間，以秒為單位。

`startedAt`

任務執行開始的 epoch 時間，以秒為單位。

`lastUpdatedAt`

任務執行最近一次更新的 epoch 時間，以秒為單位。

`versionNumber`

工作執行的版本。每次 AWS IoT 任務服務從裝置收到更新時，任務執行的版本編號就會增加。

`executionNumber`

可識別在裝置上之任務執行的編號。

ErrorResponse

ErrorResponse Data Type

包含在 AWS IoT 任務服務操作期間所發生的錯誤資訊。

Syntax (10)

```
{
    "code": "ErrorCode",
    "message": "string",
    "clientToken": "string",
    "timestamp": timestamp,
    "executionState": JobExecutionState
}
```

Description (10)

`code`

ErrorCode 可設定為：

InvalidTopic

會將請求傳送至不與任何 API 對應的 AWS IoT 任務命名空間中之主題。

InvalidJson

請求的內容無法解譯為有效的 UTF-8 編碼 JSON。

InvalidRequest

請求的內容無效。例如，當 `UpdateJobExecution`請求包含了無效的狀態詳細資訊，此代碼就會傳回。訊息包含錯誤的詳細資訊。

InvalidStateTransition

由於任務執行的目前狀態，而使嘗試變更任務執行狀態的更新無效 (例如，嘗試變更將 SUCCEEDED 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 `executionState`欄位。

ResourceNotFound

請求主題指定的 `JobExecution` 不存在。

VersionMismatch

請求所指定的預期版本不符合 AWS IoT 任務服務中的任務執行版本。在此情況下，錯誤訊息的本文也會包含 `executionState`欄位。

InternalError

處理請求時發生內部錯誤。

RequestThrottled

請求受到調節。

TerminalStateReached

在處於結束狀態的工作上執行描述工作的命令時發生。

`message`

錯誤訊息字串。

`clientToken`

用於將請求與回覆建立關聯的任意字串。

`timestamp`

Epoch 時間，以秒為單位。

`executionState`

JobExecutionState (p. 437) 物件。只有當 `code`欄位有 `InvalidStateTransition` 或 `VersionMismatch` 值，此欄位才會包含在內。在這些情況下，就不必另外執行 `DescribeJobExecution`請求以獲得工作執行狀態資料。

裝置命令

以下命令可透過 MQTT 和 HTTPS 通訊協定執行。

GetPendingJobExecutions

GetPendingJobExecutions Command

為物件取得所有未處於結束狀態之任務的清單。

MQTT (12)

欲呼叫此 API，請發佈訊息於 `$aws/things/`*`thingName`*`/jobs/get`。

請求承載：

```
{ "clientToken": "string" }
```

`clientToken`

選用。用於將請求和回應建立關聯的用戶端符記。在此輸入任意值，就會在回應中反映出來。

欲收到回應，請訂閱：

- `$aws/things/`*`thingName`*`/jobs/get/accepted` 和
- `$aws/things/`*`thingName`*`/jobs/get/rejected` 或

- `$aws/things/thingName/jobs/get/#` 兩者皆是。

回應承載：

```
{
    "inProgressJobs" : [ JobExecutionSummary ... ],
    "queuedJobs" : [ JobExecutionSummary ... ],
    "timestamp" : 1489096425069,
    "clientToken" : "client-001"
}
```

`inProgressJobs`

狀態為 JobExecutionSummary (p. 437)的 `IN_PROGRESS` 物件清單。

`queuedJobs`

狀態為 JobExecutionSummary (p. 437)的 `QUEUED` 物件清單。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。

`timestamp`

訊息傳送的 epoch 時間，以秒為單位。

HTTPS (12)

要求:

```
GET /things/thingName/jobs
```

`thingName`

與裝置關聯的物件名稱。

回應：

```
{
    "inProgressJobs" : [ JobExecutionSummary ... ],
    "queuedJobs" : [ JobExecutionSummary ... ]
}
```

`inProgressJobs`

JobExecutionSummary (p. 437)物件的清單。

`queuedJobs`

JobExecutionSummary (p. 437)物件的清單。

CLI (12)

概要：

```
aws iot-jobs-data  get-pending-job-executions \
    --thing-name <value>  \
    [--cli-input-json <value>] \
```

```
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "thingName": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。

輸出：

```
{
  "inProgressJobs": [
    {
      "jobId": "string",
      "queuedAt": long,
      "startedAt": long,
      "lastUpdatedAt": long,
      "versionNumber": long,
      "executionNumber": long
    }
  ],
  "queuedJobs": [
    {
      "jobId": "string",
      "queuedAt": long,
      "startedAt": long,
      "lastUpdatedAt": long,
      "versionNumber": long,
      "executionNumber": long
    }
  ]
}
```

CLI 輸出欄位：

名稱	類型	描述
inProgressJobs	清單 成員：JobExecutionSummary Java 類別：java.util.List	狀態為 IN_PROGRESS 的 JobExecutionSummary 物件清單。
JobExecutionSummary	JobExecutionSummary	
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

名稱	類型	描述
queuedAt	長整數	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	長整數 Java 類別：java.lang.Long	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
versionNumber	長整數	工作執行的版本。每次 AWS IoT 任務服務從裝置收到更新時，任務執行的版本編號就會增加。
executionNumber	長整數 Java 類別：java.lang.Long	可識別在裝置上之任務執行的編號。
queuedJobs	清單 成員：JobExecutionSummary Java 類別：java.util.List	狀態為 QUEUED 的 JobExecutionSummary 物件清單。
JobExecutionSummary	JobExecutionSummary	
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
queuedAt	長整數	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	長整數 Java 類別：java.lang.Long	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
versionNumber	長整數	工作執行的版本。每次 AWS IoT 任務服務從裝置收到更新時，任務執行的版本編號就會增加。
executionNumber	長整數 Java 類別：java.lang.Long	可識別在裝置上之任務執行的編號。

StartNextPendingJobExecution

StartNextPendingJobExecution Command

取得並開始物件 (狀態為 IN_PROGRESS 或 QUEUED) 下一個待定的任務執行。

- 狀態為 IN_PROGRESS 的工作執行會先傳回。
- 工作執行會依照其建立的順序傳回。

- 如果下一個待定工作執行為 QUEUED，則其狀態會變更為 IN_PROGRESS，而工作執行狀態詳細資訊也會依指定設定。
- 如果下一個待定工作執行已經為 IN_PROGRESS，則其狀態詳細資訊不會變更。
- 如果沒有工作執行為待定，則回應不會包括 `execution`欄位。
- 您可以選擇透過設定 `stepTimeoutInMinutes` 的屬性來新增步驟計時器。如果您未透過執行 `UpdateJobExecution` 來更新此屬性的值，則步驟計時器逾期時，任務執行將逾時。

MQTT (13)

欲呼叫此 API，請發佈訊息於 `$aws/things/`*`thingName`*`/jobs/start-next`。

請求承載：

```
{
    "statusDetails": {
        "string": "job-execution-state"
        ...
    },
    "stepTimeoutInMinutes": long,
    "clientToken": "string"
}
```

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 `statusDetails` 不會改變。

`stepTimeOutInMinutes`

指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在 `stepTimeoutInMinutes` 欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定此逾時不會影響任務執行逾時，該任務執行逾時可能已在建立任務時加以指定 (使用欄位 `timeoutConfig` 的 `CreateJob`)。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。在此輸入任意值，就會在回應中反映出來。

欲收到回應，請訂閱：

- `$aws/things/`*`thingName`*`/jobs/start-next/accepted` 和
- `$aws/things/`*`thingName`*`/jobs/start-next/rejected` 或
- `$aws/things/`*`thingName`*`/jobs/start-next/#` 兩者皆是。

回應承載：

```
{
    "execution" : JobExecutionData,
    "timestamp" : timestamp,
    "clientToken" : "string"
}
```

`execution`

JobExecution (p. 436) 物件。例如：

```
{
    "execution" : {
```

```
        "jobId" : "022",
        "thingName" : "MyThing",
        "jobDocument" : "< contents of job document >",
        "status" : "IN_PROGRESS",
        "queuedAt" : 1489096123309,
        "lastUpdatedAt" : 1489096123309,
        "versionNumber" : 1,
        "executionNumber" : 1234567890
    },
    "clientToken" : "client-1",
    "timestamp" : 1489088524284,
}
```

`timestamp`

訊息傳送到裝置的 epoch 時間，以毫秒為單位。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。

HTTPS (13)

要求:

```
PUT /things/thingName/jobs/$next
{
    "statusDetails": {
        "string": "string"
        ...
    },
    "stepTimeoutInMinutes": long
}
```

`thingName`

與裝置關聯的物件名稱。

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 `statusDetails` 不會改變。

`stepTimeOutInMinutes`

指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在 `stepTimeoutInMinutes` 欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定此逾時不會影響任務執行逾時，該任務執行逾時可能已在建立任務時加以指定 (使用欄位 `timeoutConfig` 的 `CreateJob`)。

回應：

```
{
    "execution" :  JobExecution
}
```

`execution`

JobExecution (p. 436) 物件。例如：

```
{
    "execution" : {
```

```
        "jobId" : "022",
        "thingName" : "MyThing",
        "jobDocument" : "< contents of job document >",
        "status" : "IN_PROGRESS",
        "queuedAt" : 1489096123309,
        "lastUpdatedAt" : 1489096123309,
        "versionNumber" : 1,
        "executionNumber" : 1234567890
    },
    "clientToken" : "client-1",
    "timestamp" : 1489088524284,
}
```

CLI (13)

概要：

```
aws iot-jobs-data  start-next-pending-job-execution \
    --thing-name <value> \
    {--step-timeout-in-minutes <value>] \
    [--status-details <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "thingName": "string",
  "statusDetails": {
    "string": "string"
  },
    "stepTimeoutInMinutes": long
}
```

`cli-input-json` 欄位：

名稱	類型	描述
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	與裝置關聯的物件名稱。
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 statusDetails 不會改變。
stepTimeOutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在 `stepTimeoutInMinutes` 欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定此逾時不會

名稱	類型	描述
		影響任務執行逾時，該任務執行逾時可能已在建立任務時加以指定 (使用欄位 `timeoutConfig` 的 `CreateJob`)。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	

輸出：

```
{
  "execution": {
    "jobId": "string",
    "thingName": "string",
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "queuedAt": long,
    "startedAt": long,
    "lastUpdatedAt": long,
    "versionNumber": long,
    "executionNumber": long,
    "jobDocument": "string"
  }
}
```

CLI 輸出欄位：

名稱	類型	描述
execution	JobExecution	JobExecution 物件。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT |	工作執行的狀態。可以是這些狀態其中之一：QUEUED、IN_PROGRESS、FAILED、SUCC 或 REMOVED。

名稱	類型	描述
	REJECTED | REMOVED | CANCELED	
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	名稱值對的集合，能夠描述任務執行的狀態。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	
queuedAt	長整數	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	長整數 Java 類別：java.lang.Long	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
executionNumber	長整數 Java 類別：java.lang.Long	可識別在裝置上之任務執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
jobDocument	字串 長度上限：32768	工作文件的內容。

DescribeJobExecution

DescribeJobExecution Command

取得工作執行的詳細資訊。

您可以將 `jobId` 設定為 `$next`，傳回物件 (狀態為 IN_PROGRESS 或 QUEUED) 的下一個待定任務執行。

MQTT (14)

欲呼叫此 API，請發佈訊息於 `$aws/things/thingName/jobs/jobId/get`。

請求承載：

```
{
```

```
    "executionNumber": long,
    "includeJobDocument": boolean,
    "clientToken": "string"
}
```

`thingName`

與裝置關聯的物件名稱。

`jobId`

此工作在建立時被指派的獨特識別符。

或者使用 `$next` 傳回物件 (狀態為 IN_PROGRESS 或 QUEUED) 的下一個待定任務執行。在此情況下，狀態為 IN_PROGRESS 的工作執行會先傳回。工作執行會依照其建立的順序傳回。

`executionNumber`

選用。可識別在裝置上之任務執行的編號。如果未指定，則會傳回最新的工作執行。

`includeJobDocument`

選用。除非設為 `false`，否則回應會包含任務文件。預設值為 `true`。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。在此輸入任意值，就會在回應中反映出來。

欲收到回應，請訂閱：

- `$aws/things/`*`thingName`*`/jobs/`*`jobId`*`/get/accepted` 和
- `$aws/things/`*`thingName`*`/jobs/`*`jobId`*`/get/rejected` 或
- `$aws/things/`*`thingName`*`/jobs/`*`jobId`*`/get/#` 兩者皆是。

回應承載：

```
{
    "execution" : JobExecutionData,
    "timestamp": "timestamp",
    "clientToken": "string"
}
```

`execution`

JobExecution (p. 436) 物件。

`timestamp`

訊息傳送的 epoch 時間，以秒為單位。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。

HTTPS (14)

任務的執行狀態必須為 `QUEUED`或 `IN_PROGRESS`。

要求:

```
GET /things/thingName/jobs/jobId?
executionNumber=executionNumber&includeJobDocument=includeJobDocument
```

`thingName`

與裝置關聯的物件名稱。

`jobId`

此工作在建立時被指派的獨特識別符。

或者使用 `$next` 傳回物件 (狀態為 IN_PROGRESS 或 QUEUED) 的下一個待定任務執行。在此情況下，狀態為 IN_PROGRESS 的工作執行會先傳回。工作執行會依照其建立的順序傳回。

`includeJobDocument`

選用。除非設為 `false`，否則回應會包含任務文件。預設值為 `true`。

`executionNumber`

選用。可識別在裝置上之任務執行的編號。如果未指定，則會傳回最新的工作執行。

回應：

```
{
    "execution" : JobExecution,
}
```

`execution`

JobExecution (p. 436) 物件。

CLI (14)

任務的執行狀態必須為 `QUEUED`或 `IN_PROGRESS`。

概要：

```
aws iot-jobs-data  describe-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--include-job-document | --no-include-job-document] \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "thingName": "string",
  "includeJobDocument": boolean,
  "executionNumber": long
}
```

`cli-input-json` 欄位：

名稱	類型	描述
jobId	字串 模式：[a-zA-Z0-9_-]+|^$next	建立此任務時，指派給該任務的唯一識別符，或是 `$next`，以傳回物件 (狀態為 IN_PROGRESS 或 QUEUED) 的下一個待定任

名稱	類型	描述
		務執行。在此情況下，狀態為 IN_PROGRESS 的工作執行會先傳回。工作執行會依照其建立的順序傳回。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	跟運作工作執行之裝置關聯的物件名稱。
includeJobDocument	布林值 Java 類別：java.lang.Boolean	選用。除非設為 false，否則回應會包含任務文件。預設為 true。
executionNumber	長整數 Java 類別：java.lang.Long	選用。可識別在裝置上之任務執行的編號。如果未指定，則會傳回最新的工作執行。

輸出：

```
{
  "execution": {
    "jobId": "string",
    "thingName": "string",
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "queuedAt": long,
    "startedAt": long,
    "lastUpdatedAt": long,
    "versionNumber": long,
    "executionNumber": long,
    "jobDocument": "string"
  }
}
```

CLI 輸出欄位：

名稱	類型	描述
execution	JobExecution	包含關於工作執行的資料。
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。
status	字串	工作執行的狀態。可以是這些狀態其中之一：

名稱	類型	描述
	列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	QUEUED、IN_PROGRESS、FAILED、SUCC 或 REMOVED。
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	名稱值對的集合，能夠描述任務執行的狀態。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	
queuedAt	長整數	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	長整數 Java 類別：java.lang.Long	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
executionNumber	長整數 Java 類別：java.lang.Long	可識別在裝置上之任務執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
jobDocument	字串 長度上限：32768	工作文件的內容。

UpdateJobExecution

UpdateJobExecution Command

更新工作執行的狀態。您可以選擇透過設定 `stepTimeoutInMinutes` 的屬性來新增步驟計時器。如果您未透過再次執行 `UpdateJobExecution` 來更新此屬性的值，則步驟計時器逾期時，任務執行將逾時。

MQTT (15)

欲呼叫此 API，請發佈訊息於 `$aws/things/thingName/jobs/jobId/update`。

請求承載：

```
{
    "status": "job-execution-state",
    "statusDetails": {
        "string": "string"
        ...
    },
    "expectedVersion": "number",
    "executionNumber": long,
    "includeJobExecutionState": boolean,
    "includeJobDocument": boolean,
    "stepTimeoutInMinutes": long,
    "clientToken": "string"
}
```

`status`

新的工作執行狀態 (IN_PROGRESS、FAILED、SUCCEEDED 或 REJECTED)。這必須在每次更新時指定。

`statusDetails`

名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 `statusDetails` 不會改變。

`expectedVersion`

預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果於 AWS IoT 任務服務存放的工作執行版本不符，則更新會被 `VersionMismatch` 錯誤拒絕，並傳回包含目前任務執行狀態資料的 ErrorResponse (p. 438)。(這樣就不必另外執行 `DescribeJobExecution` 請求以獲得任務執行狀態資料。)

`executionNumber`

選用。可識別在裝置上之任務執行的編號。如果未指定，則會使用最新的工作執行。

`includeJobExecutionState`

選用。若被包括在內並設定為 `true`，則回應會包含 `JobExecutionState` 欄位。預設值為 `false`。

`includeJobDocument`

選用。若被包括在內並設定為 `true`，則回應會包含 `JobDocument`。預設值為 `false`。

`stepTimeoutInMinutes`

指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (同樣地，藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在此欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定或重設此逾時不會影響任務執行逾時，該任務執行逾時可能已在建立任務時加以指定 (透過使用 `timeoutConfig` 和 `CreateJob`)。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。在此輸入任意值，就會在回應中反映出來。

欲收到回應，請訂閱：

- `$aws/things/thingName/jobs/jobId/update/accepted` 和
- `$aws/things/thingName/jobs/jobId/update/rejected` 或
- `$aws/things/thingName/jobs/jobId/update/#` 兩者皆是。

回應承載：

```
{
```

```
    "executionState": JobExecutionState,
    "jobDocument": "string",
    "timestamp": timestamp,
    "clientToken": "string"
}
```

`executionState`

JobExecutionState (p. 437) 物件。

`jobDocument`

工作文件 (p. 363)物件。

`timestamp`

訊息傳送的 epoch 時間，以秒為單位。

`clientToken`

用於將請求和回應建立關聯的用戶端符記。

HTTPS (15)

要求:

```
POST /things/thingName/jobs/jobId
{
    "status": "job-execution-state",
    "statusDetails": {
        "string": "string"
        ...
    },
    "expectedVersion": "number",
    "includeJobExecutionState": boolean,
    "includeJobDocument": boolean,
    "stepTimeoutInMinutes": long,
    "executionNumber": long
}
```

`thingName`

與裝置關聯的物件名稱。

`jobId`

此工作在建立時被指派的獨特識別符。

`status`

新的工作執行狀態 (IN_PROGRESS、FAILED、SUCCEEDED 或 REJECTED)。這必須在每次更新時指定。

`statusDetails`

選用。名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 `statusDetails` 不會改變。

`expectedVersion`

選用。預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果於 AWS IoT 任務服務存放的工作執行版本不符，則更新會被 `VersionMismatch` 錯誤拒絕，並傳回包含目前任務執行狀態資料的 ErrorResponse (p. 438)。(這樣就不必另外執行 `DescribeJobExecution` 請求以獲得任務執行狀態資料。)

`includeJobExecutionState`

選用。若被包括在內並設定為 `true`，則回應會包含 JobExecutionState 資料。預設值為 `false`。

`includeJobDocument`

選用。當設定為 `true`，回應就會包含工作文件。預設值為 `false`。

`stepTimeoutInMinutes`

指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (同樣地，藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在此欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定或重設此逾時不會影響任務執行逾時，該任務執行逾時可能已在建立任務時加以指定 (透過使用 `timeoutConfig` 和 `CreateJob`)。

`executionNumber`

選用。可識別在裝置上之任務執行的編號。

回應：

```
{
    "executionState": JobExecutionState,
    "jobDocument": "string"
}
```

`executionState`

JobExecutionState (p. 437) 物件。

`jobDocument`

工作文件 (p. 363)的內容。

CLI (15)

概要：

```
aws iot-jobs-data  update-job-execution \
    --job-id <value> \
    --thing-name <value> \
    --status <value> \
    [--status-details <value>] \
    [--expected-version <value>] \
    [--include-job-execution-state | --no-include-job-execution-state] \
    [--include-job-document | --no-include-job-document] \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
    [--step-timeout-in-minutes <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "jobId": "string",
  "thingName": "string",
  "status": "string",
  "statusDetails": {
    "string": "string"
  },
  "stepTimeoutInMinutes": number,
  "expectedVersion": long,
  "includeJobExecutionState": boolean,
```

```
  "includeJobDocument": boolean,
  "executionNumber": long
}
```

cli-input-json 欄位：

名稱	類型	描述
jobId	字串 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	此工作在建立時被指派的獨特識別符。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	與裝置關聯的物件名稱。
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	新的工作執行狀態 (IN_PROGRESS、FAILED、SUCCEEDED 或 REJECTED)。這必須在每次更新時指定。
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	選用。名稱值對的集合，能夠描述任務執行的狀態。如果未指定，則 statusDetails 不會改變。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	
stepTimeoutInMinutes 長整數 指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (同樣地，藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在此欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會設為 `TIMED_OUT`。設定或重設此逾時不會影響任務執行逾時，該任務執行逾時可能		

名稱	類型	描述
已在建立任務時加以指定 (透過使用 `timeoutConfig` 和 `CreateJob`)。		
expectedVersion	長整數 Java 類別：java.lang.Long	選用。預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果存放於 AWS IoT 任務服務的任務執行版本不符，則更新會被 VersionMismatch 錯誤拒絕，並傳回包含目前任務執行狀態資料的 ErrorResponse。(這樣就不必另外執行 DescribeJobExecution 請求以獲得任務執行狀態資料。)
includeJobExecutionState	布林值 Java 類別：java.lang.Boolean	選用。若被包括在內並設定為 true，則回應會包含 JobExecutionState 資料。預設值為 false。
includeJobDocument	布林值 Java 類別：java.lang.Boolean	選用。當設定為 True，回應就會包含工作文件。預設值為 false。
executionNumber	長整數 Java 類別：java.lang.Long	選用。可識別在裝置上之任務執行的編號。

輸出：

```
{
  "executionState": {
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "versionNumber": long
  },
  "jobDocument": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
executionState	JobExecutionState	JobExecutionState 物件。
status	字串 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED	工作執行的狀態。可以是這些狀態其中之一：QUEUED、IN_PROGRESS、FAILED、SUCC 或 REMOVED。

名稱	類型	描述
statusDetails	對應 金鑰：DetailsKey 值：DetailsValue	名稱值對的集合，能夠描述任務執行的狀態。
DetailsKey	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
DetailsValue	字串 長度上限：1024；下限：1 模式：[^\\p{C}]*+	
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
jobDocument	字串 長度上限：32768	任務文件的內容。

JobExecutionsChanged

JobExecutionsChanged Message

每當物件的待定工作執行清單新增或移除工作執行時就會傳送。

MQTT (16)

主題：`$aws/things/thingName/jobs/notify`

訊息承載：

```
{
    "jobs" : {
        "JobExecutionState": [ JobExecutionSummary (p. 390) ... ]
    },
    "timestamp": timestamp,
}
```

HTTPS (16)

不可用。

CLI (16)

無.

NextJobExecutionChanged

NextJobExecutionChanged Message

每當物件之待定工作清單上的下一個工作執行變更就會傳送，亦即定義 DescribeJobExecution (p. 447)時加上 jobId `$next`。當下一個工作的執行詳細資訊變更，此訊息並

不會傳送，只有在加上 jobId `DescribeJobExecution`之 `$next` 傳回的下一個工作變更時才會傳送。考量狀態為 QUEUED 的工作執行 J1 與 J2。J1 是下一個待定工作執行。如果 J2 的狀態變更為 IN_PROGRESS 而 J1 的狀態保持不變，則會傳送此通知並包含 J2 的詳細資訊。

MQTT (17)

主題：`$aws/things/thingName/jobs/notify-next`

訊息承載：

```
{
    "execution" : JobExecution (p. 389),
    "timestamp": timestamp,
}
```

HTTPS (17)

不可用。

CLI (17)

不可用。

任務推展和中止組態

您可以使用變數推展速率來部署 AWS IoT 任務，該速率必須滿足特定條件和閾值。如果失敗任務的數量符合一組條件，則您也可以中止任務推展。這些推展組態可讓您更精準地控制任務的範圍。任務推展速率條件是在建立任務時，透過 `JobExecutionsRolloutConfig` 物件加以設定。任務中止條件是在建立任務時，透過 `AbortConfig` 物件加以設定。

使用任務推展速率

您可以在執行 `CreateJob` API 時，透過設定 `JobExecutionsRolloutConfig` 物件的 `ExponentialRolloutRate` 屬性來設定任務推展速率。以下範例使用 `exponentialRate` 參數來設定變數推展速率。

```
{
...
  "jobExecutionsRolloutConfig": {
      "exponentialRate": {
         "baseRatePerMinute": 50,
         "incrementFactor": 2,
         "rateIncreaseCriteria": {
            "numberOfNotifiedThings": 1000, // Set one or the other
            "numberOfSucceededThings": 1000 // of these two values.
         },
         "maximumPerMinute": 1000
...
}
```

`baseRatePerMinute` 參數指定達到 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 閾值之前的任務執行速率。

`incrementFactor` 參數指定推展速率在達到 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 閾值之後提高的指數因數。

`rateIncreaseCriteria` 參數是指定 `numberOfNotifiedThings` 或 `numberOfSucceededThings` 閾值的物件。

`maximumPerMinute` 參數指定可以發生任務執行的速率上限。有效值範圍從 1 到 1000。當您傳遞 `ExponentialRate` 物件時，此為必要參數。在變數速率推展中，此值會建立任務推展速率的上限。

使用上述組態的任務推展將以每分鐘 50 個任務執行的速率開始。系統會以該速率繼續，直到 1000 個物件已收到任務執行通知 (如果已指定 `numberOfNotifiedThings` 的值)，或已發生 1000 次成功的任務執行 (如果已指定 `numberOfSucceededThings` 的值) 為止。

下表說明推展如何在前四個增量中繼續進行。

每分鐘推展速率	50	100	200	400
已通知裝置或成功執行的數量	1000	2000	3000	4000

以下組態設定靜態推展速率。

```
{
...
  "jobExecutionsRolloutConfig": {
         "maximumPerMinute": 1000
...
}
```

`maximumPerMinute` 參數指定可以發生任務執行的速率上限。有效值範圍從 1 到 1000。此為選用參數。如果您未指定值，則系統會使用預設值 1000。

使用任務推展中止組態

當您執行 `CreateJob` API 時，您可以設定選用的 `AbortConfig` 物件來設定任務中止條件。本節說明以下範例組態對發生多次失敗執行之任務推展的影響。

```
   "abortConfig": {
      "criteriaList": [
         {
            "action": "CANCEL",
            "failureType": "FAILED",
            "minNumberOfExecutedThings": 100,
            "thresholdPercentage": 20
         },
         {
            "action": "CANCEL",
            "failureType": "TIMED_OUT",
            "minNumberOfExecutedThings": 200,
            "thresholdPercentage": 50
         }
      ]
    }
```

`action` 參數會指定符合中止條件時所要採取的動作。此為必要參數，且 `CANCEL` 是唯一有效的值。

`failureType` 參數會指定應觸發任務中止的失敗類型。有效值為 `FAILED`、`REJECTED`、`TIMED_OUT`、`ALL`。

`minNumberOfExecutedThings` 參數指定在服務檢查是否符合任務中止條件之前，必須發生的任務執行完成數量。在此範例中，AWS IoT 不會檢查是否在至少 100 個裝置完成任務執行之前發生任務中止。

`thresholdPercentage` 參數指定啟動任務中止的已執行物件總數。在此範例中，如果完成 100 次執行後，至少有 20% 的已完成執行以任何方式失敗，AWS IoT 將啟動任務中止並取消任務推展。

> **Note**
>
> 刪除任務執行會影響已完成執行總數的計算值。當任務中止時，服務會建立自動化的 `comment` 和 `reasonCode`，以區分使用者驅動的取消和任務中止的取消。

工作限制

如需工作限制資訊，請參閱 AWS IoT 中的「AWS General Reference 工作限制」。

AWS IoT 安全通道

在遠端網站的受限制防火牆後面部署裝置時，您需要一種方法來存取這些裝置，以進行故障診斷、設定更新和其他操作任務。安全通道可協助客戶透過由 AWS IoT 管理的安全連線，與遠端裝置建立雙向通訊。安全通道不需要更新現有的輸入防火牆規則，因此您可以在遠端網站保留防火牆規則所提供的相同安全層級。

例如，位於幾百英里外工廠的感應器裝置在測量工廠溫度時遇到麻煩。您可以使用安全通道，開啟並快速啟動該感應器裝置的工作階段。找出問題之後 (例如，不當的組態檔)，您可以重設檔案並透過相同的工作階段重新啟動感應器裝置。相較於傳統的故障診斷 (例如，派遣技術人員到工廠調查感應器裝置)，安全通道可減少事故回應和回復時間，以及操作成本。

安全通道概念

目標裝置

您要存取的遠端裝置。

目標應用程式

在目標裝置上執行的應用程式。例如，目標應用程序可以是一個 SSH 協助程式，用於使用安全通道建立 SSH 工作階段。

來源裝置

此裝置通常是筆記型電腦或桌上型電腦，操作人員用來啟動目標裝置的工作階段。

通道

一種通過 AWS IoT 的邏輯途徑，可在來源裝置與目標裝置之間進行雙向通訊。

用戶端存取字符 (CAT)

建立新通道時，由安全通道產生的一對字符。CAT 是由來源和目標裝置用來連線至安全通道服務。

裝置代理程式

連線至 AWS IoT 裝置閘道並透過 MQTT 接聽新通道通知的 IoT 應用程式。

本機代理

在來源和目標裝置上執行，並在安全通道服務與裝置應用程式之間轉送資料串流的軟體代理。本機代理可以在源模式或目標模式下執行。如需更多詳細資訊，請參閱 本機代理 (p. 466)。

安全通道教學課程

在本教學課程中，您會開啟一個通道，並將其用來啟動遠端裝置的 SSH 工作階段。遠端裝置位於防火牆後面，這些防火牆會封鎖所有輸入流量，使得 SSH 無法直接進入裝置。開始之前，請確定您了解如何在 AWS IoT 登錄中註冊裝置，並將裝置連線至 AWS IoT 裝置閘道。

先決條件

- 遠端裝置所在的防火牆必須允許連接埠 443 上的輸出流量。
- 你已在 AWS IoT 登錄中建立了一個名為 `RemoteDeviceA` 的 IoT 物件。
- 您有一個 IoT 裝置代理程式在連線至 AWS IoT 裝置閘道，並使用 MQTT 主題訂閱進行設定的遠端裝置上執行。本教學課程包括一個程式碼片段，顯示如何實作代理程式。如需詳細資訊，請參閱 IoT Agent Snippet (p. 467)。
- 您必須具有一個在遠端裝置上執行的 SSH 協助程式。

- 您已從 GitHub 下載了本機代理原始碼，並為您選擇的平台建置它。在本教學課程中，我們將建置的本機代理可執行檔案稱作 `localproxy`。

開啟通道

1. 開啟 AWS IoT 主控台。在導覽窗格中，選擇 Manage (管理)，然後選擇 Tunnels (通道)。
2. 選擇 Open a tunnel (開啟通道)。
3. 在 Open a new tunnel (開啟新通道) 頁面上，輸入通道的選擇性描述。
4. 在 Choose a thing to open a tunnel for (選擇要開啟通道的物件) 底下，選擇 RemoteDeviceA。
5. 在 Service (服務) 底下，輸入 **`SSH`**，然後選擇 Open New (開啟新的)。如需詳細資訊，請參閱「IoT Agent Snippet (p. 467)」
6. 在 New tunnel opened (開啟新通道) 頁面中，下載您的來源和目標存取字符，並將它們儲存到您的筆記型電腦。這是你可以擷取字符的唯一時間。
7. 請選擇 Done (完成)。

如果您在呼叫 `OpenTunnel` 時設定目標，安全通道服務會透過 MQTT 和保留的 MQTT 主題 (`$aws/things/RemoteDeviceA/tunnels/notify`)，將目標用戶端存取字符交付給遠端裝置。如需詳細資訊，請參閱 MQTT 保留主題。在收到 MQTT 訊息後，遠端裝置上的 IoT 代理程式會以目標模式啟動本機 Proxy。如果您想要透過其他方法將目標用戶端存取字符交付給遠端裝置，則可以省略目標組態。如需詳細資訊，請參閱 設定遠端裝置 (p. 468)。

啟動本機代理

在筆記本電腦上開啟終端機、複製來源用戶端存取字符，並使用它在來源模式下啟動本機代理。在下列命令中，本機代理會設定為在連接埠 5555 上接聽新連線。

./localproxy -r us-east-1 -s 5555 -t ***`<source-client-access-token>`***

> **Note**
>
> 此命令中的 AWS 區域必須與建立通道的 AWS 區域相同。

-r

指定建立通道的 AWS 區域。

-s

指定代理應該連線的連接埠。

-t

指定用戶端字符文字。

> **Note**
>
> 如果您看到下列錯誤，請參閱 GitHub 以取得設定 CA 路徑的詳細資訊。
> `Could not perform SSL handshake with proxy server: certificate verify failed`

啟動 SSH 工作階段

開啟另一個終端機並使用以下命令，透過連線至連接埠 5555 上的本機代理來啟動新的 SSH 工作階段。

ssh ***`<username>`***@localhost -p 5555

系統可能會提示您輸入 SSH 工作階段的密碼。當您完成了 SSH 工作階段時，請輸入 **exit** 以關閉工作階段。

關閉通道

1. 開啟 AWS IoT 主控台。
2. 選擇通道，然後從 Actions (動作) 中選擇 Close (關閉)。關閉通道會導致兩個本機代理執行個體關閉。

安全通道生命週期

通道可以有下列其中一種狀態：

- OPEN
- CLOSED

連線可以具有下列其中一種狀態：

- CONNECTED
- DISCONNECTED

當您開啟通道時，它的狀態為 OPEN。通道的來源和目標連線狀態會設定為 DISCONNECTED。當裝置 (來源或目標) 連線至通道時，對應的連線狀態會變更為 CONNECTED，而通道狀態仍會保持 OPEN。當裝置與通道中斷連線時，對應的連線狀態會變回 DISCONNECTED。只要通道仍然保持 OPEN，裝置就可以重複連線至通道，以及與通道中斷連線。

當您呼叫 CloseTunnel 或通道保持 OPEN 的時間超過 MaxLifetimeTimeout 值時，通道的狀態就會變成 CLOSED。您可以在呼叫 OpenTunnel 時設定 MaxLifetimeTimeout。如果您未指定一值，則 MaxLifetimeTimeout 會預設為 12 小時。

在通道的狀態為 CLOSED 之後，它可能會在 AWS IoT 主控台中顯示至少三個小時，然後才會加以刪除。通道顯示時，您可以呼叫 DescribeTunnel 和 ListTunnels 來檢視通道中繼資料。狀態為 CLOSED 時，無法重新開啟通道。安全通道服務會拒絕嘗試連線至 CLOSED 通道。

控制 Kibana 的存取

安全通道服務提供可用於 IAM 許可政策的下列服務特定動作、資源和條件內容金鑰。

通道存取先決條件

- 了解如何使用 IAM 政策保護 AWS 資源。
- 了解如何建立和評估 IAM 條件。
- 了解如何使用資源標籤保護 AWS 資源。

iot:OpenTunnel

iot:OpenTunnel 政策動作可授與委託人呼叫 OpenTunnel 的許可。您必須在 IAM 政策陳述式的 Resource 元素中指定萬用字元通道 ARN arn:aws:iot:*<aws-region>*:*<aws-account-id>*:tunnel/*。您可以在 IAM 政策陳述式的 Resource 元素中指定 ARN (arn:aws:iot:*<aws-region>*:*<aws-account-id>*:thing/ *<thing-name>*)，以管理特定 IoT 物件的 OpenTunnel 許可。

例如，以下政策陳述式允許您開啟一個通道，前往名為 `TestDevice` 的 IoT 物件。

```
{
    "Effect": "Allow",
    "Action": "iot:OpenTunnel",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*",
        "arn:aws:iot:<aws-region>:<aws-account-id>:thing/TestDevice"
    ]
}
```

`iot:OpenTunnel` 政策動作支援下列條件金鑰：

- `iot:ThingGroupArn`
- `iot:TunnelDestinationService`
- `aws:RequestTag/`*`<tag-key>`*
- `aws:TagKeys`

以下策略陳述式允許您對物件開啟通道，前提是該物件屬於名稱開頭為 `TestGroup` 的物件群組，並且通道上設定的目標服務是 SSH。

```
{
    "Effect": "Allow",
    "Action": "iot:OpenTunnel",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*"
    ],
    "Condition": {
        "ForAnyValue:StringLike": {
            "iot:ThingGroupArn": [
                "arn:aws:iot:<aws-region>:<aws-account-id>:thinggroup/TestGroup*"
            ]
        },
        "ForAllValues:StringEquals": {
            "iot:TunnelDestinationService": [
                "SSH"
            ]
        }
    }
}
```

您也可以使用資源標籤來控制開啟通道的許可。例如，如果標籤索引鍵 `Owner` 的值為 `Admin` 且未指定其他標籤，則下列政策陳述式允許開啟通道。

```
{
    "Effect": "Allow",
    "Action": "iot:OpenTunnel",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/Owner": "Admin"
        },
        "ForAllValues:StringEquals": {
            "aws:TagKeys": "Owner"
        }
    }
}
```

iot:DescribeTunnel

iot:DescribeTunnel 政策動作可授與委託人呼叫 DescribeTunnel 的許可。您可以指定完整的通道 ARN (例如，arn:aws:iot:<aws-region>: <aws-account-id>:tunnel/<tunnel-id>) 或在 IAM 政策陳述式的 Resource 元素中使用萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。

iot:DescribeTunnel 政策動作支援下列條件金鑰：

- aws:ResourceTag/<tag-key>

如果使用值為 Admin 的金鑰 Owner 標記所請求的通道，下列政策陳述式可讓您呼叫 DescribeTunnel。

```
{
    "Effect": "Allow",
    "Action": "iot:DescribeTunnel",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/Owner": "Admin"
        }
    }
}
```

iot:ListTunnels

iot:ListTunnels 政策動作可授與委託人呼叫 ListTunnels 的許可。您必須在 IAM 政策陳述式的 Resource 元素中指定萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。若要管理所選 IoT 物件的 ListTunnels 許可，您也可以在 IAM 政策陳述式的 Resource 元素中指定 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:thing/<thing-name>)。

以下策略陳述式允許您列出名為 TestDevice 之物件的通道。

```
{
    "Effect": "Allow",
    "Action": "iot:ListTunnels",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*",
        "arn:aws:iot:<aws-region>:<aws-account-id>:thing/TestDevice"
    ]
}
```

iot:ListTagsForResource

iot:ListTagsForResource 政策動作可授與委託人呼叫 ListTagsForResource 的許可。您可以指定完整的通道 ARN (arn:aws:iot:<aws-region>: <aws-account-id>:tunnel/<tunnel-id>) 或在 IAM 政策陳述式的 Resource 元素中使用萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。

iot:CloseTunnel

iot:CloseTunnel 政策動作可授與委託人呼叫 CloseTunnel 的許可。您可以指定完整的通道 ARN (arn:aws:iot:<aws-region>: <aws-account-id>:tunnel/<tunnel-id>) 或在 IAM 政策陳

述式的 Resource 元素中使用萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。

iot:CloseTunnel 政策動作支援下列條件金鑰：

- iot:Delete
- aws:ResourceTag/<tag-key>

如果請求的 Delete 參數為 false，並使用值為 QATeam 的金鑰 Owner 標記所請求的通道，下列政策陳述式可讓您呼叫 CloseTunnel。

```
{
    "Effect": "Allow",
    "Action": "iot:CloseTunnel",
    "Resource": [
        "arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*"
    ],
    "Condition": {
        "Bool": {
            "iot:Delete": "false"
        },
        "StringEquals": {
            "aws:ResourceTag/Owner": "QATeam"
        }
    }
}
```

iot:TagResource

iot:TagResource 政策動作可授與委託人呼叫 TagResource 的許可。您可以指定完整的通道 ARN (arn:aws:iot:<aws-region>: <aws-account-id>:tunnel/<tunnel-id>) 或在 IAM 政策陳述式的 Resource 元素中使用萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。

iot:UntagResource

iot:UntagResource 政策動作可授與委託人呼叫 UntagResource 的許可。您可以指定完整的通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/<tunnel-id>) 或在 IAM 政策陳述式的 Resource 元素中使用萬用字元通道 ARN (arn:aws:iot:<aws-region>:<aws-account-id>:tunnel/*)。

如需 AWS IoT 安全的詳細資訊，請參閱適用於 AWS IoT 的 Identity and Access Management (p. 179)。

本機代理

本機代理是充當傳入 TCP 連線之收件人或寄件者的程序。它透過 WebSocket 安全連線傳輸裝置應用程式透過安全通道服務所傳送的資料。您可以從 GitHub下載本機代理原始碼。本機代理可以在兩種模式下執行：source 或 destination。在來源模式中，本機代理會在與啟動 TCP 連線的用戶端應用程式相同的裝置或網路上執行。在目標模式中，本機代理會與目標應用程式一起在遠端裝置上執行。目前，單一通道一次只能支援一個 TCP 連線。

本機代理會先建立安全通道服務的連線。當您啟動本機代理時，請使用 -r 引數來指定開啟通道的 AWS 區域。使用 -t 引數傳遞從 OpenTunnel 傳回的來源或目標用戶端存取字符。使用相同用戶端存取字符值的兩個本機代理不能同時連線。

建立 WebSocket 連線之後，本機代理會執行來源模式或目標模式行為，取決於其組態。

根據預設，如果發生任何 I/O 錯誤，或 WebSocket 連線意外關閉，本機代理會嘗試重新連線至安全通道服務。這會導致 TCP 連線關閉。如果發生任何 TCP 通訊端錯誤，本機代理會透過通道傳送訊息，通知對方關閉其 TCP 連線。根據預設，本機代理一律使用 SSL 通訊。

使用通道之後，可以安全地終止本機代理程序。建議您呼叫 `CloseTunnel` 來明確地關閉通道。呼叫 `CloseTunnel` 之後，作用中的通道用戶端可能不會立即關閉。

本機代理安全最佳實務

執行本機代理時，請遵循下列安全最佳實務：

- 避免使用 `-t` 本機代理引數傳入存取字符。建議您使用 `AWSIOT_TUNNEL_ACCESS_TOKEN` 環境變數來設定本機代理的存取字符。
- 在作業系統或環境中以最低權限執行本機代理可執行檔。
  - 避免在 Windows 上以系統管理員身分執行本機代理。
  - 避免在 Linux 和 macOS 上以 root 身分執行本機代理。
- 考慮在個別的主機、容器、沙箱，chroot jail 或虛擬化環境上執行本機代理。
- 使用相關的安全標誌構建本地代理，具體取決於您的工具鏈。
- 在具有多個網路界面的裝置上，使用 `-b` 引數將 TCP 通訊端繫結至用來與目標應用程式通訊的網路界面。

IoT Agent Snippet

IoT 代理程式用來接收包含用戶端存取字符的 MQTT 訊息，並在遠端裝置上啟動本機代理。如果您想要安全通道服務交付用戶端存取字符，則必須在遠端裝置上安裝並執行 IoT 代理程式。IoT 代理程式必須訂閱下列保留的 IoT MQTT 主題：

```
$aws/things/<thing-name>/tunnels/notify
```

其中 `thing-name` 是與遠端裝置相關聯的 IoT 物件名稱。

以下是 MQTT 訊息承載的範例：

```
{
    "clientAccessToken": "<destination-client-access-token>",
    "clientMode": "destination",
    "region": "<aws-region",
    "services": ["destination-service"]
}
```

在收到 MQTT 訊息之後，IoT 代理程式必須使用適當的參數，在遠端裝置上啟動本機代理。

下列 Java 程式碼示範如何使用 Java 程式庫中的 AWS IoT 裝置 SDK 和 ProcessBuilder，來建置簡單的 IoT 代理程式，以使用安全通道服務。

```
// Find the IoT device endpoint for your AWS account
final String endpoint = iotClient.describeEndpoint(new
 DescribeEndpointRequest().withEndpointType("iot:Data-ATS")).getEndpointAddress();

// Instantiate the IoT Agent with your AWS credentials
final String thingName = "RemoteDeviceA";
final String tunnelNotificationTopic = String.format("$aws/things/%s/tunnels/notify",
 thingName);
final AWSIotMqttClient mqttClient = new AWSIotMqttClient(endpoint, thingName,
```

```
                 "your_aws_access_key", "your_aws_secret_key");

try {
    mqttClient.connect();
    final TunnelNotificationListener listener = new
 TunnelNotificationListener(tunnelNotificationTopic);
    mqttClient.subscribe(listener, true);
}
finally {
    mqttClient.disconnect();
}

private static class TunnelNotificationListener extends AWSIotTopic {
    public TunnelNotificationListener(String topic) {
        super(topic);
    }

    @Override
    public void onMessage(AWSIotMessage message) {
        try {
            // Deserialize the MQTT message
            final JSONObject json = new JSONObject(message.getStringPayload());

            final String accessToken = json.getString("clientAccessToken");
            final String region = json.getString("region");

            final String clientMode = json.getString("clientMode");
            if (!clientMode.equals("destination")) {
                throw new RuntimeException("Client mode " + clientMode + " in the MQTT
 message is not expected");
            }

            final JSONArray servicesArray = json.getJSONArray("services");
            if (servicesArray.length() > 1) {
                throw new RuntimeException("Services in the MQTT message has more than 1
 service");
            }
            final String service = servicesArray.get(0).toString();
            if (!service.equals("SSH")) {
                throw new RuntimeException("Service " + service + " is not supported");
            }

            // Start the destination local proxy in a separate process to connect to the
 SSH Daemon listening port 22
            final ProcessBuilder pb = new ProcessBuilder("localproxy",
                        "-t", accessToken,
                        "-r", region,
                        "-d", "localhost:22");
            pb.start();
        }
        catch (Exception e) {
            log.error("Failed to start the local proxy", e);
        }
    }
}
```

設定遠端裝置

如果您想要透過訂閱保留的 IoT MQTT 主題以外的方法將目標用戶端存取字符交付給遠端裝置，則在遠端裝置上您可能需要兩個元件：

- 目標用戶端存取字符接聽程式。

- 本機代理。

目標用戶端存取字符接聽程式應該使用您選擇的用戶端存取字符交付機制。它必須能夠在目標模式下啟動本機代理。

裝置佈建

當您使用 AWS IoT 佈建裝置時，需要建立一組資源，以便您的裝置和 AWS IoT 能夠安全地通訊。可以建立其他資源來協助您管理裝置機群。可以在佈建過程中建立下列資源：

- IoT 物件
- X.509 憑證
- IoT 政策

> **Note**
>
> 不需要建立 IoT，但它可讓您依物件類型，物件群組和物件屬性搜索裝置更有效地管理您的裝置機群。如需更多詳細資訊，請參閱 機群索引服務 (p. 487)。

IoT 物件是 AWS IoT 裝置登錄中的項目，每一個物件都包含唯一的名稱、一組屬性，並與實體裝置相關聯。物件可以使用物件類型來定義，也可以分組為物件群組。如需詳細資訊，請參閱 利用 AWS IoT 管理裝置 (p. 106)。

裝置會使用 X.509 憑證搭配 AWS IoT 來執行交互身分驗證。您可以註冊現有憑證，或使 AWS IoT 為您產生並註冊新憑證。您可以將憑證附加至代表裝置的物件，將該憑證與裝置產生關聯。您也必須將憑證和關聯的私有金鑰複製到裝置。裝置在連線至 AWS IoT 時會顯示憑證。如需詳細資訊，請參閱 身分驗證 (p. 130)。

IoT 政策定義了裝置可在 AWS IoT 中執行哪些操作。IoT 政符會附加至裝置憑證。當裝置向 AWS IoT 顯示憑證時，其會被授與政策中指定的許可。如需詳細資訊，請參閱 ??? (p. 145)。每個裝置都需要憑證才能與 AWS IoT 進行通訊。

AWS IoT 支援使用佈建範本來自動佈建機群組。佈建範本描述了 AWS IoT 佈建裝置所需的資源，如先前所述。範本包含的變數可讓您使用一個範本來佈建多個裝置。佈建裝置時，您可以使用字典 (map) 為裝置特定的變數指定值。若要佈建另一個裝置，請在字典中指定新值。

無論您的裝置是否具有唯一憑證 (及其關聯的私有金鑰)，您都可以使用自動佈建。

使用機群佈建來佈建沒有裝置憑證的裝置

> The fleet provisioning feature is in beta and is subject to change. We recommend that you use this feature with test devices only. It is currently available in the us-east-1 region.

當您使用 AWS IoT 機群佈建時，AWS IoT 可以在您的裝置第一次連線至 AWS IoT 時，產生裝置憑證和私有金鑰，並將其提供給您的裝置。裝置憑證已註冊，以供日常使用。使用機群佈建的方式有兩種：

- 透過要求。
- 透過信任的使用者。

透過要求佈建

裝置可以使用內嵌的佈建要求憑證和私有金鑰 (這是特殊用途的登入資料憑證) 來製造。如果這些憑證已向 AWS IoT 註冊，AWS IoT 可以將它們交換為裝置可用於一般操作的唯一裝置憑證。這包括下列步驟：

1. 您可以使用 `CreateProvisioningTemplate` API 來建立佈建範本。這個 API 傳回範本 ARN。如需詳細資訊，請參閱 機群佈建 API (p. 472)。您也可以從 IoT 主控台建立機群佈建範本。若要執行此作業：

   1. 從導覽窗格中，選擇 Onboard (入門)，然後選擇 Fleet provisioning templates (機群佈建範本)。
   2. 選擇 Create (建立)，並依照提示進行。
2. 您可以建立要用作佈建要求的憑證和關聯私有金鑰。
3. 您可以向 AWS IoT 註冊這些憑證，並將其與限制憑證使用的 IoT 政策建立關聯。下列範例 IoT 政策會限制使用與此政策相關聯的憑證來佈建裝置。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": ["iot:Connect"],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": ["iot:Publish","iot:Receive"],
            "Resource": [
                "arn:aws:iot:us-east-1:225901001567:topic/$aws/certificates/create/*",
                "arn:aws:iot:us-east-1:225901001567:topic/$aws/provisioning-templates/
{{name}}/provision/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iot:Subscribe",
            "Resource": [
                "arn:aws:iot:us-east-1:225901001567:topicfilter/$aws/certificates/
create/*",
                "arn:aws:iot:us-east-1:225901001567:topicfilter/$aws/provisioning-
templates/{{name}}/provision/*"
            ]
        }
    ]
}
```

4. 佈建裝置時，您可以授與 AWS IoT 服務許可，以建立或更新 IoT 資源，例如帳戶中的物件和憑證。您可以透過將 `AWSIoTThingsRegistration` 受管政策附加到信任 AWS IoT 服務委託人的 IAM 角色 (稱為佈建角色) 來執行此操作。
5. 裝置製造時，安全地嵌入佈建要求。
6. 裝置會使用佈建要求，搭配 AWS IoT 使用 AWS IoT 裝置 SDK 進行身分驗證。
7. 裝置會收到裝置憑證和私有金鑰，其會存放在安全位置並用於未來搭配 AWS IoT 進行身分驗證。同時，機群佈建服務會建立雲端資源，例如 IoT 物件、物件群組和屬性，如佈建範本中所定義。

   Important

   佈建要求私有金鑰應該始終加以保護，包括在裝置上。建議您使用 AWS IoT CloudWatch 指標和日誌來監控是否有誤用的跡象。如果您偵測到誤用，請停用佈建要求憑證，使其無法用於裝置佈建。

由信任的使用者佈建

在許多情況下，使用者或安裝技術人員第一次使用行動應用程式，在其部署的位置設定裝置時，裝置會連線至 AWS IoT。此程序包含以下步驟：

1. 您可以使用 `CreateProvisioningTemplate` API 來建立佈建範本。這個 API 傳回範本 ARN。
2. 您可以建立受信任使用者用來啟動佈建程序的 IAM 角色。佈建範本只允許該使用者佈建裝置。例如：

```
{
    "Effect": "Allow",
    "Action": [
        "iot:CreateProvisioningClaim",
    ],
    "Resource": [
        "arn:aws:iot:us-west-2:123456789012:provisioningtemplate/<templateName>"
    ]
}
```

3. 佈建裝置時，您可以授與 AWS IoT 服務許可，以建立或更新 IoT 資源，例如帳戶中的物件和憑證。此作法為將 `AWSIoTThingsRegistration` 受管政策附加到信任 AWS IoT 服務委託人的 IAM 角色 (稱為佈建角色)。
4. 受信任使用者會登入您的佈建行動應用程式或 Web 服務。
5. 行動應用程式或 Web 應用程式會使用 IAM 角色和 `CreateProvisioningClaim` API，從 AWS IoT 取得臨時佈建要求。
6. 行動應用程式或 Web 應用程式會提供佈建要求給裝置，也會提供組態資訊 (例如 Wi-Fi 登入資料)。
7. 裝置會使用臨時佈建要求，以使用 AWS IoT 軟體開發套件 (p. 678) 來連線至 AWS IoT。
8. 裝置會收到唯一的裝置憑證和私有金鑰，其會存放在安全位置並用於未來連線至 AWS IoT。同時，機群佈建服務會建立雲端資源，例如 IoT 物件、物件群組和屬性，如佈建範本中所定義。

機群佈建 API

機群佈建中使用的 API 有三種類別：

- 控制平面 API，用來建立和管理機群佈建範本，以及設定受信任的使用者政策。
  - CreateProvisioningTemplate
  - CreateProvisioningTemplateVersion
  - DeleteProvisioningTemplate
  - DeleteProvisioningTemplateVersion
  - DescribeProvisioningTemplate
  - DescribeProvisioningTemplateVersion
  - ListProvisioningTemplates
  - ListProvisioningTemplateVersions
- 受信任使用者會使用控制平面 API 來產生在 Wi-Fi 設定或執行類似方法期間傳遞至裝置的臨時入門要求。此類別中有一個 API：CreateProvisioningClaim。
- 裝置在佈建過程中使用的 API (使用內嵌在裝置中的佈建要求，或是由受信任使用者傳遞給它的佈建要求)。如需詳細資訊，請參閱裝置佈建 MQTT API

裝置佈建 MQTT API

機群佈建服務支援兩個 MQTT API：`CreateKeysAndCertificate` 和 `RegisterThing`。

CreateKeysAndCertificate

> **Note**
>
> 本文件中的範例使用 JSON 來提供可讀性，但您也可以使用 CBOR 格式。

若要建立憑證，請在 `$aws/certificates/create/cbor` 或 `$aws/certificates/create/json` 上發佈訊息。

請求承載：

```
{}
```

回應承載：

若要收到回應，請訂閱 `$aws/certificates/create/cbor/accepted` 或 `$aws/certificates/create/json/accepted`

```
{
            "certificateId": "string",
            "certificatePem": "string",
            "privateKey": "string",
            "certificateOwnershipToken": "string"
        }
```

`certificateId`

憑證 ID。

`certificatePem`

憑證資料 (PEM 格式)。

`privateKey`

私有金鑰。

certificateOwnershipToken

在佈建期間證明憑證擁有權的字符。

若要接收錯誤回應，請訂閱 `$aws/certificates/create/cbor/rejected` 或 `$aws/certificates/create/json/rejected`

錯誤承載：

```
{
    "statusCode": int,
    "errorCode": "string",
    "errorMessage": "string"
}
```

`statusCode`

狀態碼。

`errorCode`

錯誤代碼。

`errorMessage`

錯誤訊息。

RegisterThing

若要佈建物件，請在 `$aws/provisioning-templates/{{templateName}}/provision/cbor` 或 `$aws/provisioning-templates/{{templateName}}/provision/json` 上發佈訊息。

請求承載：

```
{
```

```
    "certificateOwnershipToken": "string",
    "parameters": {
        "string": "string"
    },
}
```

templateName

佈建範本名稱。

certificateOwnershipToken

證明憑證擁有權的字符。字符是由 AWS IoT 在您透過 MQTT 建立憑證時產生的。

參數

選用。佈建期間要傳送至裝置的名稱值對。

若要收到回應，請訂閱 `$aws/provisioning-templates/{{templateName}}/provision/cbor/accepted` 或 `$aws/provisioning-templates/{{templateName}}/provision/json/accepted`

回應承載：

```
{
    "deviceConfiguration": {
        "string": "string"
    },
    "thingName": "string"
}
```

`thingName`

佈建期間建立的 IoT 物件名稱。

`deviceConfiguration`

範本中定義的裝置組態。

若要接收錯誤回應，請訂閱 `$aws/provisioning-templates/{{templateName}}/provision/cbor/rejected` 或 `$aws/provisioning-templates/{{templateName}}/provision/json/rejected`

錯誤承載：

```
{
    "statusCode": int,
    "errorCode": "string",
    "errorMessage": "string"
}
```

`statusCode`

狀態碼。

`errorCode`

錯誤代碼。

`errorMessage`

錯誤訊息。

佈建具有裝置憑證的裝置

> The fleet provisioning feature is in beta and is subject to change. We recommend that you use this feature with test devices only. It is currently available in the us-east-1 region.

AWS IoT 提供三種方式，以在裝置上已有裝置憑證 (以及關聯的私有金鑰) 時佈建這些裝置：

- 含佈建範本的單一物件佈建。如果您一次只需要佈建一部裝置，適合使用此選項。
- 含範本的即時佈建 (JITP)，其會在裝置第一次連線到 AWS IoT 時佈建裝置。如果您需要註冊大量裝置，但您沒有裝置的相關資訊，無法組合到大量佈建清單內，適合使用此選項。
- 大量註冊此選項可讓您指定儲存在 S3 儲存貯體內的檔案中的單一物件佈建範本值清單。如果您有大量已知裝置且您可將其所需特性組合到清單內，適合使用此選項。

單一物件佈建

若要佈建物件，請使用 RegisterThing API 或 `register-thing` CLI 命令。`register-thing`CLI 命令會使用以下引數：

--template-body

該佈建範本。

--parameters

佈建範本使用的名稱/值對參數清單 (JSON 格式) (例如，`{"ThingName" : "MyProvisionedThing", "CSR" : "<csr-text>"}`)。

請參閱「佈建範本 (p. 478)」。

RegisterThing 或 `register-thing` 會針對資源及其建立的憑證文字，回傳 ARN：

```
{
    "certificatePem": "<certificate-text>",
    "resourceArns": {
    "PolicyLogicalName": "arn:aws:iot:us-
west-2:123456789012:policy/2A6577675B7CD1823E271C7AAD8184F44630FFD7",
    "certificate": "arn:aws:iot:us-west-2:123456789012:cert/
cd82bb924d4c6ccbb14986dcb4f40f30d892cc6b3ce7ad5008ed6542eea2b049",
    "thing": "arn:aws:iot:us-west-2:123456789012:thing/MyProvisionedThing"
    }
}
```

若字典省略參數，則會使用預設值。若未指定預設值，則參數不會替換成值。

即時佈建

您可以在裝置首次嘗試連接至 AWS IoT 時，即刻進行佈建。即時佈建 (JITP) 於 CA 憑證中設定。若要佈建裝置，您必須啟用自動註冊，而且您在佈建裝置時，其裝置憑證簽署所用的 CA 憑證，必須與佈建範本建立關聯。

在使用 RegisterCACertificate API 或 `register-ca-certificate` CLI 命令來註冊 CA 憑證時，您可以進行這些設定：

```
aws iot register-ca-certificate --ca-certificate <your-ca-cert> --verification-cert
```

```
                <your-verification-cert> --set-as-active --allow-auto-registration --
registration-config file://<your-template>
```

如需詳細資訊，請參閱註冊 CA 憑證相關文章。

您也可以使用 UpdateCACertificate API 或 `update-ca-certificate` CLI 命令，更新 CA 憑證的設定：

```
$ aws iot update-ca-certificate --cert-id <caCertificateId> --new-auto-registration-status
 ENABLE --registration-config file://<your-template>
```

當裝置使用已註冊 CA 憑證所簽署的憑證來嘗試連接至 AWS IoT 時，AWS IoT 會自該 CA 憑證載入範本，並使用該範本呼叫 RegisterThing。JITP 工作流程會先註冊狀態值為 PENDING_ACTIVATION 的憑證。裝置佈建完成時，憑證狀態將變為 ACTIVE。

AWS IoT 定義下列您可於佈建範本內宣告及參考的參數：

- `AWS::IoT::Certificate::Country`
- `AWS::IoT::Certificate::Organization`
- `AWS::IoT::Certificate::OrganizationalUnit`
- `AWS::IoT::Certificate::DistinguishedNameQualifier`
- `AWS::IoT::Certificate::StateName`
- `AWS::IoT::Certificate::CommonName`
- `AWS::IoT::Certificate::SerialNumber`
- `AWS::IoT::Certificate::Id`

這些佈建範本參數的值僅限於 JITP 會自佈建中的裝置憑證的主旨欄位擷取的值。`AWS::IoT::Certificate::Id` 參數代表的是內部產生的 ID，而不是憑證內包含的 ID。您可以使用 AWS IoT 規則內的 `principal()` 函數取得此 ID 的值。

以下 JSON 檔案為完整 JITP 範本的範例。`templateBody` 欄位的值必須是指定為溢出字串的 JSON 物件，且只能使用前述清單中的值。您可以使用各種工具來建立必要的 JSON 輸出，例如 `json.dumps` (Python) 或 `JSON.stringify` (節點)。`roleARN` 欄位的值必須為連接 `AWSIoTThingsRegistration` 之角色的 ARN。此外，您的範本可以使用現有的 `PolicyName`，而不是範例中的內嵌 `PolicyDocument`。(第一個範例加入換行以增加可讀性，但您可以直接複製並貼上底下出現的範本。)

```
        {
            "templateBody" : "{
                \r\n    \"Parameters\" : {\r\n
                        \"AWS::IoT::Certificate::CommonName\": {\r\n    \"Type\":
 \"String\"\r\n        },\r\n
                        \"AWS::IoT::Certificate::SerialNumber\": {\r\n  \"Type\":
 \"String\"\r\n        },\r\n
                        \"AWS::IoT::Certificate::Country\": {\r\n   \"Type\": \"String
\"\r\n        },\r\n
                        \"AWS::IoT::Certificate::Id\": {\r\n    \"Type\": \"String\"\r
\n        }\r\n    },\r\n
                    \"Resources\": {\r\n
                        \"thing\": {\r\n
                            \"Type\": \"AWS::IoT::Thing\",\r\n
                            \"Properties\": {\r\n
                                \"ThingName\": {\r\n    \"Ref\":
 \"AWS::IoT::Certificate::CommonName\"\r\n  },\r\n
                                \"AttributePayload\": {\r\n
                                    \"version\": \"v1\",\r\n
                                    \"serialNumber\": {\r\n
                                        \"Ref\": \"AWS::IoT::Certificate::SerialNumber
\"\r\n    }\r\n   },\r\n
```

```
                                    \"ThingTypeName\": \"lightBulb-versionA\",\r\n

                                    \"ThingGroups\": [\r\n
                                        \"v1-lightbulbs\",\r\n  {\r\n

                                            \"Ref\": \"AWS::IoT::Certificate::Country\"\r
\n }\r\n   ]\r\n   },\r\n
                            \"OverrideSettings\": {\r\n
                                \"AttributePayload\": \"MERGE\",\r\n
                                \"ThingTypeName\": \"REPLACE\",\r\n
                                \"ThingGroups\": \"DO_NOTHING\"\r\n }\r\n   },\r\n
                        \"certificate\": {\r\n
                            \"Type\": \"AWS::IoT::Certificate\",\r\n
                            \"Properties\": {\r\n
                                \"CertificateId\": {\r\n    \"Ref\":
 \"AWS::IoT::Certificate::Id\"\r\n  },\r\n
                                \"Status\": \"ACTIVE\"\r\n  },\r\n
                            \"OverrideSettings\": {\r\n
                                \"Status\": \"DO_NOTHING\"\r\n  }\r\n   },\r\n
                        \"policy\": {\r\n
                            \"Type\": \"AWS::IoT::Policy\",\r\n
                            \"Properties\": {\r\n
                                \"PolicyDocument\": \"{
                                    \\\"Version\\\": \\\"2012-10-17\\\",
                                    \\\"Statement\\\": [{
                                    \\\"Effect\\\": \\\"Allow\\\",
                                    \\\"Action\\\":[\\\"iot:Publish\\\"],
                                    \\\"Resource\\\": [\\\"arn:aws:iot:us-
east-1:123456789012:topic\/sample\/topic\\\"] }] }\"\r\n   }\r\n   }\r\n
                                }\r\n}",
            "roleArn" : "arn:aws:iam::123456789012:role/Provisioning-JITP"
        }
```

此處為您可複製並貼上的版本：

```
    {
      "templateBody" : "{\r\n    \"Parameters\" : {\r\n
 \"AWS::IoT::Certificate::CommonName\": {\r\n            \"Type\": \"String\"\r\n        },
\r\n        \"AWS::IoT::Certificate::SerialNumber\": {\r\n            \"Type\": \"String
\"\r\n        },\r\n        \"AWS::IoT::Certificate::Country\": {\r\n            \"Type\":
 \"String\"\r\n        },\r\n        \"AWS::IoT::Certificate::Id\": {\r\n            \"Type
\": \"String\"\r\n        }\r\n    },\r\n    \"Resources\": {\r\n        \"thing\": {\r\n
            \"Type\": \"AWS::IoT::Thing\",\r\n            \"Properties\": {\r\n
      \"ThingName\": {\r\n                    \"Ref\": \"AWS::IoT::Certificate::CommonName
\"\r\n                },\r\n                \"AttributePayload\": {\r\n
   \"version\": \"v1\",\r\n                    \"serialNumber\": {\r\n
      \"Ref\": \"AWS::IoT::Certificate::SerialNumber\"\r\n                    }\r\n
           },\r\n                \"ThingTypeName\": \"lightBulb-versionA\",\r\n
      \"ThingGroups\": [\r\n                    \"v1-lightbulbs\",\r\n
 {\r\n                        \"Ref\": \"AWS::IoT::Certificate::Country\"\r\n
        }\r\n                ]\r\n            },\r\n            \"OverrideSettings\": {\r
\n                \"AttributePayload\": \"MERGE\",\r\n                \"ThingTypeName\":
 \"REPLACE\",\r\n                \"ThingGroups\": \"DO_NOTHING\"\r\n            }\r\n
   },\r\n        \"certificate\": {\r\n            \"Type\": \"AWS::IoT::Certificate\",\r\n
            \"Properties\": {\r\n                \"CertificateId\": {\r\n
   \"Ref\": \"AWS::IoT::Certificate::Id\"\r\n                },\r\n                \"Status
\": \"ACTIVE\"\r\n            },\r\n            \"OverrideSettings\": {\r\n
  \"Status\": \"DO_NOTHING\"\r\n            }\r\n        },\r\n        \"policy\": {\r\n
            \"Type\": \"AWS::IoT::Policy\",\r\n            \"Properties\": {\r\n
      \"PolicyDocument\": \"{ \\\"Version\\\": \\\"2012-10-17\\\", \\\"Statement\\\": [{ \
\\\"Effect\\\": \\\"Allow\\\", \\\"Action\\\":[\\\"iot:Publish\\\"], \\\"Resource\\\": [\\
\"arn:aws:iot:us-east-1:123456789012:topic\/foo\/bar\\\"] }] }\"\r\n            }\r\n
  }\r\n    }\r\n}",
```

```
    "roleArn" : "arn:aws:iam::123456789012:role/JITPRole"
  }
```

此範例範本會宣告擷取自憑證並使用在 `Resources` 部分的 `AWS::IoT::Certificate::CommonName`、`AWS::IoT::Certificate::SerialNumber`、`AWS::IoT::Certific` 和 `AWS::IoT::Certificate::Id` 佈建參數的值。JITP 工作流程接著使用此範本執行下列動作：

- 註冊憑證並將其狀態設為 PENDING_ACTIVE。
- 建立一個物件資源。
- 建立一個政策資源。
- 將政策連接至憑證。
- 將憑證連接至物件。
- 更新憑證狀態為 ACTIVE。

您應該會看到憑證註冊在 AWS CloudTrail 中顯示為記錄事件 (`RegisterCACertificate`)。您也可以使用 CloudTrail 來排除 JITP 範本的問題。

> **Note**
>
> JITP 在佈建程序中會呼叫其他 AWS IoT 控制平面 API。這些呼叫可能會超過針對您帳戶設定的 AWS IoT 調節限制，並導致調節呼叫。如有必要，請聯絡 AWS 客戶支援以提高您的調節限制。

大量註冊

您可以使用 `start-thing-registration-task` 命令來大量註冊物件。此命令需要註冊範本、S3 儲存貯體名稱、金鑰名稱，以及可存取 S3 儲存貯體檔案的角色 ARN。S3 儲存貯體的檔案，內含用於替換範本參數的值。該檔案必須為換行分隔的 JSON 檔案。每行均包含用於註冊單一裝置的所有參數值。例如：

```
{"ThingName": "foo", "SerialNumber": "123", "CSR": "csr1"}
{"ThingName": "bar", "SerialNumber": "456", "CSR": "csr2"}
```

下列與大量註冊相關的 API 也許有用：

- ListThingRegistrationTasks：列出目前的大量實物佈建任務。
- DescribeThingRegistrationTask：提供特定大量物件註冊任務的相關資訊。
- StopThingRegistrationTask：停止大量物件註冊任務。
- ListThingRegistrationTaskReports：用於檢查大量物件註冊任務的結果及/或失敗。

> **Note**
>
> - 一次僅可執行一個大量註冊操作任務 (每個帳戶)。
> - 大量註冊操作會呼叫其他 AWS IoT 控制平面 API。這些呼叫可能會超過您帳戶中的 AWS IoT 調節限制，並導致調節錯誤。如有必要，請聯絡 AWS 客戶支援以提高您的 AWS IoT 調節限制。

佈建範本

佈建範本為 JSON 文件，本文件使用參數來描述您裝置與 AWS IoT 互動所需的必要資源。範本內含兩個部分：`Parameters`和 `Resources`。AWS IoT 中有兩種類型的佈建範本。一個用於 JITP 和大量註冊，第二個則用於機群佈建。

參數部分

Parameters 部分宣告 Resources 部分所使用的參數。每個參數都會宣告名稱、類型和可選的預設值。與範本一同傳入的字典若未包含該參數的值，將使用預設值。範本文件的 Parameters部分如下：

```
{
    "Parameters" : {
        "SerialNumber" : {
            "Type" : "String"
        },
        "Location" : {
            "Type" : "String",
            "Default" : "WA"
        },
        "CSR" : {
            "Type" : "String"
        }
    }
}
```

此範本片段宣告四項參數：ThingName、SerialNumber、Location 和 CSR。這些參數均為類型 String。Location參數宣告的預設值為 "WA"。

資源部分

此範本的 Resources 區段宣告您裝置與 AWS IoT 通訊的所需資源：一個實物、一個憑證和一個或多個 IoT 政策。每一資源都要指定一個邏輯名稱、一種類型和一組屬性。

邏輯名稱可讓您引用範本其他位置的資源。

類型則指定您欲宣告的資源種類。有效類型為：

- AWS::IoT::Thing
- AWS::IoT::Certificate
- AWS::IoT::Policy

您指定的屬性取決於您欲宣告的資源類型。

物件資源

物件資源宣告使用下列屬性：

- ThingName: 字串.
- AttributePayload：選用。名稱/值對清單。
- ThingTypeName：選用。與該物件類型相關的字串。
- ThingGroups：選用。該物件所屬的群組清單。

憑證資源

您可以使用下列其中一種方式來指定憑證：

- 憑證簽署要求 (CSR)。
- 現有裝置憑證的憑證 ID。(只有憑證 ID 可與機群佈建範本搭配使用)
- 搭配 CA 憑證向 AWS IoT 註冊所建立的裝置憑證。如果您的多個 CA 憑證註冊使用相同的主體欄位，您也必須輸入用於簽署該裝置憑證的 CA 憑證。

> **Note**
>
> 若您使用範本宣告憑證，請透過這些方法其中之一。例如，若您使用 CSR，則無法同時指定憑證 ID 或裝置憑證。如需詳細資訊，請參閱 AWS IoT 和憑證。

如需更多詳細資訊，請參閱 X.509 憑證概觀 (p. 130)。

憑證資源宣告使用下列屬性：

- `CertificateSigningRequest`: 字串.
- `CertificateID`: 字串.
- `CertificatePem`: 字串.
- `CACertificatePem`: 字串.
- `Status`：選用。可以是 `ACTIVE` 或 `INACTIVE` 的字串。預設為 ACTIVE。

範例：

- CSR 指定的憑證：

```
{
    "certificate" : {
        "Type" : "AWS::IoT::Certificate",
        "Properties" : {
            "CertificateSigningRequest": {"Ref" : "CSR"},
            "Status" : "ACTIVE"
        }
    }
}
```

- 以現有憑證 ID 指定的憑證：

```
{
    "certificate" : {
        "Type" : "AWS::IoT::Certificate",
        "Properties" : {
            "CertificateId": {"Ref" : "CertificateId"}
        }
    }
}
```

- 以現有憑證 .pem 和 CA 憑證 .pem 指定的憑證：

```
{
    "certificate" : {
        "Type" : "AWS::IoT::Certificate"
        "Properties" : {
            "CACertificatePem": {"Ref" : "CACertificatePem"},
            "CertificatePem": {"Ref" : "CertificatePem"}
        }
    }
}
```

政策資源

政策資源宣告使用下列屬性之一：

- `PolicyName`：選用。字串.預設為政策文件雜湊。如果您使用的是現有 AWS IoT 政策，則請在 `PolicyName` 屬性輸入該政策名稱。請勿包含 `PolicyDocument` 屬性。

- PolicyDocument：選用。指定為逸出字串的 JSON 物件。若未提供 PolicyDocument，則該政策必已建立。

  Note

  若有 Policy部分，必須指定 PolicyName 或 PolicyDocument 其一。

覆寫設定

若範本指定已存在的資源，OverrideSettings部分可讓您指定欲採取的動作：

DO_NOTHING

保留原本資源。

REPLACE

以範本指定的資源來取代原有資源。

FAIL

因 ResourceConflictsException造成請求失敗。

MERGE

僅適用 ThingGroups的 AttributePayload 及 thing 屬性。將物件現有屬性或群組成員資格，與範本指定的屬性合併。

當您宣告物件資源時，可為下列屬性指定 OverrideSettings：

- ATTRIBUTE_PAYLOAD
- THING_TYPE_NAME
- THING_GROUPS

當您宣告憑證資源時，可為 OverrideSettings 屬性指定 Status。

OverrideSettings 不適用於政策資源。

資源範例

下列範本片段宣告一個物件、一個憑證及一個政策：

```
{
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : { "version" : "v1", "serialNumber" :  {"Ref" :
 "SerialNumber"}},
                "ThingTypeName" :  "lightBulb-versionA",
                "ThingGroups" : ["v1-lightbulbs", {"Ref" : "Location"}]
            },
            "OverrideSettings" : {
                "AttributePayload" : "MERGE",
                "ThingTypeName" : "REPLACE",
                "ThingGroups" : "DO_NOTHING"
            }
        },
        "certificate" : {
```

```
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateSigningRequest": {"Ref" : "CSR"},
                "Status" : "ACTIVE"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : "{ \"Version\": \"2012-10-17\", \"Statement\":
 [{ \"Effect\": \"Allow\", \"Action\":[\"iot:Publish\"], \"Resource\": [\"arn:aws:iot:us-
east-1:123456789012:topic/foo/bar\"] }] }"
            }
        }
    }
}
```

物件以下列項目進行宣告：

- 邏輯名稱 `"thing"`。
- 類型 `AWS::IoT::Thing`。
- 物件的一組屬性。

  物件的屬性包含物件名稱、一組屬性、選用的物件類型名稱，以及該物件所屬的物件群組選用清單。

`{"Ref":"<parameter-name>"}`會參考這些參數。在評估範本時，參數將取代為與範本一同傳入的字典的參數值。

憑證以下列項目進行宣告：

- 邏輯名稱 `"certificate"`。
- 類型 `AWS::IoT::Certificate`。
- 一組屬性。

  這些屬性包括憑證的 CSR 及設定為 `ACTIVE`的狀態。與範本一同傳入的字典內，CSR 文字以參數傳遞。

政策以下列項目進行宣告：

- 邏輯名稱 `"policy"`。
- 類型 `AWS::IoT::Policy`。
- 現有政策名稱或政策文件。

JITP 和大量註冊的範本範例

下列 JSON 檔案為以 CSR 指定憑證的完整佈建範本的其中一個範例：

(`PolicyDocument` 欄位值必須是指定為逸出字串的 JSON 物件。)

```
{
    "Parameters" : {
        "SerialNumber" : {
            "Type" : "String"
        },
        "Location" : {
            "Type" : "String",
            "Default" : "WA"
```

```
        },
        "CSR" : {
            "Type" : "String"
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : { "version" : "v1", "serialNumber" :  {"Ref" :
 "SerialNumber"}},
                "ThingTypeName" :  "lightBulb-versionA",
                "ThingGroups" : ["v1-lightbulbs", {"Ref" : "Location"}]
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateSigningRequest": {"Ref" : "CSR"},
                "Status" : "ACTIVE"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : "{ \"Version\": \"2012-10-17\", \"Statement\":
 [{ \"Effect\": \"Allow\", \"Action\":[\"iot:Publish\"], \"Resource\": [\"arn:aws:iot:us-
east-1:123456789012:topic/foo/bar\"] }] }"
            }
        }
    }
}
```

下列 JSON 檔案為以憑證 ID 指定現有憑證的完整佈建範本的其中一個範例：

```
{
    "Parameters" : {
        "SerialNumber" : {
            "Type" : "String"
        },
        "Location" : {
            "Type" : "String",
            "Default" : "WA"
        },
        "CertificateId" : {
            "Type" : "String"
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : { "version" : "v1", "serialNumber" :  {"Ref" :
 "SerialNumber"}},
                "ThingTypeName" :  "lightBulb-versionA",
                "ThingGroups" : ["v1-lightbulbs", {"Ref" : "Location"}]
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateId": {"Ref" : "CertificateId"}
            }
        },
        "policy" : {
```

```
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : "{ \"Version\": \"2012-10-17\", \"Statement\":
 [{ \"Effect\": \"Allow\", \"Action\":[\"iot:Publish\"], \"Resource\": [\"arn:aws:iot:us-
east-1:123456789012:topic/foo/bar\"] }] }"
            }
        }
    }
}
```

機群佈建

機群佈建範本是由 AWS IoT 用來設定雲端和裝置組態。這些範本會使用與 JITP 和大量註冊範本相同的參數和資源。如需詳細資訊，請參閱 佈建範本 (p. 478)。機群佈建範本可以包含一個 `Mapping` 區段和一個 `DeviceConfiguration` 區段。您可以使用機群佈建範本內的內建函數來產生裝置特定的組態。機群佈建範本是具名資源，由 ARN 識別 (例如，`arn:aws:iot:us-west-2:1234568788:provisioningtemplate/<template-name>`)。

映射項目

選用的 `Mappings` 區段會比對索引鍵與對應的一組命名值。例如，如果您想要根據 AWS 區域來設定值，則可以建立映射，而此映射使用 AWS 區域名稱做為索引鍵，並包含您要為每個特定區域指定的值。您可以使用 `Fn::FindInMap` 內部函數來擷取映射中的值。

您不能在 `Mappings` 區段中包含參數、虛擬參數或呼叫內部函數。

裝置組態

裝置組態區段包含您要在佈建時傳送至裝置的任意資料。例如：

```
{
    "DeviceConfiguration": {
        "DeviceConfiguration": {
            "Foo":"Bar"
        }
    }
}
```

內部函數

內部函數用於佈建範本的任何區段，但 `Mappings` 區段除外。

`Fn:Join`

將一組值附加至單一值，並以指定的分隔符號隔開。如果分隔符號是空白字串，系統即會串連這一組值，而不使用分隔符號。

`Fn:Select`

依索引從物件清單傳回單一物件。

> **Important**
>
> `Fn::Select` 不會檢查 `null` 值或索引是否超出陣列邊界。這兩種情況都會導致佈建錯誤，因此您應該確保選擇了有效的索引值，並且清單中包含非空值。

`Fn:FindInMap`

傳回對應 `Mappings` 區段所宣告之兩個層級映射之索引鍵的值。

`Fn:Split`

將字串分割成字串值清單，以便您可以從字串清單中選取元素。您可以指定分隔符號來決定分割字串的位置 (例如，逗號)。分割字串後，請使用 `Fn::Select` 來選取元素。

例如，如果以逗號分隔的子網路 ID 字串匯入至您的堆疊範本，您可以在每個逗號處分割字串。從子網路 ID 清單中，使用 `Fn::Select` 來指定資源的子網路 ID。

`Fn:Sub`

用您指定的值替代輸入字串中的變數。您可以使用此函數來建構命令或輸出，其中包含建立或更新堆疊後才可供使用的數值。

機群佈建範本範例

```
{
    "Parameters" : {
        "DeviceLocation": {
            "Type": "String"
        }
    },
    "Mappings": {
        "LocationTable": {
            "Seattle": {
                "LocationUrl": "https://example.aws"
            }
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : {
                    "version" : "v1",
                    "serialNumber" : "serialNumber"
                },
                "ThingTypeName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingGroups" : ["v1-lightbulbs", "WA"],
                "BillingGroup": "BillingGroup"
            },
            "OverrideSettings" : {
                "AttributePayload" : "MERGE",
                "ThingTypeName" : "REPLACE",
                "ThingGroups" : "DO_NOTHING"
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
                "Status" : "Active"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : {
                    "Version": "2012-10-17",
                    "Statement": [{
                        "Effect": "Allow",
                        "Action":["iot:Publish"],
                        "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/foo/bar"]
```

```
                    }]
                }
            }
        }
    },
    "DeviceConfiguration": {
        "FallbackUrl": "https://www.example.com/test-site",
        "LocationUrl": {
            "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"}, "LocationUrl"]}
        }
    }
}
```

機群索引服務

機群索引是一項受管服務，您可以用來在雲端中編製索引、搜尋和彙總您的登錄資料、影子資料，以及裝置連線能力資料 (裝置生命週期事件)。機群索引設定完成後，本服務會管理您的所有實物群組、實物登錄和裝置影子的更新索引。如需彙總查詢的詳細資訊，請參閱彙總資料的查詢 (p. 498)。您可以使用簡單的查詢語言來搜尋此資料。您也可以使用搜尋查詢建立動態物件群組。

> **Note**
>
> 機群索引服務在建立或刪除實物之後，最多可能需要 30 秒的時間，才能更新機群索引。

當您啟用索引時，AWS IoT 會為您的實物或實物群組建立索引。在啟用後，您可以對您的索引執行查詢，例如尋找所有手持式且電池電量大於 70% 的裝置。AWS IoT 可讓它不斷以您的最新資料進行更新。

`AWS_Things` 是為所有物件建立的索引。`AWS_ThingGroups` 是包含所有物件群組的索引。

您可以使用 AWS IoT 主控台，管理建立索引的組態並執行搜尋查詢。選擇您想要在主控台設定頁面中使用的索引。若您偏好以程式設計方式存取，則可以使用 AWS 開發套件或 AWS CLI。

如需有關此項和其他服務的定價資訊，請參閱 AWS IoT 裝置管理定價頁面。

管理實物索引

`AWS_Things`為您所有物件的索引。您可以控制要建立索引的項目：登錄資料、影子資料和裝置連線狀態資料 (由裝置生命週期事件推動)。

啟用實物索引

您可以使用 update-indexing-configuration CLI 命令或 UpdateIndexingConfiguration API 來建立 `AWS_Things` 索引，並控制其組態。`--thing-indexing-configuration` (`thingIndexingConfiguration`) 參數可讓您控制要編製索引的資料類型 (例如，登錄、影子和裝置連線能力資料)。

`--thing-indexing-configuration` 參數採用具有以下結構的字符：

```
{
  "thingIndexingMode": "OFF"|"REGISTRY"|"REGISTRY_AND_SHADOW",
  "thingConnectivityIndexingMode": "OFF"|"STATUS",
  "customFields": [
    { name: <field-name>, type: String | Number | Boolean },
    ...
  ]
}
```

`thingIndexingMode` 屬性控制要編製索引的資料類型。有效值為：

OFF

不編製索引。

登錄檔

編製登錄資料的索引。

REGISTRY_AND_SHADOW

編製登錄資料和物件影子資料的索引。

`thingConnectivityIndexingMode` 屬性指定物件連線能力資料是否編製索引。有效值為：

OFF

物件連線能力資料不編製索引。

狀態

物件連線能力資料編製索引。

`customFields` 屬性是欄位和資料類型對的清單。彙總查詢可以根據資料類型透過這些欄位執行。您選擇的索引模式 (REGISTRY 或 REGISTRY_AND_SHADOW) 會影響可以在 `customFields` 中指定的欄位。例如，如果您指定 `REGISTRY` 索引模式，則無法從物件影子中指定欄位。您必須在 `customFields` 中指定自訂欄位，才能編製索引。

如果組態中的自訂欄位與正要編製索引的值之間出現類型不一致，機群索引服務會忽略彙總查詢的不一致值。CloudWatch 日誌在疑難排解彙總查詢問題時很有幫助。如需詳細資訊，請參閱疑難排解彙總查詢 (p. 501)。

受管欄位包含與 IoT 物件、物件群組和裝置影子相關聯的資料。受管欄位的資料類型是 AWS IoT 由定義。當你建立 IoT 物件時，請指定每個受管欄位的值。例如，物件名稱，物件群組和物件描述都是受管欄位。機群索引服務會根據您指定的索引模式編製受管欄位的索引：

- 登錄的受管欄位

```
"managedFields" : [
  {name:thingId, type:String},
  {name:thingName, type:String},
  {name:registry.version, type:Number},
  {name:registry.thingType, type:String},
  {name:registry.thingGroupNames, type:String},
]
```

- 物件影子的受管欄位

```
"managedFields" : [
  {name:shadow.version, type:Number},
  {name:shadow.delta, type:Boolean}
]
```

- 物件連線能力的受管欄位

```
"managedFields" : [
  {name:connectivity.timestamp, type:Number},
  {name:connectivity.version, type:Number},
  {name:connectivity.connected, type:Boolean}
]
```

- 物件群組的受管欄位

```
"managedFields" : [
  {name:description, type:String},
  {name:parentGroupNames, type:String},
```

```
  {name:thingGroupId, type:String},
  {name:thingGroupName, type:String},
  {name:version, type:Number},
]
```

無法在 `customFields` 中變更或顯示受管欄位。

以下是如何使用 update-indexing-configuration 來設定索引的範例：

aws iot update-indexing-configuration --thing-indexing-configuration 'thingIndexingMode=REGISTRY_AND_SHADOW,customFields=[{name=attributes.version,type=Number}, {name=attributes.color, type=String},{name=shadow.desired.power, type=Boolean}}]

這個命令會啟用針對登錄和影子資料編製索引。彙總查詢會根據資料類型使用受管欄位和所提供的 `customFields`。

您可以使用 get-indexing-configuration CLI 命令或 GetIndexingConfiguration API 來擷取目前的索引組態。

以下命令顯示如何使用 get-indexing-configuration CLI 命令來擷取目前物件索引組態，其中定義五個自訂欄位 (三個登錄自訂欄位和兩個影子自訂欄位)。

aws iot get-indexing-configuration

```
{
    "thingGroupIndexingConfiguration": {
        "thingGroupIndexingMode": "OFF"
    },
    "thingIndexingConfiguration": {
        "thingConnectivityIndexingMode": "STATUS",
        "customFields": [
            {
                "name": "attributes.customField_NUM",
                "type": "Number"
            },
            {
                "name": "shadow.desired.customField_STR",
                "type": "String"
            },
            {
                "name": "shadow.desired.customField_NUM",
                "type": "Number"
            },
            {
                "name": "attributes.customField_STR",
                "type": "String"
            },
            {
                "name": "attributes.customField_BOOL",
                "type": "Boolean"
            }
        ],
        "thingIndexingMode": "REGISTRY_AND_SHADOW",
        "managedFields": [
            {
                "name": "shadow.hasDelta",
                "type": "Boolean"
            },
            {
                "name": "registry.thingGroupNames",
                "type": "String"
            },
            {
                "name": "connectivity.version",
```

```
                "type": "Number"
            },
            {
                "name": "registry.thingTypeName",
                "type": "String"
            },
            {
                "name": "connectivity.connected",
                "type": "Boolean"
            },
            {
                "name": "registry.version",
                "type": "Number"
            },
            {
                "name": "thingId",
                "type": "String"
            },
            {
                "name": "connectivity.timestamp",
                "type": "Number"
            },
            {
                "name": "thingName",
                "type": "String"
            },
            {
                "name": "shadow.version",
                "type": "Number"
            }
        ]
    }
}
```

下表提供允許的 `thingIndexingMode` 和 `thingConnectivityIndexingMode` 組合，以及其關聯的效果。必要參數 `thingIndexingMode` 會指定 `AWS_Things` 索引是否只包含登錄資料或登錄和影子資料。選用參數 `thingConnectivityIndexingMode` 會指定索引是否也包含連線狀態資料 (當裝置在上次連線至 AWS IoT 後中斷連線時)。

thingIndexingMode	**thingConnectivityIndexingM**	結果
OFF	未指定。	沒有索引或刪除索引。
OFF	OFF	相當於先前的項目。
REGISTRY	未指定。	建立或設定 `AWS_Things`索引，僅建立登錄檔資料的索引。
REGISTRY	OFF	相當於先前的項目。(只有登錄資料會建立索引。)
REGISTRY_AND_SHADOW	未指定。	建立或設定 `AWS_Things`索引，同時建立登錄資料與影子資料的索引。
REGISTRY_AND_SHADOW	OFF	相當於先前的項目。(登錄資料和影子資料都會建立索引。)
REGISTRY	狀態	建立或設定 `AWS_Things` 索引以建立登錄資料和實物連線狀態資料的索引 (REGISTRY_AND_CONNECTIVITY_STATUS)。

thingIndexingMode	thingConnectivityIndexingM	結果
REGISTRY_AND_SHADOW	狀態	建立或設定 AWS_Things 索引以建立登錄資料、影子資料和實物連線狀態資料的索引 (REGISTRY_AND_SHADOW_AND_CONNECTI

您可以使用 AWS IoT update-indexing-configuration CLI 命令，更新您的索引組態。下列範例顯示如何使用 update-indexing-configuration CLI 命令。

簡短語法：

```
aws iot update-indexing-configuration --thing-indexing-configuration
"thingIndexingMode=REGISTRY_AND_SHADOW,thingConnectivityIndexingMode=STATUS,customFields=[{name=attrib
{name=attributes.color,type=String},{name=shadow.desired.power,type=Boolean}]"
```

JSON 語法：

```
aws iot update-indexing-configuration --cli-input-json \ '{ "thingIndexingConfiguration":
{ "thingIndexingMode": "REGISTRY_AND_SHADOW", "thingConnectivityIndexingMode": "STATUS",
"customFields": [ { "name": "shadow.desired.power", "type": "Boolean" }, { "name": "attributes.color", "type":
"String" }, { "name": "attributes.version", "type": "Number" } ] } }'
```

這些命令的輸出如下：

```
{
    "thingIndexingConfiguration": {
        "thingConnectivityIndexingMode": "STATUS",
        "customFields": [
            {
                "type": "String",
                "name": "attributes.color"
            },
            {
                "type": "Number",
                "name": "attributes.version"
            },
            {
                "type": "Boolean",
                "name": "shadow.desired.power"
            }
        ],
        "thingIndexingMode": "REGISTRY_AND_SHADOW",
        "managedFields": [
            {
                "type": "Boolean",
                "name": "connectivity.connected"
            },
            {
                "type": "String",
                "name": "registry.thingTypeName"
            },
            {
                "type": "String",
                "name": "thingName"
            },
            {
                "type": "Number",
                "name": "shadow.version"
            },
            {
```

```
                "type": "String",
                "name": "thingId"
            },
            {
                "type": "Boolean",
                "name": "shadow.hasDelta"
            },
            {
                "type": "Number",
                "name": "connectivity.timestamp"
            },
            {
                "type": "String",
                "name": "registry.thingGroupNames"
            },
            {
                "type": "Number",
                "name": "connectivity.version"
            },
            {
                "type": "Number",
                "name": "registry.version"
            }
        ]
    },
    "thingGroupIndexingConfiguration": {
        "thingGroupIndexingMode": "OFF"
    }
}
```

在下列範例中，新的自訂欄位會新增至組態：

aws iot update-indexing-configuration --thing-indexing-configuration 'thingIndexingMode=REGISTRY_AND_SHADOW,customFields=[{name=attributes.version,type=Number},{name=attributes.color,type=String},{name=shadow.desired.power,type=Boolean},{name=shadow.desired.intensity,type=Number}]'

此命令已將 `shadow.desired.intensity` 新增至索引組態。

> **Note**
>
> 更新索引組態中的自訂欄位會覆寫所有現有的自訂欄位。請務必在呼叫 update-indexing-configuration 時指定所有自訂欄位。

在重建索引之後，您可以在剛新增的欄位上使用彙總查詢，然後搜尋登錄資料、影子資料和物件連線狀態資料。

變更索引模式時，請使用新的索引模式確定所有自訂欄位都有效。例如，如果您以自訂欄位稱為 `shadow.desired.temperature` 的 `REGISTRY_AND_SHADOW` 模式開始，則必須先刪除 `shadow.desired.temperature` 自訂欄位，然後再將索引模式變更為 `REGISTRY`。如果您的索引組態包含未被索引模式編製索引的自訂欄位，更新就會失敗。

描述實物索引

以下命令說明如何使用 describe-index CLI 命令，擷取實物索引的目前狀態。

```
aws iot describe-index --index-name "AWS_Things"
{
    "indexName": "AWS_Things",
    "indexStatus": "BUILDING",
    "schema": "REGISTRY_AND_SHADOW_AND_CONNECTIVITY_STATUS"
```

```
}
```

您首次啟用索引後，AWS IoT 即會為您建立索引。若 `indexStatus` 為 `BUILDING` 狀態，則您無法在該索引內進行查詢。實物索引的 `schema` 會指出系統會為哪些類型的資料 (`REGISTRY_AND_SHADOW_AND_CONNECTIVITY_STATUS`) 建立索引。

若您變更索引的組態，將導致索引重新建立。在此程序期間的 `indexStatus` 為 `REBUILDING`。您可以在重建實物索引時，對實物索引內的資料執行查詢。例如，若您將索引組態從 `REGISTRY` 變更為 `REGISTRY_AND_SHADOW`，重建索引時，您可以查詢登錄資料 (包括最新的更新)。不過，您無法查詢影子資料，除非重建完成。建立或重建索引所需時間，取決於資料量。

查詢實物索引

使用 search-index CLI 命令來查詢索引中的資料。

aws iot search-index --index-name "AWS_Things" --query-string "thingName:mything*"

```
{
    "things":[{
         "thingName":"mything1",
         "thingGroupNames":[
            "mygroup1"
         ],
         "thingId":"a4b9f759-b0f2-4857-8a4b-967745ed9f4e",
         "attributes":{
            "attribute1":"abc"
         },
         "connectivity": {
            "connected":false,
            "timestamp":1556649874716
         }
    },
    {
        "thingName":"mything2",
        "thingTypeName":"MyThingType",
        "thingGroupNames":[
            "mygroup1",
            "mygroup2"
        ],
        "thingId":"01014ef9-e97e-44c6-985a-d0b06924f2af",
        "attributes":{
            "model":"1.2",
            "country":"usa"
        },
        "shadow":{
            "desired":{
                "location":"new york",
                "myvalues":[3, 4, 5]
            },
            "reported":{
                "location":"new york",
                "myvalues":[1, 2, 3],
                "stats":{
                    "battery":78
                }
            },
            "metadata":{
                 "desired":{
                       "location":{
                            "timestamp":123456789
                        },
                       "myvalues":{
```

```
                            "timestamp":123456789
                        }
                    },
                    "reported":{
                          "location":{
                              "timestamp":34535454
                           },
                          "myvalues":{
                              "timestamp":34535454
                          },
                          "stats":{
                              "battery":{
                                    "timestamp":34535454
                              }
                          }
                    }
                },
                "version":10,
                "timestamp":34535454
            },
            "connectivity": {
                "connected":true,
                "timestamp":1556649855046
            }
        }],
        "nextToken":"AQFCuvk7zZ3D9pOYMbFCeHbdZ+h=G"
}
```

在 JSON 回應中，`"connectivity"` (透過 `thingConnectivityIndexingMode=STATUS` 設定啟用) 會提供布林值和時間戳記，指出裝置是否已連接到 AWS IoT Core。裝置 `"mything1"` 在 POSIX 時間 `1556649874716` 中斷連線 (`false`)：

```
"connectivity": {
    "connected":false,
    "timestamp":1556649874716
}
```

裝置 `"mything2"` 在 POSIX 時間 `1556649855046` 連線 (`true`)：

```
"connectivity": {
    "connected":true,
    "timestamp":1556649855046
}
```

時間戳記是以從 Epoch 算起的毫秒提供，因此 `1556649855046` 代表 2019 年 4 月 30 日星期二下午 6:44:15.046 (GMT)。

> **Important**
>
> 如果裝置已中斷連線大約一小時，則可能會缺少連線狀態的 `"timestamp"` 值。

法規與限制

這些是 `AWS_Things` 的限制。

類型複雜的影子欄位

只有當欄位的值為簡單類型、不包含陣列的 JSON 物件，或完全由簡單類型組成的陣列時，才會建立該影子欄位的索引。簡單類型是指字串、數值或下列其中一個常值：`true` 或 `false`。例如，假設下列影子狀態，欄位 `"palette"` 的值不會建立索引，因為它是包含複雜類型項目的陣列。而因為該陣列中的每個值皆為字串，所以 `"colors"` 欄位的值會建立索引。

```
{
    "state": {
        "reported": {
            "switched": "ON",
            "colors": [ "RED", "GREEN", "BLUE" ],
            "palette": [
                {
                    "name": "RED",
                    "intensity": 124
                },
                {
                    "name": "GREEN",
                    "intensity": 68
                },
                {
                    "name": "BLUE",
                    "intensity": 201
                }
            ]
        }
    }
}
```

巢狀影子欄位名稱

巢狀影子欄位的名稱會儲存為以句號 (.) 分隔的字串。例如，假設影子文件為：

```
{
  "state": {
    "desired": {
      "one": {
        "two": {
          "three": "v2"
        }
      }
    }
  }
}
```

`three` 欄位的名稱會儲存為 `desired.one.two.three`。如果您也有類似這個的影子文件：

```
{
  "state": {
    "desired": {
      "one.two.three": "v2"
    }
  }
}
```

同時符合 `shadow.desired.one.two.three:v2` 的查詢。最佳實務是不要在影子欄位名稱中使用句點。

影子中繼資料

只有當建立影子的 `"state"` 區段中之對應欄位的索引時，才會建立影子中繼資料區段中之欄位的索引。(在前面的範例中，也不會建立影子中繼資料區段中 `"palette"` 欄位的索引。)

未登錄的影子

如果您使用 UpdateThingShadow 建立影子，而用於建立影子的實物名稱尚未在您的 AWS IoT 帳戶中登錄，即不會建立此影子中欄位的索引。

數值

如果服務將任何登錄或影子資料辨識為數值，則會因其為數值而建立索引。您可以對數值運用包含範圍與比較運算子的查詢 (例如 `"attribute.foo<5"` 或 `"shadow.reported.foo:[75 TO 80]"`)。若要讓資料的值可識別為數值，其必須為有效的 JSON 數字類型常值 (介於 -2^53...2^53-1 之間的整數，或使用選擇性指數表示法的雙精度浮點)，或必須屬於僅包含這類值的陣列。

Null 值

Null 值未編製索引。

彙總查詢的自訂欄位數量上限

5

彙總查詢所要求的百分位數上限。

100

授權

在 AWS IoT 的政策動作中，您可以將實物索引指定為資源 ARN，如下所示。

動作	資源
`iot:SearchIndex`	索引 ARN (例如，`arn:aws:iot:<your-aws-region><your-aws-account>:index/AWS_Things`)。
`iot:DescribeIndex`	索引 ARN (例如，`arn:aws:iot:<your-aws-region>:index/AWS_Things`)。

> **Note**
>
> 如果您有查詢機群索引的許可，您就可以存取整個機群的物件資料。

管理實物群組索引

`AWS_ThingGroups` 是包含所有實物群組的索引。您可以使用此索引，根據群組名稱、描述、屬性和所有父系群組名稱搜尋群組。

啟用實物群組索引

您可以在 UpdateIndexingConfiguration API 中使用 `thing-group-indexing-configuration` 設定，來建立 `AWS_ThingGroups` 索引並控制其組態。您可以使用 GetIndexingConfiguration API，擷取目前的索引組態。

使用 get-indexing-configuration CLI 命令，擷取目前實物和實務群組索引組態。

aws iot get-indexing-configuration

```
{
   "thingGroupIndexingConfiguration": {
        "thingGroupIndexingMode": "ON"
    }
}
```

使用 update-indexing-configuration CLI 命令，更新實物群組索引的組態。

```
aws iot update-indexing-configuration --thing-group-indexing-configuration
 thingGroupIndexingMode=ON
```

> **Note**
>
> 您也可以單一命令，如下所示，更新實物和實務群組索引的組態。
>
> ```
> aws iot update-indexing-configuration --thing-indexing-configuration
> thingIndexingMode=REGISTRY --thing-group-indexing-configuration
> thingGroupIndexingMode=ON
> ```

以下是 `thingGroupIndexingMode` 的有效值。

OFF

不建立索引/刪除索引。

ON

建立或設定 `AWS_ThingGroups` 索引。

描述群組索引

使用 describe-index CLI 命令，擷取 `AWS_ThingGroups` 索引的目前狀態。

```
aws iot describe-index --index-name "AWS_ThingGroups"
{
   "indexStatus": "ACTIVE",
   "indexName": "AWS_ThingGroups",
   "schema": "THING_GROUPS"
}
```

您首次啟用索引後，AWS IoT 即會為您建立索引。若 `indexStatus` 為 `BUILDING`，您無法在該索引內進行查詢。

查詢實物群組索引

使用 search-index CLI 命令來查詢索引中的資料：

```
aws iot search-index --index-name "AWS_ThingGroups" --query-string
 "thingGroupName:mythinggroup*"
```

授權

在 AWS IoT 的政策動作中，您可以將實物群組索引指定為資源 ARN，如下所示。

動作	資源
`iot:SearchIndex`	索引 ARN (例如，`arn:aws:iot:`*`<your-aws-region>`*`:index/AWS_ThingGroups`)。
`iot:DescribeIndex`	索引 ARN (例如，`arn:aws:iot:`*`<your-aws-region>`*`:index/AWS_ThingGroups`)。

彙總資料的查詢

AWS IoT 提供三個 API (`GetStatistics`, `GetCardinality`、和 `GetPercentiles`)，可讓您在裝置機群中搜尋彙總資料。

GetStatistics

GetStatistics API 和 get-statistics CLI 命令會針對指定的彙總欄位傳回計數、平均值、總和、最小值、最大值、平方和、方差和標準差。

get-statistics CLI 命令會使用下列參數：

`index-name`

要搜尋的索引名稱。預設值為 `AWS_Things`。

`query-string`

用於搜尋索引的查詢。您可以指定 `"*"`，以取得 AWS 帳戶中所有索引實物的計數。

`aggregationField`

選用。要彙總的欄位。此欄位必須是您在呼叫 update-indexing-configuration 時定義的受管或自訂欄位。如果你未指定彙總欄位，則 `registry.version` 會用作彙總欄位。

`query-version`

要使用的查詢版本。預設值為 `2017-09-30`。

彙總欄位的類型可以影響傳回的統計資料。

搭配字串值的 GetStatistics

如果您在字串欄位上進行彙總，則呼叫 `GetStatistics` 會傳回多少裝置具有符合查詢的屬性。例如：

aws iot get-statistics --aggregation-field 'attributes.stringAttribute' --query-string '*'

此命令會傳回多少裝置包含稱為 `stringAttribute` 的屬性：

```
{
  "statistics": {
    "count": 3
  }
}
```

搭配布林值的 GetStatistics

當你使用布林彙總欄位呼叫 `GetStatistics` 時：

- AVERAGE 是符合查詢的裝置百分比。
- 根據以下規則，MINIMUM 為 0 或 1：
  - 如果彙總欄位的所有值是 `false`，MINIMUM 為 0。
  - 如果彙總欄位的所有值是 `true`，MINIMUM 為 1。
  - 如果彙總欄位的值是 `false` 和 `true` 的混合，MINIMUM 為 0。
- 根據以下規則，MAXIMUM 為 0 或 1：
  - 如果彙總欄位的所有值是 `false`，MAXIMUM 為 0。
  - 如果彙總欄位的所有值是 `true`，MAXIMUM 為 1。
  - 如果彙總欄位的值是 `false` 和 `true` 的混合，MAXIMUM 為 0。

- SUM 是相當於布林值的整數和。
- COUNT 是符合查詢的物件數量。

搭配數值的 GetStatistics

當您呼叫 `GetStatistics` 並指定類型為 `Number` 的彙總欄位時，`GetStatistics` 會傳回下列值：

計數

具有符合查詢之欄位的裝置數量。

average

符合查詢之數值的平均值。

sum

符合查詢之數值的總和。

minimum

符合查詢的最小數值。

maximum

符合查詢的最大數值。

sumOfSquares

符合查詢之數值的平方和。

variance

符合查詢之數值的方差。一組值的方差是每個值與集合平均值之平方差的平均值。

stdDeviation

符合查詢之數值的標準差。一組值的標準差是值如何分佈的度量。

以下範例展示如何使用數字自訂欄位呼叫 get-statistics。

aws iot get-statistics --aggregation-field 'attributes.numericAttribute2' --query-string '*'

```
{
  "statistics": {
    "count": 3,
    "average": 33.333333333333336,
    "sum": 100.0,
    "minimum": -125.0,
    "maximum": 150.0,
    "sumOfSquares": 43750.0,
    "variance": 13472.22222222222,
    "stdDeviation": 116.06990230986766
  }
}
```

對於數值彙總欄位，如果欄位值超過最大雙精度值，則統計資料為空白

GetCardinality

GetCardinality API 和 get-cardinality CLI 命令會傳回符合查詢之 唯一值的近似計數。例如，您可能想要尋找電池電量低於 50% 的裝置數量：

aws iot get-cardinality --index-name AWS_Things --query-string "batterylevel > 50" --aggregation-field "shadow.reported.batterylevel"。

這個命令會傳回電池電量超過 50% 的物件數量：

```
{
    "cardinality": 100
}
```

`cardinality` 一律由 get-cardinality 傳回，即使沒有相符的欄位。例如：

aws iot get-cardinality --query-string "thingName:Non-existent*" --aggregation-field "attributes.customField_STR"

```
{
    "cardinality": 0
}
```

get-cardinality CLI 命令會使用下列參數：

`index-name`

要搜尋的索引名稱。預設值為 `AWS_Things`。

`query-string`

用於搜尋索引的查詢。您可以指定 `"*"`，以取得 AWS 帳戶中所有索引實物的計數。

`aggregationField`

要彙總的欄位。

`query-version`

要使用的查詢版本。預設值為 `2017-09-30`。

GetPercentiles

GetPercentiles API 和 get-percentiles CLI 命令會將符合查詢的彙總值分組為百分位數群組。雖然預設的百分位數群組是：1,5,25,50,75,95,99，但是您可以在呼叫 `GetPercentiles` 時指定自己的百分位數群組。此函數會傳回所指定之每個百分位數群組 (預設百分位數群組) 的值。百分位數群組 "1" 包含的彙總欄位值，大約發生在符合查詢的百分之一值。百分位數群組 "5" 包含的彙總欄位值，大約發生在符合查詢的百分之五值，依此類推。結果是一個近似值，符合查詢的值越多，百分位數值就越精確。

以下範例顯示如何呼叫 get-percentiles CLI 命令。

aws iot get-percentiles --query-string "thingName:*" --aggregation-field "attributes.customField_NUM" --percents 10 20 30 40 50 60 70 80 90 99

```
{
    "percentiles": [
        {
            "value": 3.0,
            "percent": 80.0
        },
        {
            "value": 2.5999999999999996,
            "percent": 70.0
        },
        {
            "value": 3.0,
            "percent": 90.0
        },
        {
            "value": 2.0,
```

```
            "percent": 50.0
        },
        {
            "value": 2.0,
            "percent": 60.0
        },
        {
            "value": 1.0,
            "percent": 10.0
        },
        {
            "value": 2.0,
            "percent": 40.0
        },
        {
            "value": 1.0,
            "percent": 20.0
        },
        {
            "value": 1.4,
            "percent": 30.0
        },
        {
            "value": 3.0,
            "percent": 99.0
        }
    ]
}
```

以下命令顯示在沒有相符文件時從 get-percentiles 傳回的輸出。

aws iot get-percentiles --query-string "thingName:Non-existent*" --aggregation-field "attributes.customField_NUM"

```
{
    "percentiles": []
}
```

get-percentile CLI 命令會使用下列參數：

`index-name`

要搜尋的索引名稱。預設值為 `AWS_Things`。

`query-string`

用於搜尋索引的查詢。您可以指定 `"*"`，以取得 AWS 帳戶中所有索引實物的計數。

`aggregationField`

要彙總的欄位，其類型必須是 `Number`。

`query-version`

要使用的查詢版本。預設值為 `2017-09-30`。

`percents`

選用。您可以使用此參數來指定自訂百分位數群組。

疑難排解彙總查詢

如果您發生類型不符錯誤，您可以使用 CloudWatch Logs 來疑難排解問題。 CloudWatch Logs 必須先啟用，然後機群索引服務才能寫入日誌。如需詳細資訊，請參閱 CloudWatch Logs。

當您對非受管欄位進行彙總查詢時，您只能指定您在 `customFields` 引數中定義的欄位傳遞給 `UpdateIndexingConfiguration` 或 update-indexing-configuration。如果欄位值與設定的欄位資料類型不一致，則在您執行彙總查詢時會忽略此值。

當欄位無法編製索引，因為類型不符時，機群索引服務會發出錯誤日誌至 CloudWatch Logs。錯誤日誌包含欄位名稱、無法轉換的值，以及裝置的物件名稱。以下是範例錯誤日誌：

```
{
  "timestamp": "2017-02-20 20:31:22.932",
  "logLevel": "ERROR",
  "traceId": "79738924-1025-3a00-a669-7bec69f7f07a",
  "accountId": "000000000000",
  "status": "SucceededWithIssues",
  "eventType": "IndexingCustomFieldFailed",
  "thingName": "thing0",
  "failedCustomFields": [
    {
      "Name": "attributeName1",
      "Value": "apple",
      "ExpectedType": "String"
    },
    {
      "Name": "attributeName2",
      "Value": "2",
      "ExpectedType": "Boolean"
    }
  ]
}
```

如果裝置已中斷連線大約一小時，則可能會缺少連線狀態的 `timestamp` 值。若為持久工作階段，當用戶端中斷連線時間超過為持久工作階段所設的存留時間 (TTL) 後，則可能會缺少此值。只有針對用戶端 ID 有相符實物名稱的連線，連線狀態資料才會建立索引。(用戶端 ID 是用於將裝置連線到 AWS IoT Core 的值。)

查詢語法

查詢是使用查詢語法指定的。

查詢語法支援以下功能。

- 詞彙和字詞
- 搜尋欄位
- 字首搜尋
- 範圍搜尋
- 布林值運算子 `AND`、`OR`、`NOT` 以及 `-` (連字號是用於從搜尋結果排除某些項目，例如 `thingName:(tv* AND -plasma)`)
- 分組
- 欄位分組
- 逸出特殊字元

查詢語法不支援以下功能：

- 字首萬用字元搜尋 (例如 "*xyz")，但若搜尋 "*"，所有物件都會相符
- 常規表達式

- 提升
- 排名
- 模糊搜尋
- 鄰近搜尋
- 排序
- 聚合

查詢語言的幾件注意事項：

- 預設運算子為 AND。對 “`thingName:abc thingType:xyz`” 的查詢等同於 “`thingName:abc AND thingType:xyz`”。
- 若未指定欄位，AWS IoT 將於所有欄位搜尋該詞彙。
- 所有欄位名稱皆會區分大小寫。
- 搜尋不區分大小寫。文字以 Java `Character.isWhitespace(int)` 定義的空白字元分隔。
- 裝置影子資料的索引包含已呈報、所需、差量和中繼資料部分。
- 無法搜尋裝置影子和登錄版本，但回應中可看見這些資訊。
- 一條查詢的詞彙數量上限為 5。

範例實物查詢

查詢是以使用查詢語法的字串指定，並傳送至 `SearchIndex` API。下表列出部分查詢字串範例。

查詢字串	結果
abc	查詢任何登錄檔或影子欄位中的 "abc"。
thingName:myThingName	可查詢名為「myThingName」的物件。
thingName:my*	可查詢名稱起始為「my」的物件。
thingName:ab?	可查詢具有「ab」加上一個額外字元的實物名稱 (例如：「aba」、「abb」、「abc」等)。
thingTypeName:aa	查詢與類型 aa 關聯的項目。
attributes.myAttribute:75	可查詢具有「myAttribute」屬性名稱的實物，且其值為 75。
attributes.myAttribute:[75 TO 80]	可查詢具有「myAttribute」屬性名稱的實物，且其值介於一組數字範圍 75 – 80 (含) 內。
attributes.myAttribute:{75 TO 80]	可查詢具有「myAttribute」屬性名稱的實物，且其值介於一組數字範圍 (>75 且 <=80) 內。
attributes.serialNumber:["abcd" TO "abcf"]	可查詢具有「serialNumber」屬性名稱的實物，且其值介於一組字母和數字的字串範圍內。以此查詢而言，傳回的實物必須具有「serialNumber」屬性且其值為「abcd」、「abce」或「abcf」。
attributes.myAttribute:i*t	可查詢具有「myAttribute」屬性名稱的實物，且其值以「i」為首、後方接任意數量的字元，最後則是「t」。

查詢字串	結果
attributes.attr1:abc AND attributes.attr2<5 NOT attributes.attr3>10	使用布林值運算式，查詢結合詞彙的物件。以此查詢而言，傳回的實物必須具有「attr1」屬性名稱且其值為「abc」、具有「attr2」屬性名稱且其值小於 5 以及具有「attr3」屬性名稱且其值不超過 10。
shadow.hasDelta:true	可查詢影子具有差量元素的物件。
非 attributes.model:legacy	可查詢屬性名稱為「model」而不是「legacy」的實物。
shadow.reported.stats.battery:{70 TO 100} (v2 OR v3) NOT attributes.model:legacy	可查詢符合下列標準的物件： • 物件影子 `stats.battery`屬性值介於 70 和 100 內。 • 物件名稱、類型名稱或屬性值出現文字「v2」或「v3」。 • 物件 `model`屬性未設定為「legacy」。
shadow.reported.myvalues:2	查詢影子回報區段中之 `myvalues`陣列包含值為 2 的物件。
shadow.reported.location:* NOT shadow.desired.stats.battery:*	可查詢符合下列標準的物件： • `location`屬性存在於影子的 `reported` 區段中。 • `stats.battery`屬性不存在於影子的 `desired` 區段中。
connectivity.connected:true	查詢所有已連線的裝置。
connectivity.connected:false	查詢所有中斷連線的裝置。
connectivity.connected:true AND connectivity.timestamp : [1557651600000 TO 1557867600000]	查詢連線時間戳記 >= 1557651600000 且 <= 1557867600000 的所有連線裝置。時間戳記是以從 Epoch 算起的毫秒提供。
connectivity.connected:false AND connectivity.timestamp : [1557651600000 TO 1557867600000]	查詢中斷連線時間戳記 >= 1557651600000 且 <= 1557867600000 的所有中斷連線裝置。時間戳記是以從 Epoch 算起的毫秒提供。
connectivity.connected:true AND connectivity.timestamp > 1557651600000	查詢連線時間戳記 > 1557651600000 的所有連線裝置。時間戳記是以從 Epoch 算起的毫秒提供。
connectivity.connected:*	查詢具連線資訊的所有裝置。

範例實物群組查詢

查詢是以使用查詢語法的字串指定，並傳送至 `SearchIndex` API。下表列出部分查詢字串範例。

查詢字串	結果
abc	任何欄位的「abc」查詢。
thingGroupName:myGroupThingName	可查詢名為「myGroupThingName」的實物群組。

查詢字串	結果
thingGroupName:my*	可查詢名稱以「my」開頭的實物群組。
thingGroupName:ab?	可查詢具有「ab」加上一個額外字元的實物群組名稱 (例如：「aba」、「abb」、「abc」等)。
attributes.myAttribute:75	可查詢具有「myAttribute」屬性名稱的實物群組，且其值為 75。
attributes.myAttribute:[75 TO 80]	可查詢具有「myAttribute」屬性名稱的實物群組，且其值介於一組數字範圍 75 – 80 (含) 內。
attributes.myAttribute:[75 TO 80]	可查詢具有「myAttribute」屬性名稱的實物群組，且其值介於一組數字範圍 (>75 且 <=80) 內。
attributes.myAttribute:["abcd" TO "abcf"]	可查詢具有「myAttribute」屬性名稱的實物群組，且其值介於一組字母和數字的字串範圍內。以此查詢而言，傳回的實物群組必須具有「serialNumber」屬性且其值為「abcd」、「abce」或「abcf」。
attributes.myAttribute:i*t	可查詢具有「myAttribute」屬性名稱的實物群組，且其值以「i」為首、後方接任意數量的字元，最後則是「t」。
attributes.attr1:abc AND attributes.attr2<5 NOT attributes.attr3>10	使用布林值表達式，查詢結合詞彙的實物群組。以此查詢而言，傳回的實物群組必須具有「attr1」屬性名稱且其值為「abc」、具有「attr2」屬性名稱且其值小於 5 以及具有「attr3」屬性名稱且其值不超過 10。
NOT attributes.myAttribute:cde	可查詢具有「myAttribute」屬性名稱的實物群組，且不是「cde」。
parentGroupNames:(myParentThingGroupName)	可查詢其父群組名稱與「myParentThingGroupName」相符的實物群組。
parentGroupNames:(myParentThingGroupName OR myRootThingGroupName)	可查詢其父群組名稱與「myParentThingGroupName」或「myRootThingGroupName」相符的實物群組。
parentGroupNames:(myParentThingGroupNa*)	可查詢其父群組名稱以「myParentThingGroupNa」開頭的實物群組。

AWS IoT Device Defender

AWS IoT Device Defender 是一項安全服務，可讓您稽核裝置組態、監控連線裝置以偵測異常行為，並且防範安全風險。它可讓您的 AWS IoT 裝置機群實施一致的安全政策，並在裝置受到入侵時快速回應。

IoT 機群可由大量的裝置組成具有多樣化的功能、長時間在線上，並且散佈在多個地理位置。這些特點使叢集設定複雜且極易出錯。由於裝置通常受限於運算能力、記憶體和儲存功能，因此限制了裝置本身的加密和其他形式的安全性的使用。此外，裝置通常會使用具有已知漏洞的軟體。這些因素讓 IoT 機群成為對駭客極具吸引力的目標，且難以持續保護裝置機群。

AWS IoT Device Defender 透過提供識別安全問題和偏離最佳實務的工具來應對這些挑戰。 AWS IoT Device Defender 可以稽核裝置機群，以確保期遵守安全最佳實務並偵測裝置上的異常行為。

> **Note**
>
> AWS IoT Device Defender 未在 China (Beijing) Region 中提供。

稽核

AWS IoT Device Defender 稽核會查看帳戶與裝置相關的設定及政策，以確保安全性措施皆已就位。稽核可協助您偵測任何來自安全性最佳實務或存取政策的不一致情況 (例如，多個裝置使用相同的身分、或者允許一個裝置讀取並更新多個其他裝置的資料之過多許可政策)。您可以根據需要執行稽核 (隨需稽核) 或安排定期執行 (排程稽核)。

AWS IoT Device Defender 稽核會執行一組預先定義的檢查，以確認常見 IoT 安全最佳實務以及裝置漏洞。預先定義的檢查範例的政策包括在多個裝置授與讀取或更新資料的許可、分享身分的裝置 (X.509 憑證)，或即將過期或是已撤回但仍使用中的憑證。

問題嚴重性

問題嚴重性指出與每個已識別的不合規執行個體相關聯的關注層級，以及建議的修復時間。

嚴重

具有此嚴重性的不合規稽核檢查會識別需要緊急注意的問題。嚴重問題通常可讓惡意人士不需很精明，也無需內幕知識或特殊登入資料，即可輕鬆存取或控制您的資產。

高

具有此嚴重性的不合規範稽核檢查需要在嚴重問題解決之後進行緊急調查和補救規劃。與嚴重問題一樣，高嚴重性問題通常可讓惡意人士存取或控制您的資產。不過，高嚴重性問題通常更難以利用。它們可能需要特殊工具、內幕知識或特定設定。

中性

具有此嚴重性的不合規稽核檢查會呈現需要注意的問題，作為持續安全狀態維護的一部分。中嚴重性問題可能會造成負面的操作影響，例如由於安全性控制故障所造成的意外中斷。這些問題也可能讓惡意人士有限度地存取或控制您的資產，或者可能會促進其部分惡意行為。

低

具有此嚴重性的不合規稽核檢查通常表示安全最佳實務遭到忽略或略過。雖然它們可能不會對自己造成即時的安全影響，但這些失誤可能會被惡意人士利用。與中嚴重性問題一樣，低嚴重性問題需要注意，作為持續安全狀態維護的一部分。

稽核檢查

> **Note**
>
> 啟用檢查後，就立即開始收集資料。如果您的帳戶中有大量資料需要收集，那麼在您啟用檢查後，該檢查結果可能會有一段時間無法使用。

支援下列稽核檢查：

REVOKED_CA_CERT_CHECK

CA 憑證已撤回，但在 AWS IoT 中仍處於作用中的狀態。

嚴重性：關鍵

Details

CA 憑證在發行機構維護的憑證撤回清單中標記為已撤銷，但在 AWS IoT 中仍標記為 ACTIVE 或 PENDING_TRANSFER。

當此檢查發現不合規的 CA 憑證時，將會傳回下列原因代碼：

- CERTIFICATE_REVOKED_BY_ISSUER

Why it matters

已撤銷的 CA 憑證不應再用來簽署裝置憑證。它可能因遭入侵而已經撤銷。新增的裝置內含使用此 CA 憑證簽署的憑證可能會造成安全威脅。

How to fix it

1. 在 AWS IoT 中使用 UpdateCACertificate 將 CA 憑證標示為 INACTIVE。您也可以使用緩解行動:
   - 套用 `UPDATE_CA_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。
2. 檢閱 CA 憑證遭撤回後的裝置憑證註冊活動，並考慮撤回任何可能在此時間內使用該憑證發行的裝置憑證。(使用 ListCertificatesByCA 列出由 CA 簽署的裝置憑證和 UpdateCertificate 撤銷的裝置憑證。)

DEVICE_CERTIFICATE_SHARED_CHECK

多個同時發生的連線使用相同的 X.509 憑證來向 AWS IoT 驗證。

嚴重性：關鍵

Details

啟用此檢查後就會立即開始資料收集，但至少會有兩小時不能使用檢查結果。

當執行一部分的隨需稽核時，此檢查將會查看稽核開始之前 31 天內裝置使用的連接憑證和用戶端 ID。關於排程稽核，此檢查將會查看從上次稽核執行的資料，以及本次執行個體開始的時間。如果於檢查的期間已採取措施以減少此種情況，請注意何時並行連線以確定是否仍有問題。

當此檢查發現不合規的憑證時，將會傳回下列原因代碼：

- CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES

此外，此檢查傳回的結果包含共享憑證的 ID、用戶端使用憑證的連線 ID 和連接/中斷時間。最近的結果最先列出。

Why it matters

每個裝置應有唯一的憑證以供 AWS IoT 進行驗證。當多個裝置使用相同憑證時，這可能表示裝置已遭入侵。其身分可能遭到複製而進一步危害系統。

How to fix it

確認裝置憑證未遭洩漏。若已遭洩漏，依照您的安全性最佳實務來減緩情況。

如果您使用相同的憑證在多個裝置上，您可能需要：

1. 佈建新的、唯一的憑證，並將它們連接到每個裝置。
2. 確認新憑證有效且裝置可使用這些憑證進行連線。
3. 在 AWS IoT 中使用 UpdateCertificate 將舊憑證標示為 REVOKED。您也可以使用緩解行動:
   - 套用 `UPDATE_DEVICE_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 套用 `ADD_THINGS_TO_THING_GROUP` 緩解行動來新增裝置到您可以採取行動的群組。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。
4. 從每個裝置分離舊憑證。

DEVICE_CERTIFICATE_KEY_QUALITY_CHECK

AWS IoT 客戶通常會依賴使用 X.509 憑證的 TLS 交互身分驗證來對 AWS IoT 訊息中介裝置進行驗證。這些憑證及其憑證授權單位憑證必須在其 AWS IoT 帳戶中註冊，然後才能使用它們。在註冊這些憑證時，AWS IoT 會對這些憑證執行基本的例行性檢查。這些檢查包括：

- 它們必須是有效的格式
- 它們必須由註冊的憑證授權單位簽署
- 它們仍然必須在其有效期間內 (換言之，它們沒有過期)
- 其加密金鑰大小必須符合所需大小下限 (若為 RSA 金鑰，必須是 2048 位元或更大)。

此稽核檢查會提供下列額外測試，來測試您的加密金鑰品質：

- CVE-2017-15361 - 檢查是否已在 Debian 型作業系統上使用 OpenSSL 0.9.8c-1 以上但不超過 0.9.8g-9 的版本產生金鑰。這些版本的 OpenSSL 使用隨機數字產生器，產生可預測的數字，使遠端攻擊者更容易對加密金鑰進行暴力猜測攻擊。
- CVE-2008-0166 - 檢查是否已使用 Infineon 信賴平台模組 (TPM) 韌體中的 Infineon RAS 程式庫 1.02.013 產生金鑰，例如 0000000000000422 - 4.34 之前的版本、000000000000062b - 6.43 之前的版本，以及 0000000000008521 - 133.33 之前的版本。該程式庫不當處理 RSA 金鑰產生，使攻擊者更容易透過目標攻擊打敗一些加密保護機制。受影響的技術範例包括搭配 TPM 1.2 的 BitLocker、YubiKey 4 (4.3.5 之前) PGP 金鑰產生，以及 Chrome 作業系統中的快取使用者資料加密功能。

如果憑證無法通過測試，則 AWS IoT Device Defender 會將這些憑證報告為不合規。

嚴重性：關鍵

Details

此檢查適用於 ACTIVE 或 PENDING_TRANSFER 的裝置憑證。

當此檢查發現不合規的憑證時，將會傳回下列原因代碼：

- CERTIFICATE_KEY_VULNERABILITY_CVE-2017-15361

  CERTIFICATE_KEY_VULNERABILITY_CVE-2008-0166

Why it matters

當裝置使用易受攻擊的憑證時，攻擊者可以更容易地危害該裝置。

How to fix it

更新您的裝置憑證，以取代具有已知弱點的憑證。

如果您使用相同的憑證在多個裝置上，您可能需要：

1. 佈建新的、唯一的憑證，並將它們連接到每個裝置。
2. 確認新憑證有效且裝置可使用這些憑證進行連線。
3. 在 AWS IoT 中使用 UpdateCertificate 將舊憑證標示為 REVOKED。您也可以使用緩解行動:
   - 套用 `UPDATE_DEVICE_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 套用 `ADD_THINGS_TO_THING_GROUP` 緩解行動來新增裝置到您可以採取行動的群組。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。
4. 從每個裝置分離舊憑證。

CA_CERTIFICATE_KEY_QUALITY_CHECK

AWS IoT 客戶通常會依賴使用 X.509 憑證的 TLS 交互身分驗證來對 AWS IoT 訊息中介裝置進行驗證。這些憑證及其憑證授權單位憑證必須在其 AWS IoT 帳戶中註冊，然後才能使用它們。在註冊這些憑證時，AWS IoT 會對這些憑證執行基本的例行性檢查，包括：

- 憑證是有效的格式。
- 憑證在其有效期間內 (換言之，未過期)。
- 其加密金鑰大小符合所需大小下限 (若為 RSA 金鑰，必須是 2048 位元或更大)。

此稽核檢查會提供下列額外測試，來測試您的加密金鑰品質：

- CVE-2017-15361 - 檢查是否已在 Debian 型作業系統上使用 OpenSSL 0.9.8c-1 以上但不超過 0.9.8g-9 的版本產生金鑰。這些版本的 OpenSSL 使用隨機數字產生器，產生可預測的數字，使遠端攻擊者更容易對加密金鑰進行暴力猜測攻擊。
- CVE-2008-0166 - 檢查是否已使用 Infineon 信賴平台模組 (TPM) 韌體中的 Infineon RAS 程式庫 1.02.013 產生金鑰，例如 0000000000000422 - 4.34 之前的版本、000000000000062b - 6.43 之前的版本，以及 0000000000008521 - 133.33 之前的版本。該程式庫不當處理 RSA 金鑰產生，使攻擊者更容易透過目標攻擊打敗一些加密保護機制。受影響的技術範例包括搭配 TPM 1.2 的 BitLocker、YubiKey 4 (4.3.5 之前) PGP 金鑰產生，以及 Chrome 作業系統中的快取使用者資料加密功能。

如果憑證無法通過測試，則 AWS IoT Device Defender 會將這些憑證報告為不合規。

嚴重性：關鍵

Details

此檢查適用於 ACTIVE 或 PENDING_TRANSFER 的 CA 憑證。

當此檢查發現不合規的憑證時，將會傳回下列原因代碼：

- CERTIFICATE_KEY_VULNERABILITY_CVE-2017-15361

  CERTIFICATE_KEY_VULNERABILITY_CVE-2008-0166

Why it matters

使用此 CA 憑證簽署的新增裝置可能會造成安全威脅。

How to fix it

1. 在 AWS IoT 中使用 UpdateCACertificate 將 CA 憑證標示為 INACTIVE。您也可以使用緩解行動:
   - 套用 `UPDATE_CA_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。
2. 檢閱 CA 憑證遭撤回後的裝置憑證註冊活動，並考慮撤回任何可能在此時間內使用該憑證發行的裝置憑證。(使用 ListCertificatesByCA 列出由 CA 簽署的裝置憑證和 UpdateCertificate 撤銷的裝置憑證。)

UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

附加到未經驗證的 Amazon Cognito 身分集區角色是過於寬鬆的政策，因為它會授與執行以下任何 AWS IoT 動作的許可：

- 管理或修改實物。
- 讀取實物管理資料。
- 管理與實物不相關的資料或資源。

或者，因為它授與可在廣泛的裝置上執行以下 AWS IoT 動作的許可：

- 使用 MQTT 連接、發佈或訂閱保留的主題 (包括影子或工作執行資料)。
- 使用 API 命令讀取或修改影子或工作執行資料。

一般而言，使用未經驗證 Amazon Cognito 身分集區角色連線的裝置應該只有受限的許可來發佈和訂閱特定實物的 MQTT 主題，或使用 API 命令來讀取和修改影子或工作執行資料的特定實物相關資料。

嚴重性：關鍵

Manage or modify things

以下 AWS IoT API 動作用來管理或修改物件。執行這些動作的許可不應授與透過未經驗證的 Amazon Cognito 身分集區連接的裝置：

- `AddThingToThingGroup`
- `AttachThingPrincipal`
- `CreateThing`

- `DeleteThing`
- `DetachThingPrincipal`
- `ListThings`
- `ListThingsInThingGroup`
- `RegisterThing`
- `RemoveThingFromThingGroup`
- `UpdateThing`
- `UpdateThingGroupsForThing`

授與許可的任何角色執行這些動作，即使是單一資源都是視為不合規。

Read thing administrative data

以下 AWS IoT API 動作用來讀取或修改物件資料。透過未經驗證的 Amazon Cognito 身分集區連接的裝置，不應授與其執行這些動作的許可。

- `DescribeThing`
- `ListJobExecutionsForThing`
- `ListThingGroupsForThing`
- `ListThingPrincipals`

範例：

- 不合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DescribeThing",
          "iot:ListJobExecutionsForThing",
          "iot:ListThingGroupsForThing",
          "iot:ListThingPrincipals"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing"
      ]
    }
  ]
}
```

這可讓裝置執行指定的動作，即使僅授與一個物件。

Manage non-things

透過未經驗證 Amazon Cognito 身分集區連接的裝置，不應被授與許可來執行非這些章節中所討論的任何其他 AWS IoT API 動作。為了利用透過未經驗證 Amazon Cognito 身分集區連接的應用程式管理您的帳戶，請建立裝置尚未使用的單獨身分集區。

Subscribe/publish to MQTT topics

會透過 AWS IoT 訊息中介裝置傳送 MQTT 訊息，並裝置會使用該訊息來執行許多動作，包括存取和修改影子狀態和工作執行狀態。與裝置連接、發佈或訂閱 MQTT 訊息許可的政策，應該對特定資源加以限制這些動作如下所示：

連線

- 不合規：

```
arn:aws:iot:<region>:<account-id>:client/*
```

萬用字元 * 可讓任何裝置連接到 AWS IoT。

```
arn:aws:iot:<region>:<account-id>:client/${iot:ClientId}
```

除非 `iot:Connection.Thing.IsAttached` 在條件金鑰中設定為 true，這相當於先前範例中的萬用字元 *。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:client/
${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

資源規範包含與連接所用的裝置名稱相符的變數。條件陳述式會檢查 MQTT 用戶端所用的憑證是否與連接到具有所用名稱的實物相符，進一步限制許可。

發布

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*/shadow/update
```

這可讓裝置更新任何裝置 (* = 所有裝置) 的陰影。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*
```

這可讓裝置讀取、更新或刪除任何裝置 (* = 所有裝置) 的陰影。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Publish" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
      ],
    }
  ]
```

```
}
```

資源規格包含萬用字元，但只符合任何陰影相關主題物件名稱連線的裝置。

訂閱

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這可讓裝置為所有裝置訂閱保留的陰影或工作主題。

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

前述範例相同，但使用 # 萬用字元。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/+/shadow/update
```

這可讓裝置看到任何裝置 (+ = 所有裝置) 的陰影更新。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Subscribe" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
      ],
    }
  ]
}
```

資源規格包含萬用字元，但只會針對實物名稱用於連線的裝置，比對任何影子相關主體與任何工作相關主題。

接收

- 合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這是允許的，因為裝置只能從已訂閱許可的主題接收訊息。

Read/modify shadow or job data

授與裝置執行 API 動作的許可以存取或修改裝置影子或工作執行資料的政策，應該於特定的資源限制這些動作。API 動作為：

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`

- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

範例：

- 不合規：

```
arn:aws:iot:<region>:<account-id>:thing/*
```

這可讓裝置在任何物件上執行指定的動作。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DeleteThingShadow",
          "iot:GetThingShadow",
          "iot:UpdateThingShadow",
          "iot:DescribeJobExecution",
          "iot:GetPendingJobExecutions",
          "iot:StartNextPendingJobExecution",
          "iot:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing1",
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing2"
      ]
    }
  ]
}
```

這可讓裝置僅在兩個實物上執行指定的動作。

Details

對此檢查，AWS IoT Device Defender 會稽核所有 Amazon Cognito 身分集區在稽核執行前的 31 天內用於連接到 AWS IoT 訊息中介裝置。已驗證或未驗證 Amazon Cognito 身分從中進行連線的所有 Amazon Cognito 身分集區都會納入稽核中。

當此檢查發現不合規未經驗證的 Amazon Cognito 身分集區角色時，將會傳回下列原因代碼：

- ALLOWS_ACCESS_TO_IOT_ADMIN_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Why it matters

由於未經驗證的使用者從未經過身分驗證，與經過身分驗證的 Amazon Cognito 身分相比其風險要大得多。如果未經驗證身分遭到入侵，則可以使用管理動作修改帳戶設定、刪除資源或獲得存取敏感資料。或者，通過廣泛的存取裝置設定，它可以存取或修改您帳戶中的所有裝置影子和工作。使用許可為的訪客使用者可能入侵您的整個叢集或啟動 DDOS 攻擊的訊息。

How to fix it

附加到未經驗證的 Amazon Cognito 身分集區角色的政策僅應授與裝置完成其工作所需的那些許可。建議下列步驟：

1. 建立新的合規角色。
2. 建立 Amazon Cognito 身分集區，並連接至合規角色。
3. 驗證您的身分可以存取 AWS IoT 以使用新的集區。
4. 驗證完成後，將合規角色連接至標記為不合規的 Amazon Cognito 身分集區。

您也可以使用緩解行動:

- 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

附加到經身分驗證的 Amazon Cognito 身分集區角色是過於寬鬆的政策，因為它授與執行以下 AWS IoT 動作的許可：

- 管理或修改實物。
- 管理與實物不相關的資料或資源。

或者，因為它授與可在廣泛的裝置上執行以下 AWS IoT 動作的許可：

- 讀取實物管理資料。
- 使用 MQTT 連接/發佈/訂閱保留的主題 (包括影子或工作執行資料)。
- 使用 API 命令讀取或修改影子或工作執行資料。

一般而言，使用已驗證 Amazon Cognito 身分集區角色連線的裝置應僅有受限的許可來讀取特定實物管理資料，發佈和訂閱特定的 MQTT 主題或使用 API 命令讀取和修改影子或工作執行資料相關的特定實物資料。

嚴重性：關鍵

Manage or modify things

下列 AWS IoT API 動作用來管理或修改實物，因此不應授與裝置透過已驗證的 Amazon Cognito 身分集區連接並執行這些許可：

- `AddThingToThingGroup`
- `AttachThingPrincipal`
- `CreateThing`
- `DeleteThing`
- `DetachThingPrincipal`
- `ListThings`
- `ListThingsInThingGroup`
- `RegisterThing`
- `RemoveThingFromThingGroup`
- `UpdateThing`

- `UpdateThingGroupsForThing`

授與許可的任何角色執行這些動作，即使是單一資源都是視為不合規。

Manage non-things

透過已驗證 Amazon Cognito 身分集區連接的裝置，不應被授與許可來執行非這些章節中所討論的任何其他 AWS IoT API 動作。為了利用透過已驗證 Amazon Cognito 身分集區連接的應用程式管理您的帳戶，請建立裝置尚未使用的單獨身分集區。

Read thing administrative data

下列 AWS IoT API 動作用來讀取實物資料，因此應授與裝置透過已驗證 Amazon Cognito 身分集區連接並僅在一組有限的實物上執行這些許可：

- `DescribeThing`
- `ListJobExecutionsForThing`
- `ListThingGroupsForThing`
- `ListThingPrincipals`

範例：

- 不合規：

```
arn:aws:iot:<region>:<account-id>:thing/*
```

  這可讓裝置在任何物件上執行指定的動作。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DescribeThing",
          "iot:ListJobExecutionsForThing",
          "iot:ListThingGroupsForThing",
          "iot:ListThingPrincipals"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing"
      ]
    }
  ]
}
```

  這可讓裝置僅在一個實物上執行指定的動作。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DescribeThing",
          "iot:ListJobExecutionsForThing",
          "iot:ListThingGroupsForThing",
```

```
          "iot:ListThingPrincipals"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing*"
      ]
    }
  ]
}
```

這是合規的，因為即使資源指定在特定字串之前使用萬用字元 (*)，並且限制了存取具有指定前綴名稱字首的實物組別。

Subscribe/publish to MQTT topics

會透過 AWS IoT 訊息中介裝置傳送 MQTT 訊息，並裝置會使用該訊息來執行許多不同的動作，包括存取和修改影子狀態和工作執行狀態。與裝置連接、發佈或訂閱 MQTT 訊息許可的政策，應該對特定資源加以限制這些動作如下所示：

連線

- 不合規：

```
arn:aws:iot:<region>:<account-id>:client/*
```

萬用字元 * 可讓任何裝置連接到 AWS IoT。

```
arn:aws:iot:<region>:<account-id>:client/${iot:ClientId}
```

除非 `iot:Connection.Thing.IsAttached` 在條件金鑰中設定為 true，這相當於先前範例中的萬用字元 *。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:client/
${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

資源規範包含與用來連接的裝置名稱符合的變數，和條件陳述式進一步限制許可的檢查，通過 MQTT 用戶端使用的憑證是否與附加到具有使用名稱的物件相符合。

發布

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*/shadow/update
```

這可讓裝置更新任何裝置 (* = 所有裝置) 的陰影。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*
```

這可讓裝置讀取/更新/刪除任何裝置 (* = 所有裝置) 的陰影。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Publish" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
      ],
    }
  ]
}
```

資源規格包含萬用字元，但只符合任何陰影相關主題物件名稱連線的裝置。

訂閱

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這可讓裝置為所有裝置訂閱保留的陰影或工作主題。

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/#
```

前述範例相同，但使用 # 萬用字元。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/+/shadow/update
```

這可讓裝置看到任何裝置 (+ = 所有裝置) 的陰影更新。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Subscribe" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
      ],
    }
  ]
}
```

資源規格包含萬用字元，但只會針對實物名稱用於連線的裝置，比對任何影子相關主體與任何工作相關主題。

接收

- 合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這是合規的，因為裝置只能從已訂閱許可的主題接收訊息。

Read or modify shadow or job data

授與裝置執行 API 動作的許可以存取或修改裝置影子或工作執行資料的政策，應該於特定的資源限制這些動作。API 動作為：

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`
- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

範例：

- 不合規：

```
arn:aws:iot:<region>:<account-id>:thing/*
```

這可讓裝置在任何物件上執行指定的動作。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DeleteThingShadow",
          "iot:GetThingShadow",
          "iot:UpdateThingShadow",
          "iot:DescribeJobExecution",
          "iot:GetPendingJobExecutions",
          "iot:StartNextPendingJobExecution",
          "iot:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing1",
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing2"
      ]
    }
  ]
}
```

這可讓裝置僅在兩個實物上執行指定的動作。

Details

對此檢查，AWS IoT Device Defender 會稽核所有 Amazon Cognito 身分集區在稽核執行前的 31 天內用於連接到 AWS IoT 訊息中介裝置。已驗證或未驗證 Amazon Cognito 身分從中進行連線的所有 Amazon Cognito 身分集區都會納入稽核中。

當此檢查發現不合規的已驗證 Amazon Cognito 身分集區角色時，將會傳回下列原因代碼：

- ALLOWS_BROAD_ACCESS_TO_IOT_THING_ADMIN_READ_ACTIONS
- ALLOWS_ACCESS_TO_IOT_NON_THING_ADMIN_ACTIONS
- ALLOWS_ACCESS_TO_IOT_THING_ADMIN_WRITE_ACTIONS

Why it matters

如果已驗證身分遭到入侵，則可以使用管理動作修改帳戶設定、刪除資源或獲得存取敏感資料。

How to fix it

附加到已驗證的 Amazon Cognito 身分集區角色的政策僅應授與裝置完成其工作所需的那些許可。建議下列步驟：

1. 建立新的合規角色。
2. 建立 Amazon Cognito 身分集區，並連接至合規角色。
3. 驗證您的身分可以存取 AWS IoT 以使用新的集區。
4. 驗證完成後，將角色連接至標記為不合規的 Amazon Cognito 身分集區。

您也可以使用緩解行動:

- 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

IOT_POLICY_OVERLY_PERMISSIVE_CHECK

AWS IoT 政策提供的許可太寬鬆或不受限制。它授與許可給一組廣泛的裝置傳送或接收 MQTT 訊息，或授與許可給一組廣泛的裝置存取或修改陰影和任務執行資料。

一般來說，裝置政策應授與其相關聯資源的存取許可，並且其他裝置不允許存取或是極少。在某些例外狀況下，使用萬用字元 (例如，「*」) 來指定資源的政策已視為太寬鬆或不受限制。

嚴重性：關鍵

MQTT permissions

會透過 AWS IoT 訊息中介裝置傳送 MQTT 訊息，並裝置會使用該訊息來執行許多動作，包括存取和修改影子狀態和工作執行狀態。與裝置連接、發佈或訂閱 MQTT 訊息許可的政策，應該對特定資源加以限制這些動作如下所示：

連線

- 不合規：

```
arn:aws:iot:<region>:<account-id>:client/*
```

萬用字元 * 可讓任何裝置連接到 AWS IoT。

```
arn:aws:iot:<region>:<account-id>:client/${iot:ClientId}
```

除非 `iot:Connection.Thing.IsAttached` 在條件金鑰中設定為 true，這相當於先前範例中的萬用字元 *。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:client/
${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

資源規範包含與連接所用的裝置名稱相符的變數。條件陳述式會檢查 MQTT 用戶端所用的憑證是否與連接到具有所用名稱的實物相符，進一步限制許可。

發布

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*/shadow/update
```

這可讓裝置更新任何裝置 (* = 所有裝置) 的陰影。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/*
```

這可讓裝置讀取、更新或刪除任何裝置 (* = 所有裝置) 的陰影。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Publish" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
      ],
    }
  ]
}
```

資源規格包含萬用字元，但只符合任何陰影相關主題物件名稱連線的裝置。

訂閱

- 不合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這可讓裝置為所有裝置訂閱保留的陰影或工作主題。

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

前述範例相同，但使用 # 萬用字元。

```
arn:aws:iot:<region>:<account-id>:topic/$aws/things/+/shadow/update
```

這可讓裝置看到任何裝置 (+ = 所有裝置) 的陰影更新。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Subscribe" ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
        "arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
      ],
    }
  ]
}
```

資源規格包含萬用字元，但只會針對實物名稱用於連線的裝置，比對任何影子相關主體與任何工作相關主題。

接收

- 合規：

```
arn:aws:iot:<region>:<account-id>:topicfilter/$aws/things/*
```

這是合規的，因為裝置只能從已訂閱許可的主題接收訊息。

Shadow and job permissions

授與裝置執行 API 動作的許可以存取或修改裝置影子或工作執行資料的政策，應該於特定的資源限制這些動作。API 動作為：

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`
- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

範例：

- 不合規：

```
arn:aws:iot:<region>:<account-id>:thing/*
```

  這可讓裝置在任何物件上執行指定的動作。

- 合規：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DeleteThingShadow",
          "iot:GetThingShadow",
          "iot:UpdateThingShadow",
          "iot:DescribeJobExecution",
          "iot:GetPendingJobExecutions",
          "iot:StartNextPendingJobExecution",
          "iot:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing1",
        "arn:aws:iot:<region>:<account-id>:/thing/MyThing2"
      ]
    }
  ]
}
```

  這可讓裝置僅在兩個實物上執行指定的動作。

Details

當此檢查發現不合規的 IoT 政策時，將會傳回下列原因代碼：

- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Why it matters

憑證、Amazon Cognito 身分或物件群組使用過多許可的政策，如果遭入侵會影響您的整個帳戶。攻擊者可以使用這類廣泛存取來讀取或修改您的所有裝置的影子、工作或工作執行。或者，攻擊者可以使用入侵的憑證連接到惡意裝置或啟動 DDOS 對您的網路進行攻擊。

How to fix it

遵循以下步驟以修正任何附加到實物、實物群組或其他實體不合規的政策：

1. 使用 CreatePolicyVersion 來建立合規的新政策版本。將 `setAsDefault` 旗標設為 true。(這可讓此新版本適用於使用該政策的所有實體。)
2. 使用 ListTargetsForPolicy 取得政策所附加至的目標清單 (憑證、實物群組)，以及判斷那些裝置包含在群組中或使用憑證來連接。
3. 驗證所有相關的裝置可以連接到 AWS IoT。如果裝置無法連接，則復原到預設政策之前的版本，使用 SetPolicyVersion 將預設政策還原為先前版本，修訂政策，然後重試。

您可以使用緩解動作：

- 套用 `REPLACE_DEFAULT_POLICY_VERSION` 緩解行動到稽核結果來產生此變更。

- 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

使用 AWS IoT 政策變數 動態參考您政策中的 AWS IoT 資源。

IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK

AWS IoT 角色別名提供一種機制，讓連線裝置使用 X.509 憑證來對 AWS IoT 進行驗證，然後從與 AWS IoT 角色別名相關聯的 IAM 角色取得短期 AWS 登入資料。必須搭配驗證內容變數使用存取政策來限定這些登入資料的許可。如果您的策略設定錯誤，您可能會讓自己受到權限提升的攻擊。此稽核檢查可確保 AWS IoT 角色別名所提供的暫時登入資料不會過於寬鬆。

如果找到下列其中一種情況，就會觸發此檢查：

- 政策將管理許可提供給角色別名 (例如，"iot:*"、"dynamodb:*"、 "iam:*" 等等) 在過去一年中所使用的任何服務。
- 政策可讓您廣泛存取物件中繼資料動作、存取受限制的 AWS IoT 動作，或廣泛存取 AWS IoT 資料平面動作。
- 政策可讓您存取安全稽核服務，例如 "iam"、"cloudtrail"、"guardduty"、"inspector" 或 "trustedadvisor"。

嚴重性：關鍵

Details

當此檢查發現不合規的 IoT 政策時，將會傳回下列原因代碼：

- ALLOWS_BROAD_ACCESS_TO_USED_SERVICES
- ALLOWS_ACCESS_TO_SECURITY_AUDITING_SERVICES
- ALLOWS_BROAD_ACCESS_TO_IOT_THING_ADMIN_READ_ACTIONS
- ALLOWS_ACCESS_TO_IOT_NON_THING_ADMIN_ACTIONS
- ALLOWS_ACCESS_TO_IOT_THING_ADMIN_WRITE_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Why it matters

藉由將許可限制為裝置執行正常作業所需的許可，您可以降低裝置遭到入侵時的帳戶風險。

How to fix it

遵循以下步驟以修正任何附加到實物、實物群組或其他實體不合規的政策：

1. 請遵循授權直接呼叫 AWS 服務 (p. 172)中的步驟，將更嚴格的政策套用至您的角色別名。

您可以使用緩解動作：

- 如果您要實作自訂動作以回應 Amazon SNS 訊息，請套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK

AWS IoT 角色別名提供一種機制，讓連線裝置使用 X.509 憑證來對 AWS IoT 進行驗證，然後從與 AWS IoT 角色別名相關聯的 IAM 角色取得短期 AWS 登入資料。必須搭配驗證內容變數使用存取政策來限定這些登入資料的許可。如果您的策略設定錯誤，您可能會讓自己受到權限提升的攻擊。此稽核檢查可確保 AWS IoT 角色別名所提供的暫時登入資料不會過於寬鬆。

如果角色別名可存取去年尚未用於 AWS IoT 裝置的服務，則會觸發此檢查。例如，稽核會報告您具有的 IAM 角色是否連結至過去一年僅使用 AWS IoT 的角色別名，但附加至角色的政策也會授與 `"iam:getRole"` 和 `"dynamodb:PutItem"` 的許可。

嚴重性：中

Details

當此檢查發現不合規的 AWS IoT 政策時，將會傳回下列原因代碼：

- ALLOWS_ACCESS_TO_UNUSED_SERVICES

Why it matters

藉由將許可限制為裝置執行正常作業所需的那些服務，您可以降低裝置遭到入侵時的帳戶風險。

How to fix it

遵循以下步驟以修正任何附加到實物、實物群組或其他實體不合規的政策：

1. 請遵循授權直接呼叫 AWS 服務 (p. 172)中的步驟，將更嚴格的政策套用至您的角色別名。

您可以使用緩解動作：

- 如果您要實作自訂動作以回應 Amazon SNS 訊息，請套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

CA_CERT_APPROACHING_EXPIRATION_CHECK

CA 憑證將於 30 天內到期或已到期。

嚴重性：中

Details

此檢查適用於 ACTIVE 或 PENDING_TRANSFER 的 CA 憑證。

當此檢查發現不合規的 CA 憑證時，將會傳回下列原因代碼：

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Why it matters

已到期的 CA 憑證不應使用於簽署新的裝置憑證。

How to fix it

請諮詢您的安全性最佳實務以了解如何進行。您可能想要：

1. 向 AWS IoT 註冊新的 CA 憑證。
2. 確認您可以使用新 CA 憑證來簽署裝置憑證。
3. 在 AWS IoT 中使用 UpdateCACertificate 將舊的 CA 憑證標示為 INACTIVE。您也可以使用緩解行動:
   - 套用 `UPDATE_CA_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

CONFLICTING_CLIENT_IDS_CHECK

多個裝置使用相同的用戶端 ID 連接。

嚴重性：高

Details

多個連線使用相同的客戶端 ID 進行連線，造成一個已連線的裝置遭到「中斷」。MQTT 規格僅允許每個客戶端 ID 有一個使用中的連線，因此當另一個裝置使用相同的客戶端 ID 連線時，就會將前一個裝置踢出連線。

當執行一部分的隨需稽核時，此檢查將會查看用戶端 ID 在稽核開始之前 31 天內如何使用連接。關於排程稽核，此檢查將會查看從上次稽核執行的資料，以及本次執行個體開始的時間。如果於檢查的期間已採取措施以減少此種情況，請注意何時連線/中斷以確定是否仍有問題。

當此檢查發現不合規時，將會傳回下列原因代碼：

- DUPLICATE_CLIENT_ID_ACROSS_CONNECTIONS

此檢查傳回的結果也包含使用來連接的用戶端 ID、委託人 ID 和中斷時間。最近的結果最先列出。

Why it matters

使用衝突 ID 的裝置會被迫持續重新連線，將可能會導致遺失訊息或讓裝置無法連線。

這可能表示裝置或裝置的登入資料已遭入侵，而且可能是 DDoS 攻擊的一部分。也可能是裝置的帳戶未設定正確或裝置連線狀況不佳，並且每分鐘幾次的強制重新連線。

How to fix it

在 AWS IoT 中將每個裝置註冊為獨特實物，並使用實物名稱做為要連接的用戶端 ID。或透過 MQTT 連接裝置時，使用 UUID 當做用戶端 ID。您也可以使用緩解行動:

- 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

DEVICE_CERT_APPROACHING_EXPIRATION_CHECK

裝置憑證將於 30 天內到期或已到期。

嚴重性：中

Details

此檢查適用於 ACTIVE 或 PENDING_TRANSFER 的裝置憑證。

當此檢查發現不合規的裝置憑證時，將會傳回下列原因代碼：

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Why it matters

不應使用過期後的裝置憑證。

How to fix it

請諮詢您的安全性最佳實務以了解如何進行。您可能想要：

1. 佈建新的憑證並將憑證與裝置連接。
2. 確認新憑證為有效且裝置可用來連線。
3. 在 AWS IoT 中使用 UpdateCertificate 將舊憑證標示為 INACTIVE。您也可以使用緩解行動:
   - 套用 `UPDATE_DEVICE_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 套用 `ADD_THINGS_TO_THING_GROUP` 緩解行動來新增裝置到您可以採取行動的群組。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

   如需更多詳細資訊，請參閱 緩解動作 (p. 559)。
4. 從裝置分離舊憑證。(請參閱 DetachThingPrincipal。)

REVOKED_DEVICE_CERT_CHECK

已撤回的裝置憑證仍然運作中。

嚴重性：中

Details

裝置憑證已在 CA 的憑證撤回清單中，但在 AWS IoT 卻仍為使用中。

此檢查適用於 ACTIVE 或 PENDING_TRANSFER 的裝置憑證。

當此檢查發現不合規時，將會傳回下列原因代碼：

- CERTIFICATE_REVOKED_BY_ISSUER

Why it matters

裝置憑證通常撤銷，因為它已遭入侵。由於錯誤或疏忽，在 AWS IoT 中可能尚未將它撤銷。

How to fix it

確認裝置憑證未遭洩漏。若已遭洩漏，依照您的安全性最佳實務來減緩情況。您可能想要：

1. 為各裝置佈建新的憑證。
2. 確認新憑證為有效且裝置可用來連線。
3. 在 AWS IoT 中使用 UpdateCertificate 將舊憑證標示為 REVOKED。您也可以使用緩解行動:
   - 套用 `UPDATE_DEVICE_CERTIFICATE` 緩解行動到稽核結果來產生此變更。
   - 套用 `ADD_THINGS_TO_THING_GROUP` 緩解行動來新增裝置到您可以採取行動的群組。
   - 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

4. 從裝置分離舊憑證。(請參閱 DetachThingPrincipal。)

LOGGING_DISABLED_CHECK

AWS IoT 日誌並未在 CloudWatch 中啟用。

嚴重性：低

Details

當此檢查發現不合規時，將會傳回下列原因代碼：

- LOGGING_DISABLED

Why it matters

CloudWatch 中的 AWS IoT 日誌可供查看 AWS IoT 內的行為，包括驗證失敗和意外連接和中斷，這可能表示裝置已遭入侵。

How to fix it

在 CloudWatch 中啟用 AWS IoT 日誌。請參閱監控工具。您也可以使用緩解行動:

- 套用 `ENABLE_IOT_LOGGING` 緩解行動到稽核結果來產生此變更。
- 如果您要實作自訂回應以回應 Amazon SNS 訊息，套用 `PUBLISH_FINDINGS_TO_SNS` 緩解行動。

如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

如何執行稽核

1. 為您的帳戶配置稽核設定。使用 UpdateAccountAuditConfiguration (p. 535) 啟用您想要用於稽核的這些檢查、設定選用通知以及設定許可。

   對於某些檢查，只要啟用檢查 AWS IoT 就會開始收集資料。
2. 建立一或多個稽核排程。使用 CreateScheduledAudit (p. 539) 指定您在稽核中要執行的檢查以及應執行這些稽核的頻率。

   或者，您可在需要時執行隨需稽核。使用 StartOnDemandAuditTask (p. 547) 指定您要執行的檢查並且立即開始執行稽核。(如果您最近啟用了隨需稽核包含的檢查，結果有時可能尚未準備好。)
3. 您可使用 AWS IoT 主控台來檢視稽核結果。

   或者，您可以使用 ListAuditFindings (p. 554) 查看稽核結果。使用此命令，您可以按檢查類型的結果篩選條件、特定資源或是稽核時間。您可以使用此資訊來減輕任何找到的問題。
4. 您可以套用緩解行動到您定義的 AWS 帳戶中任何不合規的結果。如需詳細資訊，請參閱套用緩解動作 (p. 565)。

通知

當稽核完成後，您可以使用每個稽核檢查結果的摘要來傳送 SNS 通知，包括找到不合規資源數量的詳細資訊。使用 `auditNotificationTargetConfigurations` 欄位輸入 UpdateAccountAuditConfiguration (p. 535) 命令。SNS 通知具有下列承載：

承載範例

```
{
    "accountId": "123456789012",
    "taskId": "4e2bcd1ccbc2a5dd15292a82ab80c380",
    "taskStatus": "FAILED|CANCELED|COMPLETED",
    "taskType": "ON_DEMAND_AUDIT_TASK|SCHEDULED_AUDIT_TASK",
    "scheduledAuditName": "myWeeklyAudit",
    "failedChecksCount": 0,
    "canceledChecksCount": 0,
    "nonCompliantChecksCount": 1,
    "compliantChecksCount": 0,
    "totalChecksCount": 1,
    "taskStartTime": 1524740766191,
    "auditDetails": [
        {
            "checkName": "DEVICE_CERT_APPROACHING_EXPIRATION_CHECK |
                          REVOKED_DEVICE_CERT_CHECK |
                          CA_CERT_APPROACHING_EXPIRATION_CHECK |
                          REVOKED_CA_CERT_CHECK |
                          DEVICE_CERTIFICATE_SHARED_CHECK |
                          IOT_POLICY_UNRESTRICTED_CHECK |
                          UNAUTHENTICATED_COGNITO_IDENTITY_UNRESTRICTED_ACCESS_CHECK |
                          AUTHENTICATED_COGNITO_IDENTITY_UNRESTRICTED_ACCESS_CHECK |
                          CONFLICTING_CLIENT_IDS_CHECK |
                          LOGGING_DISABLED_CHECK",

            "checkRunStatus": "FAILED |
                               CANCELED |
                               COMPLETED_COMPLIANT |
                               COMPLETED_NON_COMPLIANT",

            "nonCompliantResourcesCount": 1,
            "totalResourcesCount": 1,

            "message": "optional message if an error occurred",
            "errorCode": "INSUFFICIENT_PERMISSIONS|AUDIT_CHECK_DISABLED"
        }
    ]
}
```

承載 JSON 結構描述

```
{
    "$schema": "http://json-schema.org/draft-07/schema#",
    "$id": "arn:aws:iot:::schema:auditnotification/1.0",
    "type": "object",
    "properties": {
        "accountId": {
            "type": "string"
        },
        "taskId": {
            "type": "string"
        },
        "taskStatus": {
            "type": "string",
            "enum": [
                "FAILED",
                "CANCELED",
                "COMPLETED"
            ]
        },
        "taskType": {
```

```
            "type": "string",
            "enum": [
                "SCHEDULED_AUDIT_TASK",
                "ON_DEMAND_AUDIT_TASK"
            ]
        },
        "scheduledAuditName": {
            "type": "string"
        },
        "failedChecksCount": {
            "type": "integer"
        },
        "canceledChecksCount": {
            "type": "integer"
        },
        "nonCompliantChecksCount": {
            "type": "integer"
        },
        "compliantChecksCount": {
            "type": "integer"
        },
        "totalChecksCount": {
            "type": "integer"
        },
        "taskStartTime": {
            "type": "integer"
        },
        "auditDetails": {
            "type": "array",
            "items": [
                {
                    "type": "object",
                    "properties": {
                        "checkName": {
                            "type": "string",
                            "enum": [
                                "DEVICE_CERT_APPROACHING_EXPIRATION_CHECK",
                                "REVOKED_DEVICE_CERT_CHECK",
                                "CA_CERT_APPROACHING_EXPIRATION_CHECK",
                                "REVOKED_CA_CERT_CHECK",
                                "LOGGING_DISABLED_CHECK"
                            ]
                        },
                        "checkRunStatus": {
                            "type": "string",
                            "enum": [
                                "FAILED",
                                "CANCELED",
                                "COMPLETED_COMPLIANT",
                                "COMPLETED_NON_COMPLIANT"
                            ]
                        },
                        "nonCompliantResourcesCount": {
                            "type": "integer"
                        },
                        "totalResourcesCount": {
                            "type": "integer"
                        },
                        "message": {
                            "type": "string",
                        },
                        "errorCode": {
                            "type": "string",
                            "enum": [
                                "INSUFFICIENT_PERMISSIONS",
                                "AUDIT_CHECK_DISABLED"
```

```
                            ]
                        }
                    },
                    "required": [
                        "checkName",
                        "checkRunStatus",
                        "nonCompliantResourcesCount",
                        "totalResourcesCount"
                    ]
                }
            ]
        }
    },
    "required": [
        "accountId",
        "taskId",
        "taskStatus",
        "taskType",
        "failedChecksCount",
        "canceledChecksCount",
        "nonCompliantChecksCount",
        "compliantChecksCount",
        "totalChecksCount",
        "taskStartTime",
        "auditDetails"
    ]
}
```

您也可以在 AWS IoT 主控台中檢視通知，以及有關裝置、裝置統計資料 (例如，上一次連線、作用中連線的數量、資料傳輸速率) 和裝置的歷史提醒。

許可

本節包含有關如何設定建立、執行和管理 AWS IoT Device Defender 稽核所需 IAM 角色和政策的資訊。如需詳細資訊，請參閱 AWS Identity and Access Management User Guide (AWS 識別和存取使用者指南)。

提供 AWS IoT Device Defender 許可收集您的資料以執行稽核

當您呼叫 UpdateAccountAuditConfiguration (p. 535) 時，您必須指定 IAM 角色使用兩個政策：許可政策和信任政策。許可政策授予 AWS IoT Device Defender 在執行稽核時使用 AWS IoT API 來存取您帳戶資料的許可。信任政策授予 AWS IoT Device Defender 擔任所需角色的許可。

許可政策

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":[
                "iot:GetLoggingOptions",
                "iot:GetV2LoggingOptions",
                "iot:ListCACertificates",
                "iot:ListCertificates",
                "iot:DescribeCACertificate",
                "iot:DescribeCertificate",
                "iot:ListPolicies",
                "iot:GetPolicy",
                "iot:GetEffectivePolicies",
                "iot:ListRoleAliases",
                "iot:DescribeRoleAlias",
```

```
                "cognito-identity:GetIdentityPoolRoles",
                "iam:ListRolePolicies",
                "iam:ListAttachedRolePolicies",
                "iam:GetRole",
                "iam:GetPolicy",
                "iam:GetPolicyVersion",
                "iam:GetRolePolicy",
                "iam:GenerateServiceLastAccessedDetails",
                "iam:GetServiceLastAccessedDetails"
            ],
            "Resource":[
                "*"
            ]
        }
    ]
}
```

信任政策

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

賦予 AWS IoT Device Defender 將通知發佈到 SNS 主題的許可

如果在 UpdateAccountAuditConfiguration (p. 535) 中使用該 `auditNotificationTargetConfigurations` 參數，您必須指定 IAM 角色使用兩個政策、許可政策和信任政策。政策授予 AWS IoT Device Defender 許可的權限，以發佈通知到您的 SNS 主題。信任政策授予 AWS IoT Device Defender 擔任所需角色的許可。

許可政策

```
{
  "Version":"2012-10-17",
  "Statement":[
      {
        "Effect":"Allow",
        "Action":[
          "sns:Publish"
        ],
        "Resource":[
          "arn:aws:sns:region:account-id:your-topic-name"
        ]
    }
  ]
}
```

信任政策

```
{
```

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

給予 IAM 使用者或群組許可以執行 AWS IoT Device Defender 稽核命令

若要允許 IAM 使用者或群組管理、執行或檢視 AWS IoT Device Defender 稽核，您必須建立並指派角色和連接政策以授予許可執行適當的命令。每個政策的內容取決於您希望使用者或群組執行的命令。

- `UpdateAccountAuditConfiguration`

政策

您必須建立 IAM 角色與附加政策在相同的帳戶，才能執行此命令。不允許跨帳戶存取。政策應擁有 `iam:PassRole` 許可 (傳遞此角色的許可)。

在下列政策範本中，`audit-permissions-role-arn` 是您使用 `roleArn` 參數，在 `UpdateAccountAuditConfiguration` 請求中傳遞至 AWS IoT Device Defender 的角色 ARN。`audit-notifications-permissions-role-arn` 是您使用 `auditNotificationTargetConfigurations` 參數，在 `UpdateAccountAuditConfiguration` 請求中傳遞至 AWS IoT Device Defender 的角色 ARN。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:UpdateAccountAuditConfiguration"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::account-id:role/audit-permissions-role-arn",
        "arn:aws:iam::account-id:role/audit-notifications-permissions-role-arn"
      ]
    }
  ]
}
```

- `DescribeAccountAuditConfiguration`
- `DeleteAccountAuditConfiguration`

- StartOnDemandAuditTask
- CancelAuditTask
- DescribeAuditTask
- ListAuditTasks
- ListScheduledAudits
- ListAuditFindings

政策

在政策的 Resource 欄位中，所有這些命令都需要有 *。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:DescribeAccountAuditConfiguration",
        "iot:DeleteAccountAuditConfiguration",
        "iot:StartOnDemandAuditTask",
        "iot:CancelAuditTask",
        "iot:DescribeAuditTask",
        "iot:ListAuditTasks",
        "iot:ListScheduledAudits",
        "iot:ListAuditFindings"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

- CreateScheduledAudit
- UpdateScheduledAudit
- DeleteScheduledAudit
- DescribeScheduledAudit

政策

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:CreateScheduledAudit",
        "iot:UpdateScheduledAudit",
        "iot:DeleteScheduledAudit",
        "iot:DescribeScheduledAudit"
      ],
      "Resource": [
          "arn:aws:iot:region:account-id:scheduledaudit/scheduled-audit-name"
      ]
    }
  ]
}
```

AWS IoT Device Defender 排定稽核角色 ARN 的格式為：

```
arn:aws:iot:region:account-id:scheduledaudit/scheduled-audit-name
```

服務限制

資源	限制	描述
排程稽核	上限為 5。	在 `LimitExceeded` 例外狀況發生前最多可建立 5 項排程稽核。
同時進行中的隨需稽核	上限為 10。	在 `LimitExceeded` 例外狀況發生前最多可建立 10 項隨需稽核。

稽核命令

管理稽核設定

使用 `UpdateAccountAuditConfiguration` 為您的帳戶配置稽核設定。此命令可啟用您想要用於稽核的這些檢查、設定選用通知以及設定許可。

檢查這些設定 `DescribeAccountAuditConfiguration`。

使用 `DeleteAccountAuditConfiguration` 刪除稽核設定。這能恢復所有預設數值，並有效地停用稽核，因所有檢查都預設為停用。

UpdateAccountAuditConfiguration

為此帳戶設定或重新設定 Device Defender 稽核設定。設定值包括如何傳送稽核通知，以及啟用或停用哪些稽核檢查。

概要

```
aws iot  update-account-audit-configuration \
    [--role-arn <value>] \
    [--audit-notification-target-configurations <value>] \
    [--audit-check-configurations <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
    "string": {
      "enabled": "boolean"
```

```
    }
  }
}
```

cli-input-json 欄位

名稱	類型	敘述
roleArn	字串 長度上限：2048；下限：20	角色的 ARN 授予 AWS IoT 許可，可在執行稽核時存取裝置、政策、憑證和其他項目的相關資訊。
auditNotificationTargetConfiguration	對應	有關傳送稽核通知的目標的資訊。
targetArn	字串	傳送稽核通知的目標 ARN (SNS 主題)。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，以傳送通知至目標。
已啟用	布林值	若啟用通知至目標則為 True。
auditCheckConfigurations	對應	指定為此帳戶啟用和停用哪些稽核檢查。使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，其中包括目前已啟用的。 有些資料收集可能會在啟用特定檢查時立即開始。當停用檢查時，目前為止與檢查相關的任何收集資料會加以刪除。 若是由任何排定的稽核使用，則不能停用檢查。您必須先從排定的稽核刪除檢查，或刪除排定的稽核本身。 在對 `UpdateAccountAuditConfiguration` 的第一個呼叫，需使用此參數，且必須指定至少一個啟用的檢查。
已啟用	布林值	若為此帳戶啟用此稽核檢查，則為 True。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeAccountAuditConfiguration

針對此帳戶取得有關 Device Defender 稽核設定的資訊。設定值包括如何傳送稽核通知，以及啟用或停用哪些稽核檢查。

概要

```
aws iot  describe-account-audit-configuration  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
    "string": {
      "enabled": "boolean"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	敘述
roleArn	字串 長度上限：2048；下限：20	角色的 ARN 授予 AWS IoT 許可，可在執行稽核時存取裝置、政策、憑證和其他項目的相關資訊。 在對 `UpdateAccountAuditConfiguration` 的第一個呼叫，需要此參數。
auditNotificationTargetConfiguration	對應	有關對此帳戶傳送稽核通知的目標的資訊。

名稱	類型	敘述
targetArn	字串	傳送稽核通知的目標 ARN (SNS 主題)。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，以傳送通知至目標。
已啟用	布林值	若啟用通知至目標則為 True。
auditCheckConfigurations	對應	為此帳戶啟用和停用哪些稽核檢查。
已啟用	布林值	若為此帳戶啟用此稽核檢查，則為 True。

錯誤

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteAccountAuditConfiguration

針對此帳戶為 Device Defender 稽核還原預設值。您輸入的任何組態資料會被刪除，而所有稽核檢查會重設為停用。

概要

```
aws iot  delete-account-audit-configuration \
    [--delete-scheduled-audits | --no-delete-scheduled-audits]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "deleteScheduledAudits": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
deleteScheduledAudits	布林值	若為 true，則刪除所有排定的稽核。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

排定稽核時程

使用 `CreateScheduledAudit` 建立一或多個已排程的稽核。此命令允許您在稽核期間要執行的檢查，以及應執行這些稽核的頻率。

透過 `ListScheduledAudits` 和 `DescribeScheduledAudit` 追蹤排定的稽核。

透過 `UpdateScheduledAudit` 變更現有的已排程的稽核，或透過 `DeleteScheduledAudit` 將其刪除。

CreateScheduledAudit

建立已排程的稽核，其以指定的時間間隔執行。

概要

```
aws iot  create-scheduled-audit \
    --frequency <value> \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    --target-check-names <value> \
    [--tags <value>] \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "scheduledAuditName": "string"
}
```

cli-input-json 欄位

名稱	類型	敘述
frequency (頻率)	字串	已排程的稽核執行的頻率。可以是「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的實際開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	字串 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程稽核的日期。可以是 1 到 31 或「最後一天」。如果 `frequency` 參數設定為「每月」，則此欄位為必填。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	字串	當週執行已排程稽核的日期。可以是「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。如果 `frequency` 參數設定為「每週」或「每兩週」，則此欄位為必填。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
targetCheckNames	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括已啟用的，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。)
tags	列出 成員：Tag Java 類別：java.util.List	用於管理排程稽核的中繼資料。
金鑰	字串	標籤的金鑰。
數值	字串	標籤的值。
scheduledAuditName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	您要指定給已排程的稽核的名稱。(最多 128 個字元)

輸出

```
{
```

```
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
scheduledAuditArn	字串	已排程的稽核的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

ListScheduledAudits

列出所有已排程的稽核。

概要

```
aws iot  list-scheduled-audits \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	說明
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。

輸出

```
{
  "scheduledAudits": [
    {
      "scheduledAuditName": "string",
      "scheduledAuditArn": "string",
      "frequency": "string",
      "dayOfMonth": "string",
      "dayOfWeek": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
scheduledAudits	清單 成員：ScheduledAuditMetadata Java 類別：java.util.List	已排程的稽核的清單。
scheduledAuditName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱。
scheduledAuditArn	字串	已排程的稽核的 ARN。
frequency (頻率)	字串	已排程的稽核執行的頻率。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	字串 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程之稽核的日期 (如果 `frequency` 是「每月」)。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	字串	當週執行已排程之稽核的日期 (如果 `frequency` 是「每週」或「每兩週」)。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
nextToken	string	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeScheduledAudit

取得有關已排程的稽核的資訊。

概要

```
aws iot  describe-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
scheduledAuditName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	您需要取得其資訊的已排程稽核的名稱。

輸出

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string",
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
frequency (頻率)	字串	已排程的稽核執行的頻率。「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的實際開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	字串	當月執行已排程稽核的日期。可以是 1 到 31 或「最後一天」。如

名稱	類型	敘述
	模式：^([1-9]|[12][0-9]|3[01])$\| ^LAST$	果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	字串	當週執行已排程稽核的日期。「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
targetCheckNames	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括已啟用的檢查，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。)
scheduledAuditName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱。
scheduledAuditArn	字串	已排程的稽核的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

UpdateScheduledAudit

更新已排程的稽核，包含執行哪些檢查，以及執行稽核的頻率。

概要

```
aws iot  update-scheduled-audit \
    [--frequency <value>] \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    [--target-check-names <value>] \
```

```
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
frequency (頻率)	字串	已排程的稽核執行的頻率。可以是「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的實際開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	字串 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程稽核的日期。可以是 1 到 31 或「最後一天」。如果 `frequency` 參數設定為「每月」，則此欄位為必填。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	字串	當週執行已排程稽核的日期。可以是「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。如果 `frequency` 參數設定為「每週」或「每兩週」，則此欄位為必填。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
targetCheckNames	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括已啟用的檢查，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。)
scheduledAuditName	字串 長度上限：128；下限：1	已排程的稽核的名稱。(最多 128 個字元)

名稱	類型	敘述
	模式：[a-zA-Z0-9_-]+	

輸出

```
{
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
scheduledAuditArn	字串	已排程的稽核的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteScheduledAudit

刪除已排程的稽核。

概要

```
aws iot  delete-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
scheduledAuditName	字串 長度上限：128；下限：1	要刪除的已排程稽核的名稱。

名稱	類型	敘述
	模式：[a-zA-Z0-9_-]+	

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

執行隨需稽核

使用 `StartOnDemandAuditTask` 指定您要執行的檢查並且立即開始執行稽核。

StartOnDemandAuditTask

開始執行 Device Defender 稽核。

概要

```
aws iot  start-on-demand-audit-task \
    --target-check-names <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "targetCheckNames": [
    "string"
  ]
}
```

`cli-input-json` 欄位

名稱	類型	敘述
targetCheckNames	列出 成員：AuditCheckName	稽核期間執行哪些檢查。必須為您的帳戶啟用您指定的檢查，或發生例外。使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括

名稱	類型	敘述
		已啟用的檢查，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。

輸出

```
{
  "taskId": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您開始執行之隨需稽核的 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

管理稽核執行個體

使用 `DescribeAuditTask` 以取得有關特定稽核執行個體的資訊。是否已在執行、包含哪些檢查失敗及通過檢查的結果、系統無法完成哪些檢查，以及稽核是否仍在進行中，哪些仍運作。

使用 `ListAuditTasks` 以尋找在指定的時間間隔內執行的稽核。

使用 `CancelAuditTask` 停止進行中的稽核。

DescribeAuditTask

取得有關 Device Defender 稽核的資訊。

概要

```
aws iot  describe-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您需要取得其資訊的稽核的 ID。

輸出

```
{
  "taskStatus": "string",
  "taskType": "string",
  "taskStartTime": "timestamp",
  "taskStatistics": {
    "totalChecks": "integer",
    "inProgressChecks": "integer",
    "waitingForDataCollectionChecks": "integer",
    "compliantChecks": "integer",
    "nonCompliantChecks": "integer",
    "failedChecks": "integer",
    "canceledChecks": "integer"
  },
  "scheduledAuditName": "string",
  "auditDetails": {
    "string": {
      "checkRunStatus": "string",
      "checkCompliant": "boolean",
      "totalResourcesCount": "long",
      "nonCompliantResourcesCount": "long",
      "errorCode": "string",
      "message": "string"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	敘述
taskStatus	字串	稽核的狀態：IN_PROGRESS、COMPLETED、FAILED 或 CANCELED 的其中之一。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
taskType	字串	稽核的類型：ON_DEMAND_AUDIT_TASK 或 SCHEDULED_AUDIT_TASK。

名稱	類型	敘述
		列舉：ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStartTime	timestamp	稽核開始的時間。
taskStatistics	TaskStatistics	有關稽核的統計資訊。
totalChecks	整數	此稽核中的檢查數量。
inProgressChecks	整數	進行中的檢查數量。
waitingForDataCollectionChecks	整數	正在等待資料收集的檢查數量。
compliantChecks	整數	找到合規資源的檢查數量。
nonCompliantChecks	integer	找到未合規之資源的檢查數量。
failedChecks	integer	檢查數量。
canceledChecks	integer	因稽核遭取消無法執行的檢查數量。
scheduledAuditName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱 (僅當稽核為已排程的稽核時)。
auditDetails	對應	有關在這個稽核期間所執行的每個檢查的詳細資訊。
checkRunStatus	字串	此檢查的完成狀態，IN_PROGRESS、WAITING_FOR_DATA_C 或 FAILED 的其中之一。 列舉：IN_PROGRESS | WAITING_FOR_DATA_COLLECTION | CANCELED | COMPLETED_COMPLIANT | COMPLETED_NON_COMPLIANT | FAILED
checkCompliant	布林值	如果檢查完成且找到所有合規資源，則為 True。
totalResourcesCount	長整數	執行檢查的資源數量。
nonCompliantResourcesCount	long	檢查發現未合規的資源數量。
errorCode	字串	當在此稽核期間執行此檢查時，遇到的任何錯誤代碼。INSUFFICIENT_PERMISSIONS 或 AUDIT_CHECK_DISABLED 的其中之一。
message	字串 長度上限：2048	當在此稽核期間執行此檢查時，遇到的任何錯誤相關訊息。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditTasks

列出特定期間執行的 Device Defender 稽核。

概要

```
aws iot  list-audit-tasks \
    --start-time <value> \
    --end-time <value> \
    [--task-type <value>] \
    [--task-status <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "startTime": "timestamp",
  "endTime": "timestamp",
  "taskType": "string",
  "taskStatus": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
startTime	timestamp	時期的開始。稽核資訊會保留一段有限時期 (180 天)。要求開始時間，然後才會在 `InvalidRequestException` 保留結果。
endTime	timestamp	時期的結束。
taskType	字串	限制輸出到指定的稽核類型的篩選條件：可以是 ON_DEMAND_AUDIT_TASK 或 SCHEDULED__AUDIT_TASK 的其中一個。

名稱	類型	敘述
		列舉：ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStatus	字串	以指定完成狀態來限制輸出到稽核的篩選條件：可以是 IN_PROGRESS、COMPLETED、FAILED 或 CANCELED 的其中一個。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。

輸出

```
{
  "tasks": [
    {
      "taskId": "string",
      "taskStatus": "string",
      "taskType": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
任務	清單 成員：AuditTaskMetadata Java 類別：java.util.List	在指定期間內執行的稽核。
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	此稽核的 ID。
taskStatus	字串	此稽核的狀態：IN_PROGRESS、COMPLETED、FAILED 或 CANCELED 的其中之一。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
taskType	字串	此稽核的類型：ON_DEMAND_AUDIT_TASK 或

名稱	類型	敘述
		SCHEDULED_AUDIT_TASK 的其中之一。 列舉： ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CancelAuditTask

取消正在進行的稽核。稽核可以是排定或隨需。如果稽核不在進行中，則會發生 `InvalidRequestException`。

概要

```
aws iot  cancel-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	敘述
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您要取消的稽核的 ID。您只能取消 IN_PROGRESS 的稽核。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

檢查稽核結果

使用 `ListAuditFindings` 以查看稽核結果。您可以依照檢查類型、特定資源或稽核時間來篩選結果。您可以使用此資訊來減輕任何找到的問題。

您可以定義緩解行動和從您的稽核套用到結果。如需更多詳細資訊，請參閱 緩解動作 (p. 559)。

ListAuditFindings

列出 Device Defender 稽核或在指定期間內執行之稽核時的發現 (結果)。(發現項目會保留 180 天)。

概要

```
aws iot  list-audit-findings \
    [--task-id <value>] \
    [--check-name <value>] \
    [--resource-identifier <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--start-time <value>] \
    [--end-time <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string",
  "checkName": "string",
  "resourceIdentifier": {
    "deviceCertificateId": "string",
    "caCertificateId": "string",
    "cognitoIdentityPoolId": "string",
    "clientId": "string",
    "policyVersionIdentifier": {
      "policyName": "string",
      "policyVersionId": "string"
    },

    "roleAliasArn": "string",
    "account": "string"
  },
  "maxResults": "integer",
  "nextToken": "string",
  "startTime": "timestamp",
```

```
  "endTime": "timestamp"
}
```

cli-input-json 欄位

名稱	類型	敘述
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	可限制具指定 ID 的稽核結果的篩選條件。您必須指定 taskId 或 startTime 和 endTime，但不能同時指定兩者。
checkName	字串	可限制指定的稽核檢查的發現結果的篩選條件。
resourceIdentifier	ResourceIdentifier	可識別不合規資源的資訊。
deviceCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	字串	Amazon Cognito 身分集區的 ID。
clientId	字串	用戶端 ID。
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
policyName	字串 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	字串 模式：[0-9]+	與資源相關聯的政策版本 ID。
roleAliasArn	string	具有過於寬鬆動作之角色別名的 ARN。 長度上限：2048；下限：1
帳戶	字串 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。
nextToken	字串	下一組結果的字符。

名稱	類型	敘述
startTime	timestamp	一個篩選功能，其能限制指定的時間過後所找到的結果。您必須指定 startTime 和 endTime 或 taskId，但不能同時指定兩者。
endTime	timestamp	一個篩選功能，其能限制指定的時間之前所找到的結果。您必須指定 startTime 和 endTime 或 taskId，但不能同時指定兩者。

輸出

```
{
  "findings": [
    {
      "taskId": "string",
      "checkName": "string",
      "taskStartTime": "timestamp",
      "findingTime": "timestamp",
      "severity": "string",
      "nonCompliantResource": {
        "resourceType": "string",
        "resourceIdentifier": {
          "deviceCertificateId": "string",
          "caCertificateId": "string",
          "cognitoIdentityPoolId": "string",
          "clientId": "string",
          "policyVersionIdentifier": {
            "policyName": "string",
            "policyVersionId": "string"
          },
          "account": "string"
        },
        "additionalInfo": {
          "string": "string"
        }
      },
      "relatedResources": [
        {
          "resourceType": "string",
          "resourceIdentifier": {
            "deviceCertificateId": "string",
            "caCertificateId": "string",
            "cognitoIdentityPoolId": "string",
            "clientId": "string",

            "iamRoleArn": "string",

            "policyVersionIdentifier": {
              "policyName": "string",
              "policyVersionId": "string"
            },
            "account": "string"
          },

          "roleAliasArn": "string",

          "additionalInfo": {
            "string": "string"
          }
```

```
        }
      ],
      "reasonForNonCompliance": "string",
      "reasonForNonComplianceCode": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	敘述
問題清單	清單 成員：AuditFinding	稽核的發現 (結果)。
taskId	字串 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	產生此結果 (發現) 的稽核 ID。
checkName	字串	產生此結果的稽核檢查。
taskStartTime	timestamp	稽核開始的時間。
findingTime	timestamp	發現結果 (尋找) 的時間。
嚴重性	字串	結果 (發現) 的嚴重性。 列舉：CRITICAL | HIGH | MEDIUM | LOW
nonCompliantResource	NonCompliantResource	發現不符合稽核檢查的資源。
resourceType	字串	未合規的資源類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	可識別不合規資源的資訊。
deviceCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	字串	Amazon Cognito 身分集區的 ID。
clientId	字串	用戶端 ID。

名稱	類型	敘述
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
policyName	字串 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	字串 模式：[0-9]+	與資源相關聯的政策版本 ID。
帳戶	字串 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
additionalInfo	對應	有關不合規資源的其他資訊。
relatedResources	列出 成員：RelatedResource	相關資源的清單。
resourceType	字串	資源的類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	可識別資源的資訊。
deviceCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	字串 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	字串	Amazon Cognito 身分集區的 ID。
clientId	字串	用戶端 ID。
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
iamRoleArn	字串 長度上限：2048；下限：20	具有過於寬鬆動作之 IAM 角色的 ARN。

名稱	類型	敘述
policyName	字串 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	字串 模式：[0-9]+	與資源相關聯的政策版本 ID。
roleAliasArn	string 長度上限：2048；下限：1	具有過於寬鬆動作之角色別名的 ARN。
帳戶	字串 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
additionalInfo	對應	關於資源的其他資訊。
reasonForNonCompliance	字串	資源未合規的原因。
reasonForNonComplianceCode	字串	可指出資源不合規原因的代碼。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

緩解動作

您可以使用 AWS IoT Device Defender 來採取動作，以緩解稽核期間發現的問題。AWS IoT Device Defender 針對不同的稽核檢查提供了預先定義的動作。您可以為您的 AWS 帳戶設定這些動作，然後套用到一組問題清單。這些問題可能是：

- 在稽核中的所有問題。您可以在 AWS IoT 主控台及使用 AWS CLI 來使用此選項。
- 個別問題的清單。僅有使用 AWS CLI 才可使用此選項。
- 在稽核中篩選過的一組問題。

下表列出稽核檢查類型，以及針對每個項目支援的緩解動作：

緩解動作映射的稽核檢查

稽核檢查	支援緩解動作
REVOKED_CA_CERT_CHECK	UPDATE_CA_CERTIFICATE PUBLISH_FINDING_TO_SNS、
DEVICE_CERTIFICATE_SHARED_CHECK	UPDATE_DEVICE_CERTIFICATE、ADD_THINGS_TO_THING_G PUBLISH_FINDING_TO_SNS、
UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
IOT_POLICY_OVERLY_PERMISSIVE_CHECK	REPLACE_DEFAULT_POLICY_VERSION PUBLISH_FINDING_TO_SNS、
CA_CERT_APPROACHING_EXPIRATION_CHECK	UPDATE_CA_CERTIFICATE PUBLISH_FINDING_TO_SNS、
CONFLICTING_CLIENT_IDS_CHECK	PUBLISH_FINDING_TO_SNS
DEVICE_CERT_APPROACHING_EXPIRATION_CHECK	UPDATE_DEVICE_CERTIFICATE、ADD_THINGS_TO_THING_G PUBLISH_FINDING_TO_SNS、
REVOKED_DEVICE_CERT_CHECK	UPDATE_DEVICE_CERTIFICATE、ADD_THINGS_TO_THING_G PUBLISH_FINDING_TO_SNS、
LOGGING_DISABLED_CHECK	ENABLE_IOT_LOGGING PUBLISH_FINDING_TO_SNS、
DEVICE_CERTFICATE_KEY_QUALITY_CHECK	UPDATE_DEVICE_CERTIFICATE、ADD_THINGS_TO_THING_G PUBLISH_FINDING_TO_SNS、
CA_CERTIFICATE_KEY_QUALITY_CHECK	UPDATE_CA_CERTIFICATE PUBLISH_FINDING_TO_SNS、
IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK	PUBLISH_FINDING_TO_SNS

所有的稽核檢查都支援發佈稽核問題到 Amazon SNS，以便您採取自訂動作來回應通知。每個稽核檢查類型都可支援其他緩解動作：

REVOKED_CA_CERT_CHECK

- 變更憑證狀態，以將其在 AWS IoT 中標示為非作用中。

DEVICE_CERTIFICATE_SHARED_CHECK

- 變更裝置憑證狀態，以將其在 AWS IoT 中標示為非作用中。
- 將使用該憑證的裝置新增到物件群組。

UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- 沒有其他支援的動作。

AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- 沒有其他支援的動作。

IOT_POLICY_OVERLY_PERMISSIVE_CHECK

- 新增空白 AWS IoT 政策版本到限制權限。

CA_CERT_APPROACHING_EXPIRATION_CHECK

- 變更憑證狀態，以將其在 AWS IoT 中標示為非作用中。

CONFLICTING_CLIENT_IDS_CHECK

- 沒有其他支援的動作。

DEVICE_CERT_APPROACHING_EXPIRATION_CHECK

- 變更裝置憑證狀態，以將其在 AWS IoT 中標示為非作用中。
- 將使用該憑證的裝置新增到物件群組。

DEVICE_CERTIFICATE_KEY_QUALITY_CHECK

- 變更裝置憑證狀態，以將其在 AWS IoT 中標示為非作用中。
- 將使用該憑證的裝置新增到物件群組。

CA_CERTIFICATE_KEY_QUALITY_CHECK

- 變更憑證狀態，以將其在 AWS IoT 中標示為非作用中。

REVOKED_DEVICE_CERT_CHECK

- 變更裝置憑證狀態，以將其在 AWS IoT 中標示為非作用中。
- 將使用該憑證的裝置新增到物件群組。

LOGGING_DISABLED_CHECK

- 啟用日誌記錄。

AWS IoT Device Defender 支援以下類型的緩解動作：

動作類型	備註
ADD_THINGS_TO_THING_GROUP	您將指定要新增裝置的群組。若成員資格超出物件可以隸屬最大群組數目，您也可以指定是否覆寫在一或多個動態群組中的成員資格。
ENABLE_IOT_LOGGING	您將透過記錄權限指定記錄層級和角色。您不能指定 `DISABLED` 的記錄層級。
PUBLISH_FINDING_TO_SNS	您將指定問題應該發佈的主題。
REPLACE_DEFAULT_POLICY_VERSION	您將指定範本名稱。以預設或空白政策取代政策版本。目前只支援 `BLANK_POLICY` 的值。
UPDATE_CA_CERTIFICATE	您將指定 CA 憑證的新狀態。目前只支援 `DEACTIVATE` 的值。
UPDATE_DEVICE_CERTIFICATE	您將指定裝置憑證的新狀態。目前只支援 `DEACTIVATE` 的值。

透過在稽核期間找到問題時設定標準動作，您可以持續針對這些問題進行回應。使用這些定義的緩解動作，也有助於更快速解決問題，並降低人為錯誤的機會。

> **Important**
>
> 套用變更憑證、新增物件到新群組的緩解動作，或置換可能影響您裝置和應用程式的政策。例如，裝置可能無法連接。請在套用緩解動作前先考量其影響。在裝置和應用程式可以正常運作之前，您可能需要採取其他動作來更正問題。例如，您可能需要提供更新的裝置憑證。緩解動作可以協助您快速限制風險，但您仍必須採取更正動作來解決基本問題。

有些動作 (例如重新啟用裝置憑證) 只能手動執行。AWS IoT Device Defender 並不提供已套用的自動復原緩解動作機制。

如何定義並管理緩解動作

您可以使用 AWS IoT 主控台或 AWS CLI 來定義和管理您 AWS 帳戶的緩解動作。

建立緩解動作

您定義的每個緩解動作，都是預先定義的動作類型和您帳戶的特定參數的結合。

使用主 AWS IoT 控台來建立緩解動作

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Mitigation Actions (緩解動作)。
3. 在 Mitigation Actions (緩解動作) 頁面上選擇 Create (建立)。

4. 在 Create a Mitigation Action (建立緩解動作) 頁面的 Action name (動作名稱) 中，輸入唯一的緩解動作名稱。
5. 在 Action type (動作類型) 中，指定您想要定義的動作類型。
6. 每個動作類型都會要求一組不同的參數。輸入動作的參數。例如，如果您選擇 Add things to thing group (新增物件到物件群組) 動作類型，選擇目標群組並選擇或清除 Override dynamic groups (覆寫動態群組)。
7. 在 Action execution role (動作執行角色) 中，選擇已套用動作權限下的角色。
8. 選擇 Save (儲存) 以儲存緩解動作到您的 AWS 帳戶。

使用 AWS CLI 來建立緩解動作

- 使用 CreateMitigationAction (p. 574) 命令來建立您的緩解動作。在您套用動作到稽核問題時，將使用您給予動作的唯一名稱。選擇有意義的名稱。

使用主 AWS IoT 控台以檢視和修改緩解動作

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Mitigation Actions (緩解動作)。

   Mitigation Actions (緩解動作) 頁面會顯示為您 AWS 帳戶中定義的所有緩解動作清單。

3. 選擇您想變更的緩解動作動作名稱連結。
4. 對緩解動作進行變更。由於緩解動作的名稱是用於識別，因此您無法更改名稱。

5. 選擇 Save (儲存) 以將緩解動作變更儲存到您的 AWS 帳戶。

使用 AWS CLI 列出緩解動作

- 使用 ListMitigationActions (p. 581) 命令列出緩解動作。如果您想要變更或刪除緩解動作，請記下其名稱。

使用 AWS CLI 更新緩解動作

- 使用 UpdateMitigationAction (p. 578) 命令來變更緩解動作。

使用 AWS IoT 主控台刪除緩解動作

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Mitigation Actions (緩解動作)。

   Mitigation Actions (緩解動作) 頁面會顯示為您 AWS 帳戶中定義的所有緩解動作。
3. 選擇您希望刪除的緩解動作省略符號 (...)，然後選擇 Delete (刪除)。

使用 AWS CLI 刪除緩解動作

- 使用 UpdateMitigationAction (p. 578) 命令來變更緩解動作。

使用 AWS IoT 主控台檢視緩解動作詳細資訊

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Mitigation Actions (緩解動作)。

   Mitigation Actions (緩解動作) 頁面會顯示為您 AWS 帳戶中定義的所有緩解動作。
3. 選擇您想變更的緩解動作動作名稱連結。
4. 在 Are you sure you want to delete the mitigation action (您確定要刪除緩解動作嗎) 視窗中，選擇 Confirm (確認)。

使用 AWS CLI 以檢視緩解動作詳細資訊

- 使用 DescribeMitigationAction (p. 582) 命令來檢視您的緩解動作詳細資訊。

套用緩解動作

在您定義一組緩解動作後，您可以將這些動作套用到稽核的問題上。在您套用動作後，您便開始了稽核緩解動作任務。這個任務可能需要一些時間才能完成，取決於您的問題組和套用的動作。例如，如果您有一個憑證已過期的大型裝置集區，便可能需要花費一些時間停用所有憑證，或將這些裝置移動到隔離群組。其他動作 (例如啟用記錄功能) 則可快速完成。

您可以檢視動作執行清單和取消尚未完成的執行。已取消動作執行的已執行動作將無法復原。如果您正在套用多個動作到一組問題上，而其中一個動作失敗了，則後續動作將略過該問題 (但仍會套用到其他問題上)。問題的任務狀態為 FAILED (失敗)。在套用至問題時，如果有一或多個動作失敗，則 `taskStatus` 設為失敗。動作皆依指定的順序套用。

每個動作執行會套用一組動作到目標。該目標可以是一個問題清單，也可以是稽核的所有問題。

下圖說明如何定義稽核緩解任務，該任務會從一個稽核取得所有問題，並將一組動作套用到這些問題。單一執行會套用一個動作到一個問題。稽核緩解動作任務會輸出執行摘要。

下圖說明如何定義稽核緩解任務，該任務會從一或多個稽核取得一份個別問題清單，並將一組動作套用到這些問題上。單一執行會套用一個動作到一個問題。稽核緩解動作任務會輸出執行摘要。

您可以使用 AWS IoT 主控台或 AWS CLI 以套用緩解動作。

透過啟動動作執行以使用 AWS IoT 主控台套用緩解動作

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Audit (稽核)，然後選擇 結果。

3. 選擇您要套用動作的稽核名稱。

4. 選擇 Start Mitigation Actions (開始緩解動作)。如果您的所有檢查都合規，則此按鈕無法使用。
5. 在 Are you sure that you want to start mitigation action task (您確定要開始緩解動作任務嗎) 中，任務名稱預設為稽核 ID，但您可以變更為其他有意義的名稱。
6. 對於稽核中每種擁有一或多個不合規的檢查類型而言，您可以選擇套用一或多個動作。只有對檢查類型有效的動作才會顯示。

   Note

   如果您還沒有為您的 AWS 帳戶設定動作，則動作清單為空的。您可以選擇 click here (按一下這裡) 連結來建立一或多個緩解動作。
7. 當您指定想要套用的所有動作後，請選擇 Confirm (確認)。

透過啟動稽核緩解動作執行以使用 AWS CLI 來套用緩解動作

1. 如果您想要套用動作到稽核的所有問題，請使用 ListAuditTasks (p. 551) 命令來尋找任務 ID。
2. 如果您僅想將動作套用到所選問題上，請使用 ListAuditFindings (p. 554) 命令以取得問題 ID。
3. 使用 ListMitigationActions (p. 581) 命令並記下您要套用的緩解動作名稱。

4. 使用 StartAuditMitigationActionsTask (p. 586) 命令來套用動作到目標。請記下任務 ID。您可以使用該 ID 來檢查動作執行狀態、檢閱詳細資訊，或將其取消。

使用 AWS IoT 主控台以檢視您的動作執行

1. 開啟 AWS IoT 主控台。
2. 在左側導覽窗格中，選擇 Defend (防禦)，然後選擇 Action Executions (動作執行)。

動作任務清單會顯示每個啟動和目前的狀態。

3. 選擇 Name (名稱) 連結以檢視任務詳細資訊。詳細資訊包括由任務套用的所有動作、他們的目標，以及任務的狀態。

您可以使用 Show executions for (顯示執行) 篩選條件，以專注於動作類型或在動作狀態。

4. 若要查看任務的詳細資訊，請在 Executions (執行) 中選擇 Show (顯示)。

使用 AWS CLI 來列出您已開始的任務

1. 使用 ListAuditMitigationActionsTasks (p. 592) 來檢視您的稽核緩解動作任務。您可以提供篩選條件來縮小結果。如果您想要檢視任務的詳細資訊，請記下任務 ID。
2. 使用 ListAuditMitigationActionsExecutions (p. 589) 以檢視特定稽核緩解動作任務的執行詳細資訊。
3. 使用 DescribeAuditMitigationActionsTask (p. 595) 以檢視任務的詳細資訊，例如該任務啟動時指定的參數。

使用 AWS CLI 來取消執行中的稽核緩解動作任務

1. 使用 ListAuditMitigationActionsTasks (p. 592) 命令來尋找您想取消執行任務的任務 ID。您可以提供篩選條件來縮小結果。
2. 使用 CancelAuditMitigationActionsTask (p. 589) 命令，使用任務 ID 來取消您的稽核緩解動作任務。您無法取消已完成的任務。在您取消任務時，剩餘的動作便不會套用，但已套用的緩解動作將不會復原。

許可

對於每個您定義的緩解動作，您必須提供用於套用該動作的角色。

緩解動作的權限

動作類型	許可政策範本	
UPDATE_DEVICE_CERTIFICATE	`{ "Version":"2012-10-17", "Statement":[ { "Effect":"Allow",`	

動作類型	許可政策範本	
	"Action":["iot:UpdateCertificate"], "Resource":["*"] }] }	

UPDATE_CA_CERTIFICATE	{
	"Version":"2012-10-17",
	"Statement":[
	{
	
	"Effect":"Allow",
	"Action":[
	
	"iot:UpdateCACertificate"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

ADD_THINGS_TO_THING_GROUP	{
	"Version":"2012-10-17",
	"Statement":[
	{
	
	"Effect":"Allow",
	"Action":[
	
	"iot:ListPrincipalThings",
	
	"iot:AddThingToThingGroup"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

動作類型	許可政策範本	
REPLACE_DEFAULT_POLICY_VERSION	`{ "Version":"2012-10-17", "Statement":[ { "Effect":"Allow", "Action":[ "iot:CreatePolicyVersion" ], "Resource":[ "*" ] } ] }`	
ENABLE_IOT_LOGGING	`{ "Version":"2012-10-17", "Statement":[ { "Effect":"Allow", "Action":[ "iot:SetV2LoggingOptions" ], "Resource":[ "*" ] }, { "Effect":"Allow", "Action":[ "iam:PassRole" ], "Resource":[ "<IAM role ARN used for setting up logging>" ] } ] }`	

動作類型	許可政策範本	
PUBLISH_FINDING_TO_SNS	{ "Version":"2012-10-17", "Statement":[{ "Effect":"Allow", "Action":["sns:Publish"], "Resource":["<The SNS topic to which the finding is published>"] }] }	

對於所有緩解動作類型，請使用下列信任政策範本：


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


Service Limits

以下服務限制適用於緩解動作和稽核緩解動作任務：

資源	限制	
在 AWS 帳戶中的緩解動作最大數量	100 個動作	
同時執行中的稽核緩解動作任務最大數量	10 個任務	
稽核緩解動作任務的保留期間	90 天	

緩解行動命令

您可以使用緩解行動命令為您的 AWS 帳戶定義一組動作，之後可以套用到一組或多組稽核結果。命令有兩種類別：

- 這些類別用於定義和管理行動。
- 這些類別用於啟動和管理應用程式的行動來稽核結果。

緩解行動命令

定義和管理行動	啟動和管理執行
CreateMitigationAction (p. 574)	CancelAuditMitigationActionsTask (p. 589)
DeleteMitigationAction (p. 586)	DescribeAuditMitigationActionsTask (p. 595)
DescribeMitigationAction (p. 582)	ListAuditMitigationActionsTasks (p. 592)
ListMitigationActions (p. 581)	StartAuditMitigationActionsTask (p. 586)
UpdateMitigationAction (p. 578)	ListAuditMitigationActionsExecutions (p. 589)

CreateMitigationAction

定義可使用 StartAuditMitigationActionsTask (p. 586) 套用到稽核結果的行動。每個緩解行動只能套用一個類型的變更。定義不套用的行動。

概要

```
aws iot  create-mitigation-action \
    --action-name <value] \
    --role-arn <value> \
    [--tags <value>] \
    --action-params <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
   "actionParams": {
      "addThingsToThingGroupParams": {
         "overrideDynamicGroups": boolean,
         "thingGroupNames": [ "string" ]
      },
      "enableIoTLoggingParams": {
         "logLevel": "string",
         "roleArnForLogging": "string"
      },
      "publishFindingToSnsParams": {
         "topicArn": "string"
      },
      "replaceDefaultPolicyVersionParams": {
         "templateName": "string"
      },
      "updateCACertificateParams": {
         "action": "string"
```

```
      },
      "updateDeviceCertificateParams": {
         "action": "string"
      }
   },
   "roleArn": "string",
   "tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

cli-input-json 欄位

名稱	類型	描述
roleArn	字串 長度上限：2048；下限：20	角色授予 AWS IoT 許可的 ARN，可在套用至此行動時存取裝置、政策、憑證和其他項目的相關資訊。
tags	標籤物件的陣列	用於管理緩解行動的中繼資料。
actionParams	對應	定義套用行動的類型和該緩解行動的參數。每個緩解行動只可以包含一個類型的參數。

您必須提供您定義的行動類型的參數。您只可以提供一個行動類型及其參數。以下是支援的行動類型：

- ADD_THINGS_TO_THING_GROUP
- ENABLE_IOT_LOGGING
- PUBLISH_FINDING_TO_SNS
- REPLACE_DEFAULT_POLICY_VERSION
- UPDATE_CA_CERTIFICATE
- UPDATE_DEVICE_CERTIFICATE

AddThingsToThingGroup 參數

名稱	類型	描述
overrideDynamicGroups	布林值	選用。指定緩解行動如果此行動可以移動觸發一個或多個動態物件群組緩解行動的物件。此設定只在物件已經達到群組最大值時使用。
thingGroupNames	字串陣列	必要。觸發緩解行動新增物件的群組清單。 您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。 您必須提供至少一個群組名稱。

名稱	類型	描述
		長度限制：最小長度為 1。長度上限為 128。 模式：[a-zA-Z0-9:_-]+

EnableIoTLogging 參數

名稱	類型	描述
`logLevel`	字串	必要。指定記錄哪些類型的資訊。 有效的值，從詳細資訊到最小資訊，為 `DEBUG`、`INFO`、`ERROR` 和 `WARN`。您無法指定記錄層級 `DISABLED`。
`roleArnForLogging`	字串	必要。用於記錄 IAM 角色的 ARN。 長度下限為 20。長度上限為 2048。

PublishingFindingToSns 參數

名稱	類型	描述
`topicArn`	字串	必要。您要發佈結果主題的 ARN。 長度下限為 20。長度上限為 2048。

ReplaceDefaultPolicyVersion 參數

名稱	類型	描述
`templateName`	字串	必要。要套用範本的名稱。 唯一支援的值為 `BLANK_POLICY`。

UpdateCACertificate 參數

名稱	類型	描述
`action`	字串	必要。您要套用到 CA 憑證的行動。 唯一支援的值為 `DEACTIVATE`。

UpdateDeviceCertificate 參數

名稱	類型	描述
action	字串	必要。您要套用到裝置憑證的行動。 唯一支援的值為 DEACTIVATE。

CLI 輸出欄位：

名稱	類型	描述
actionArn	字串	新的緩解行動的 ARN。
actionId	字串	新的緩解行動的唯一識別符。

錯誤

InvalidRequestException

請求的內容無效。

LimitExceededException

已超出限制。如需有關緩解行動限制的詳細資訊，請參閱 服務限制。

RequestAlreadyExistsException

此緩解行動名稱已存在。此錯誤只會發生於已存在相同的緩解行動名稱但是不同的參數。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

範例

此範例定義緩解行動，當套用時停用裝置的憑證。

```
$ aws iot create-mitigation-action --action-name "UpdateCACertName" --role-
arn arn:aws:iam::123456789012:role/MitigationActionsValidRole --action-params
 "updateDeviceCertificateParams={action=DEACTIVATE}"
```

回應看起來會與以下內容相似。

```
{
    "actionArn": "arn:aws:iot:us-east-1:123456789012:mitigationaction/UpdateCACertName",
    "actionId": "6a22b98e-0e27-4396-9b25-637d04959429"
}
```

UpdateMitigationAction

更新的定義可使用 StartAuditMitigationActionsTask (p. 586) 套用到稽核結果的行動。每個緩解行動只能套用一個類型的變更。

概要

```
aws iot update-mitigation-action \
    --action-name <value> \
    --role-arn <value> \
    --action-params <value]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
   "actionParams": {
      "addThingsToThingGroupParams": {
         "overrideDynamicGroups": boolean,
         "thingGroupNames": [ "string" ]
      },
      "enableIoTLoggingParams": {
         "logLevel": "string",
         "roleArnForLogging": "string"
      },
      "publishFindingToSnsParams": {
         "topicArn": "string"
      },
      "replaceDefaultPolicyVersionParams": {
         "templateName": "string"
      },
      "updateCACertificateParams": {
         "action": "string"
      },
      "updateDeviceCertificateParams": {
         "action": "string"
      }
   },
   "roleArn": "string"
   ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`roleArn`	字串 長度上限：2048；下限：20	角色授予 AWS IoT 許可的 ARN，可在套用至此行動時存取裝置、政策、憑證和其他項目的相關資訊。
`actionParams`	對應	定義套用行動的類型和該緩解行動的參數。您只可以指定一種類型的行動及其參數。

您必須提供您定義的行動類型的參數。您只可以提供一個行動類型及其參數。以下是支援的行動類型：

- `ADD_THINGS_TO_THING_GROUP`

- ENABLE_IOT_LOGGING
- PUBLISH_FINDING_TO_SNS
- REPLACE_DEFAULT_POLICY_VERSION
- UPDATE_CA_CERTIFICATE
- UPDATE_DEVICE_CERTIFICATE

AddThingsToThingGroup 參數

名稱	類型	描述
overrideDynamicGroups	布林值	選用。指定緩解行動如果此行動可以移動觸發一個或多個動態物件群組緩解行動的物件。此設定只在物件已經達到群組最大值時使用。
thingGroupNames	字串陣列	必要。觸發緩解行動新增物件的群組清單。 您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。 您必須提供至少一個群組名稱。 長度限制：最小長度為 1。長度上限為 128。 模式：[a-zA-Z0-9:_-]+

EnableIoTLogging 參數

名稱	類型	描述
logLevel	字串	必要。指定哪些類型的資訊需記錄。 有效的值，從詳細資訊到最小資訊，為 DEBUG、INFO、ERROR 和 WARN。您無法指定記錄層級 DISABLED。
roleArnForLogging	字串	必要。用於記錄 IAM 角色的 ARN。 長度下限為 20。長度上限為 2048。

PublishingFindingToSns 參數

名稱	類型	描述
topicArn	字串	必要。您要發佈結果主題的 ARN。

名稱	類型	描述
		長度下限為 20。長度上限為 2048。

ReplaceDefaultPolicyVersion 參數

名稱	類型	描述
`templateName`	字串	必要。要套用範本的名稱。 唯一支援的值為 `BLANK_POLICY`。

UpdateCACertificate 參數

名稱	類型	描述
`action`	字串	必要。您要套用到 CA 憑證的行動。 唯一支援的值為 `DEACTIVATE`。

UpdateDeviceCertificate 參數

名稱	類型	描述
`action`	字串	必要。您要套用到裝置憑證的行動。 唯一支援的值為 `DEACTIVATE`。

cli 輸出欄位：

名稱	類型	描述
`actionArn`	字串	緩解行動的 ARN。
`actionId`	字串	緩解行動的唯一識別符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

沒有發現特定名稱的緩解行動。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListMitigationActions

您可以使用 ListMitigationActions 命令來取得您 AWS 帳戶中的所有緩解行動的清單。

概要

```
aws iot list-mitigation-actions \
    [--action-type <value>] \
    [--max-results <value>] \
    [--next-token <value>]
```

輸入參數

名稱	類型	描述
`action-type`	字串	使用特定的行動類型指定一個值來限制緩解行動的結果。 支援的值為 `UPDATE_DEVICE_CERTIFICATE`、`UPDATE_CA` 或 `PUBLISH_FINDING_TO_SNS`。
`max-results`	integer	一次可回傳結果的數量上限。 預設為 25。有效的範圍為 1 到 250。
`next-token`	字串	用來擷取下一組結果的符記。

輸出 JSON：

```
{
   "actionIdentifiers": [
      {
         "actionArn": "string",
         "actionName": "string",
         "creationDate": number
      }
   ],
   "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
`actionIdentifiers`	對應	有關符合指定參數的緩解行動集合資訊。
`actionArn`	字串	緩解行動的 ARN。
`actionName`	字串	新的緩解行動的唯一識別符。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

名稱	類型	描述
creationDate	timestamp	建立緩解行動的日期和時間。
nextToken	字串	用來擷取下一組結果的符記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

範例

此範例列出區域中所有定義您的帳戶的緩解行動。

```
$ aws iot list-mitigation-actions
```

回應看起來會與以下內容相似。

```
{
    "actionIdentifiers": [
        {
            "actionName": "UpdateCACertName",
            "actionArn": "arn:aws:iot:us-east-1:123456789012:mitigationaction/
UpdateCACertName",
            "creationDate": 1560173584.521
        }
    ]
}
```

DescribeMitigationAction

您可以使用 DescribeMitigationAction 命令來檢視您的 AWS 帳戶中緩解行動的詳細資訊。

概要

```
aws iot describe-mitigation-action --action-name <value>
```

輸入參數

名稱	類型	描述
action-name	字串	唯一識別緩解行動易記的名稱。

名稱	類型	描述
		長度上限為 128。 模式：[a-zA-Z0-9_-]+

輸出 JSON：

```
{
   "actionArn": "string",
   "actionId": "string",
   "actionName": "string",
   "actionParams": {
      "addThingsToThingGroupParams": {
         "overrideDynamicGroups": boolean,
         "thingGroupNames": [ "string" ]
      },
      "enableIoTLoggingParams": {
         "logLevel": "string",
         "roleArnForLogging": "string"
      },
      "publishFindingToSnsParams": {
         "topicArn": "string"
      },
      "replaceDefaultPolicyVersionParams": {
         "templateName": "string"
      },
      "updateCACertificateParams": {
         "action": "string"
      },
      "updateDeviceCertificateParams": {
         "action": "string"
      }
   },
   "actionType": "string",
   "creationDate": number,
   "lastModifiedDate": number,
   "roleArn": "string"
}
```

每個緩解行動只有一個類型，因此只會顯示一組的參數。

CLI 輸出欄位：

名稱	類型	描述
`actionArn`	字串	緩解行動的 ARN。
`actionId`	字串	緩解行動的唯一識別符。
`actionName`	字串	新的緩解行動的唯一易記名稱。 長度上限為 128。 模式：[a-zA-Z0-9_-]+
`actionParams`	對應	定義套用行動的類型和該緩解行動的參數。

名稱	類型	描述
addThingsToThingGroupParams	對應	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
overrideDynamicGroups	布林值	指定緩解行動如果此行動可以移動觸發緩解行動的物件，即使該物件為一個或多個動態物件群組的一部分。
thingGroupNames	字串陣列	觸發緩解行動新增物件的群組清單。 您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。 您必須提供至少一個群組名稱。 長度下限為 1。長度上限為 128。 模式：[a-zA-Z0-9:_-]+
enableIoTLoggingParams	對應	參數定義緩解行動，讓 AWS IoT 能以指定的細節層級記錄詳細資訊。
logLevel	字串	指定記錄的資訊類型。 有效的值，從詳細資訊到最小資訊，為 DEBUG、INFO、ERROR、WARN 和 DISABLED。
roleArnForLogging	字串	用於記錄 IAM 角色的 ARN。 長度下限為 20。長度上限為 2048。
publishFindingToSnsParams	對應	參數定義緩解行動，發佈結果到 Amazon SNS。您可以實作自訂的行動來回應 Amazon SNS 訊息。
topicArn	字串	您要發佈結果主題的 ARN。 長度下限為 20。長度上限為 2048。
replaceDefaultPolicyVersionParams	對應	參數定義緩解行動，新增空白政策來限制權限。
templateName	字串	要套用範本的名稱。 唯一支援的值為 BLANK_POLICY。

名稱	類型	描述
updateCACertificateParams	對應	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。
action	字串	您要套用到 CA 憑證的行動。 唯一支援的值為 DEACTIVATE。
updateDeviceCertificateParams	對應	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
action	字串	您要套用到裝置憑證的行動。 唯一支援的值為 DEACTIVATE。
actionType	字串	正在套用的行動類型。
creationDate	timestamp	建立緩解行動的日期和時間。
lastModifiedDate	timestamp	最近變更緩解行動的日期和時間。
roleArn	字串	當套用此緩解行動角色使用的 ARN。

錯誤

ResourceNotFoundException

沒有發現特定名稱的緩解行動。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

範例

此範例定義緩解行動的名稱為 UpdateCACertName。

```
$ aws iot describe-mitigation-action --action-name UpdateCACertName
```

回應看起來會與以下內容相似。

```
{
    "actionName": "UpdateCACertName",
    "actionType": "UPDATE_DEVICE_CERTIFICATE",
    "actionArn": "arn:aws:iot:us-east-1:123456789012:mitigationaction/UpdateCACertName",
```

```
    "actionId": "6a22b98e-0e27-4396-9b25-637d04959429",
    "roleArn": "arn:aws:iam::123456789012:role/MitigationActionsValidRole",
    "actionParams": {
        "updateDeviceCertificateParams": {
            "action": "DEACTIVATE"
        }
    },
    "creationDate": 1560173584.521,
    "lastModifiedDate": 1560173584.521
}
```

DeleteMitigationAction

您可以使用 DeleteMitigationAction 命令從 AWS 帳戶移除緩解行動。為了讓 DeleteMitigationAction 命令等冪，如果您嘗試刪除不存在的緩解行動，不會擲回 `ResourceNotFoundException`。反之，當動作名稱不存在時 DeleteMitigationAction 會回傳成功。

概要

```
aws iot delete-mitigation-action --action-name <value>
```

輸入參數

名稱	類型	描述
`action-name`	字串	唯一識別緩解行動易記的名稱。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

StartAuditMitigationActionsTask

您可以使用 StartAuditMitigationActionsTask 命令來套用一組緩解行動到一個或多個稽核結果。

概要

```
aws iot  start-audit-mitigation-actions-task \
    --task-id <value> \
    --audit-task-id <value> \
    --audit-checks-to-action-mapping <value> \
```

```
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
   "auditCheckToActionsMapping": {
      "string" : [ "string" ]
   },
   "clientRequestToken": "string",
   "target": {
      "auditTaskId": "string",
      "findingIds": [ "string" ]
   }
}
```

輸入參數

名稱	類型	描述
taskId	字串	稽核緩解任務的唯一識別符。 長度上限為 128。 模式：[a-zA-Z0-9_-]+
auditCheckToActionsMapping	字串陣列對應的字串	指定行動清單套用到特定的稽核檢查。 陣列必須至少包含 1 個但不超過 5 個的成員。 長度上限為 128。 模式：[a-zA-Z0-9_-]+
clientRequestToken	字串	每個稽核緩解任務都必須有唯一的用戶端請求字符。如果您嘗試啟動新的任務與現有任務相同的任務 ID，但使用不同的字符，則會發生例外狀況。如果您省略這個值，會自動產生唯一用戶端請求字符。 長度下限為 1。長度上限為 64。 模式：[a-zA-Z0-9_-]+
target	對應	指定稽核結果套用到哪些緩解行動。您可以將其套用到稽核任務結果或一組結果。
auditTaskId	字串	您要套用一組行動的稽核結果唯一識別符。auditCheckToReasonCodeFilter 可以從稽核進一步篩選結果。如果您要限制行動到特定的一組結果，請改為使用 findingIds。 長度上限為 40。

名稱	類型	描述
		模式：[a-zA-Z0-9\-]+
findingIds	字串陣列	如果任務套用緩解行動到一個或多個列出的結果，這個值可唯一識別其結果。 陣列成員：最少為 1 個。最多為 25 個。 最大長度為每個項目 128。 模式：[a-zA-Z0-9_-]+

輸出 JSON：

```
{
   "taskId": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
taskId	字串	新建立的稽核緩解行動任務的唯一識別符。 如果您需要取消任務 (CancelAuditMitigationActionsTask) 或如果您要使用 DesribeAuditMitigationAtionsTask 檢視任務詳細資訊，請使用此識別符。

錯誤

TaskAlreadyExistsException

如果您嘗試開始已存在相同任務 id 的任務，但是不同的 clientRequestToken，則會發生此例外狀況。

InvalidRequestException

請求的內容無效。

LimitExceededException

如果您嘗試開始超過 10 個緩解動作任務，則會發生此例外狀況。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

範例

此範例開始任務，從稽核 taskId 為 `aef320b958891041e0c60c088afac64c` 且稽核檢查為 `CA_CERTIFICATE_EXPIRING_CHECK`。套用您定義的稽核結果到 `UpdateCACertAction`。

```
$ aws iot start-audit-mitigation-actions-task --task-id "myActionsTaskId" --target
 "auditTaskId=aef320b958891041e0c60c088afac64c" --audit-check-to-actions-mapping
 "CA_CERTIFICATE_EXPIRING_CHECK=UpdateCACertAction" --client-request-token "adhadhahda"
```

回傳的結果如下所示。

```
{
    "taskId": "myActionsTaskId"
}
```

CancelAuditMitigationActionsTask

您可以使用 CancelAuditMitigationActionsTask 命令來停止執行尚未完成的稽核緩解任務。

概要

```
aws iot cancel-audit-mitigation-actions-task --task-id <value>
```

輸入參數

名稱	類型	描述
`task-id`	字串	您要取消的稽核緩解任務的唯一識別符。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

錯誤

`ResourceNotFoundException`

沒有發現具有特定任務 ID 的稽核緩解任務。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditMitigationActionsExecutions

您可以使用 ListAuditMitigationActionsExecutions 命令來顯示已開始稽核緩解任務的詳細資訊。

概要

```
aws iot list-audit-mitigation-actions-executions \
    [--action-status <value>] \
    --finding-id <value> \
    [--max-results <value>] \
    [--next-token <value>]  \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

輸入參數

名稱	類型	描述
action-status	字串	使用特定的狀態指定此篩選條件來限制執行的結果。 支援的值為 IN_PROGRESS、COMPLETED、FAILED、CANCE 和 PENDING。
finding-id	字串	指定此篩選條件來限制結果，套用到特定的稽核結果。 長度上限為 128。 模式：[a-zA-Z0-9_-]+
max-results	數字	一次可回傳結果的數量上限。預設為 25。 有效範圍：下限為 1，上限為 250。
next-token	字串	下一組結果的字符。
task-id	字串	您要顯示詳細資訊的稽核緩解任務的唯一識別符。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

輸出 JSON：

```
{
   "actionsExecutions": [
      {
         "actionId": "string",
         "actionName": "string",
         "endTime": number,
         "errorCode": "string",
         "findingId": "string",
         "message": "string",
         "startTime": number,
         "status": "string",
         "taskId": "string"
      }
   ],
   "nextToken": "string"
```

```
}
```

CLI 輸出欄位：

名稱	類型	描述
actionsExecutions	對應	您 AWS 帳戶已經開始的稽核緩解任務執行的資訊集合。
actionId	字串	正在由任務套用的緩解動作的唯一識別符。
actionName	字串	正在由任務套用的緩解動作的易記名稱。
endTime	timestamp	任務完成或取消的日期和時間。
errorCode	字串	如果發生錯誤，程式碼指出錯誤類型。
findingId	字串	唯一識別碼用於任務和關聯套用的緩解行動的結果。
message	字串	如果發生錯誤，訊息會描述該錯誤。
startTime	timestamp	任務開始的日期和時間。
status	字串	正在執行任務目前的狀態。支援的值為 IN_PROGRESS、COMPLETED、FAILED、CANCE 和 PENDING。
taskId	字串	套用到任務緩解動作的唯一識別符。
nextToken	字串	下一組結果的字符。

狀態欄位可能具有下列值：

狀態	代表什麼意思
IN_PROGRESS	AWS IoT Device Defender 會套用到緩解任務的結果。
COMPLETED (已完成)	緩解行動成功套用到結果。
FAILED (失敗)	緩解行動套用到結果失敗。錯誤代碼提供更多資訊。
CANCELED (已取消)	執行的行動已取消，因為使用者取消任務。
略過	行動執行已略過，因為清單中其中一項任務失敗。
待定	尚未開始的執行行動。

可能回傳以下錯誤代碼：

INSUFFICIENT_PERMISSIONS

- 提供緩解行動的 roleArn 沒有套用動行動的權限。

INVALID_STATE_OF_RESOURCE

- 結果中的資源不在允許緩解動作成功套用的狀態中。如果物件已經在允許群組的最大值中，則會發生此錯誤 ADD_THINGS_TO_THING_GROUP 動作類型。如果憑證已經遭到撤銷，則會發生錯誤 UPDATE_DEVICE_CERTIFICATE 和 UPDATE_CA_CERTIFICATE 緩解動作類型。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditMitigationActionsTasks

您可以使用 ListAuditMitigationActionsTasks 命令以獲得稽核緩解動作任務符合指定篩選條件的清單。

概要

```
aws iot list-audit-mitigation-actions-tasks
    [--audit-task-id <value>] \
    --end-time <value> \
    [--finding-id <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    --start-time <value> \
    [--task-status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

輸入參數

名稱	類型	描述
audit-task-id	字串	您要顯示稽核緩解任務的唯一識別符。 長度下限為 1。長度上限為 40。 模式：[a-zA-Z0-9_-]+
end-time	timestamp	必要。指定此篩選條件，列出在此日期結束或之前的任務。此日期不得大於過去 90 天。
finding-id	字串	指定此篩選條件，列出套用到特定結果的任務。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

名稱	類型	描述
max-results	數字	一次可回傳結果的數量上限。預設為 25。 有效範圍：最小值為 1。最大值為 250。
next-token	字串	下一組結果的字符。
start-time	timestamp	必要。指定此篩選條件，將結果限制在此日期和時間點上開始或之後的任務。此日期不得大於過去 90 天。
task-status	字串	指定此篩選條件，將結果限制在指定狀態的任務中。 支援的值為：IN_PROGRESS、COMPLETED、FAILED 和 CANCELED。

輸出 JSON：

```
{
   "nextToken": "string",
   "tasks": [
      {
         "startTime": number,
         "taskId": "string",
         "taskStatus": "string"
      }
   ]
}
```

CLI 輸出欄位：

名稱	類型	描述
nextToken	字串	下一組結果的字符。
tasks	對應	符合篩選條件的稽核緩解動作任務集合。
startTime	timestamp	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
taskId	字串	任務的唯一識別符。
taskStatus	字串	任務目前的狀態。

錯誤

InvalidRequestException

請求的內容無效。如果您指定的日期大於過去 90 天則會發生此錯誤。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

範例

此範例列出在特定時間內執行的稽核緩解任務。

```
$ aws iot list-audit-mitigation-actions-tasks --start-time 1560001663 --end-time
 1560174463
```

回傳的結果如下所示。

```
{
    "tasks": [
        {
            "taskId": "actionsTaskId",
            "startTime": 1560174232.07,
            "taskStatus": "CANCELED"
        },
        {
            "taskId": "4e8acaf8-b7f0-4484-9b0d-01b979e61d8d",
            "startTime": 1560060994.965,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "0e8f5a95-e43f-43fa-9aa4-57ff3efb946d",
            "startTime": 1560060860.243,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "92d60009-33a1-45a7-a3c6-b28e938ec68a",
            "startTime": 1560060707.653,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "bdea63c8-58bd-4798-9cb3-e949c7f1969a",
            "startTime": 1560060531.123,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "61e146ef-69f4-41bf-9d37-5d667f72ef3d",
            "startTime": 1560060388.035,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "2ef289dc-9bbd-422b-95ba-d96b7e626a60",
            "startTime": 1560060256.695,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "a6cdc16b-6be3-4800-bafa-2b86d3f5d639",
            "startTime": 1560060097.613,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "7ccf351b-e560-4eb2-8796-1dc1bcb25396",
```

```
            "startTime": 1560059925.477,
            "taskStatus": "COMPLETED"
        },
        {
            "taskId": "8f69ee5d-3e35-4c91-88e8-3c5f84c96fde",
            "startTime": 1560059345.473,
            "taskStatus": "COMPLETED"
        }
    ]
}
```

DescribeAuditMitigationActionsTask

您可以使用 DescribeAuditMitigationActionsTask 命令來顯示已開始稽核緩解任務的詳細資訊。

概要

```
aws iot describe-audit-mitigation-actions-task --taskId <value>
```

輸入參數

名稱	類型	描述
taskId	字串	必要。您要顯示詳細資訊的稽核緩解任務的唯一識別符。 長度上限為 128。 模式：[a-zA-Z0-9_-]+

輸出 JSON：

```
{
   "actionsDefinition": [
      {
         "actionParams": {
            "addThingsToThingGroupParams": {
               "overrideDynamicGroups": boolean,
               "thingGroupNames": [ "string" ]
            },
            "enableIoTLoggingParams": {
               "logLevel": "string",
               "roleArnForLogging": "string"
            },
            "publishFindingToSnsParams": {
               "topicArn": "string"
            },
            "replaceDefaultPolicyVersionParams": {
               "templateName": "string"
            },
            "updateCACertificateParams": {
               "action": "string"
            },
            "updateDeviceCertificateParams": {
               "action": "string"
            }
         },
         "id": "string",
```

```
            "name": "string",
            "roleArn": "string"
         }
      ],
      "auditCheckToActionsMapping": {
         "string" : [ "string" ]
      },
      "endTime": number,
      "startTime": number,
      "target": {
         "auditCheckToReasonCodeFilter": {
            "string" : [ "string" ]
         },
         "auditTaskId": "string",
         "findingIds": [ "string" ]
      },
      "taskStatistics": {
         "string" : {
            "canceledFindingsCount": number,
            "failedFindingsCount": number,
            "skippedFindingsCount": number,
            "succeededFindingsCount": number,
            "totalFindingsCount": number
         }
      },
      "taskStatus": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
`>actionsDefinition`	對應	透過此稽核緩解行動任務的一組動作 (及其中繼資料)。
`actionParams`	對應	套用緩解行動時使用的參數。
`addThingsToThingGroupParam`	對應	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
`overrideDynamicGroups`	布林值	指定緩解行動如果此行動可以移動觸發緩解行動的物件，即使該物件為一個或多個動態物件群組的一部分。
`thingGroupNames`	字串陣列	此稽核緩解行動任務新增物件到目標的群組清單。
`enableIoTLoggingParams`	對應	參數定義緩解行動，讓 AWS IoT 能以指定的細節層級記錄詳細資訊。
`logLevel`	字串	指定記錄的資訊類型。
`roleArnForLogging`	字串	用於記錄 IAM 角色的 ARN。
`publishFindingToSnsParams`	對應	參數定義緩解行動，發佈結果到 Amazon SNS。您可以實作自訂的行動來回應 Amazon SNS 訊息。

名稱	類型	描述
topicArn	字串	您要發佈結果主題的 ARN。
replaceDefaultPolicyVersionParams	對應	參數定義緩解行動，新增空白政策來限制權限。
templateName	字串	要套用範本的名稱。 唯一支援的值為 BLANK_POLICY。
updateCACertificateParams	對應	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。
action	字串	您要套用到 CA 憑證的行動。 唯一支援的值為 DEACTIVATE。
updateDeviceCertificateParams	對應	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
action	字串	您要套用到裝置憑證的行動。 唯一支援的值為 DEACTIVATE。
id	字串	緩解行動的唯一識別符套用為此稽核緩解任務的一部分。
name	字串	緩解行動的易記名稱套用為此稽核緩解任務的一部分。
roleArn	字串	套用此緩解任務的角色 ARN。
auditCheckToActionsMapping	對應	對於稽核檢查的類型，指定哪些緩解行動會套用於此稽核緩解任務。
endTime	timestamp	如果稽核緩解任務完成或遭取消，執行停止的日期和時間。
startTime	timestamp	此稽核緩解行動任務開始執行的日期和時間。
target	對應	套用目標作為此稽核緩解行動任務部分的資訊。
auditCheckToReasonCodeFilter	對應	指定一組稽核檢查和原因代碼用於辨識稽核結果，套用到此稽核緩解行動任務。
auditTaskId	字串	指定套用到哪個稽核結果的稽核緩解行動任務。
findingIds	字串陣列	指定一組稽核結果套用到稽核緩解行動任務。
taskStatistics	對應	此稽核緩解行動任務的一組執行的統計資料。

名稱	類型	描述
canceledFindingsCount	數字	如果任務遭取消，此稽核緩解行動任務目標中的結果數量不會套用到緩解行動。
failedFindingsCount	數字	此稽核緩解行動任務目標中的結果數量套用到緩解行動失敗。
skippedFindingsCount	數字	此稽核緩解行動任務目標中的結果數量無法套用到緩解行動，因為遭到任務套用篩選條件排除。
succeededFindingsCount	數字	此稽核緩解行動任務目標中的結果數量套用到緩解行動沒有錯誤或取消。
totalFindingsCount	數字	此稽核緩解行動任務在目標中的結果總數量。
taskStatus	字串	稽核緩解行動任務的目前狀態。如果一個或多個結果行動套用失敗，則狀態為失敗。

錯誤

`ResourceNotFoundException`

沒有找到指定的 `taskId` 稽核緩解任務。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

偵測

AWS IoT Device Defender Detect 可讓您識別不尋常的行為，其可能透過監控裝置的行為來表示遭入侵裝置。結合使用雲端指標 (來自 AWS IoT) 和裝置端指標 (從安裝在裝置上的代理程式)，您可以偵測連線模式的變更、溝通未經授權或未識別端點的裝置，以及傳入和傳出裝置流量模式中的變更。您可以建立安全性描述檔，其中包含定義預期裝置行為，並將它們指派給一組裝置或套用到機群中的所有裝置。AWS IoT Device Defender Detect 使用這些安全性描述檔來偵測異常，並透過 Amazon CloudWatch 指標和 Amazon Simple Notification Service 通知傳送提醒。

AWS IoT Device Defender Detect 能夠偵測經常在連線裝置中找到的安全問題：

- 流量從裝置到已知惡意的 IP 地址或未經授權的端點，其表示潛在惡意命令和控制管道。
- 異常流量如傳出流量的峰值，其表示有某個裝置正遭受 DDoS 攻擊。
- 具遠端管理界面和可遠端存取連接埠的裝置。
- 傳送到您帳戶的訊息速率峰值。(例如，從 Rogue 裝置可導致每條訊息過多的費用)。

使用案例:

測量攻擊表面

您可以使用 AWS IoT Device Defender Detect 測量您裝置的攻擊表面。例如，您可以透過服務連接埠識別裝置，其通常是攻擊活動的目標 (在連接埠 23/2323 上執行的 telnet 服務、在連接埠 22 上執行的 SSH 服務、在連接埠 80/443/8080/8081 上執行的 HTTP/S 服務)。雖然這些服務連接埠可能有使用於裝置的合法理由，它們通常也是對手攻擊表面的一部分，附帶關聯的風險。在 Detect 提醒您攻擊面後，您可以決定它將攻擊面縮到最小 (排除未使用的網路服務)，或執行額外的評估來識別安全弱點 (例如，設定常見、預設或弱密碼的 telnet)。

偵測裝置的異常行為的可能安全性根本原因

您可以使用 AWS IoT Device Defender Detect 提醒您未預期的裝置行為指標 (開放連接埠的數目、連線數目、未預期的開放連接埠、未預期 IP 地址的連線)，這可能表示安全漏洞。例如，比預期的 TCP 連線更高可能表示裝置正用於 DDoS 攻擊。在您預期以外的連接埠接聽的處理序可能表示安裝於裝置上進行遠端控制的後門。您可以使用 Detect 探查裝置機群的運作狀態，並驗證您的安全假設 (例如，沒有裝置監聽連接埠 23 或 2323)。

偵測設定錯誤的裝置

從裝置到您的帳戶傳送的訊息數量或大小的峰值，可能表示設定錯誤的裝置。這類裝置可能增加每條訊息的費用。同樣地，多次授權失敗的裝置可能需要重新設定政策。

概念

指標

AWS IoT Device Defender Detect 使用指標來偵測異常行為。偵測會比較指標的回報值與您提供的預期值。這些指標來自兩個來源：雲端指標和裝置端指標：

透過使用雲端端指標如授權失敗的數量，或裝置透過 AWS IoT 傳送或接收的訊息數量或大小，偵測 AWS IoT 網路上的異常行為。

AWS IoT Device Defender Detect 也可以收集、彙總及監控 AWS IoT 裝置產生的指標資料，例如，裝置接聽的連接埠、已傳送的位元組或封包數，或裝置的 TCP 連線。

您可以僅透過雲端指標來使用 AWS IoT Device Defender Detect。若要使用裝置端指標，您必須先在您的 AWS IoT 連線裝置或裝置閘道上部署 AWS IoT SDK，以收集指標並將指標傳送到 AWS IoT。請參閱從裝置傳送指標 (p. 614)。

安全性描述檔

安全性描述檔定義一組裝置或您帳戶中的所有裝置的異常行為 (實物群組)，並指定當偵測到異常狀況時要採取哪些動作。您可以使用 AWS IoT 主控台或 API 命令建立安全性描述檔，並將其與一組裝置產生關聯。AWS IoT Device Defender Detect 開始記錄安全性相關的資料，並利用安全性描述檔中定義的行為，來偵測裝置行為的異常。

行為

行為會告訴 AWS IoT Device Defender Detect 如何辨識裝置的異常行為。每個行為由名稱、指標、運算子和一個值或統計閾值所組成。對於某些指標，也需要時段 (`durationSeconds`)。任何不符合所定義的行為說明的裝置動作都會觸發提醒。

提醒

當偵測到異常時，提醒通知可以透過 CloudWatch 指標 (請參閱 AWS IoT 指標 (p. 202)) 或 SNS 通知傳送。AWS IoT CDM 主控台也會顯示提醒通知，以及更多有關提醒的資訊，以及裝置的提醒記錄。當監控裝置停止呈現異常行為，或當它造成提醒需求但停止報告一段延長的時期時，也會傳送提醒。

行為

安全性描述檔包含一組行為。每個行為都包含一個指標，其會為您帳戶中的一組裝置或所有裝置指定正常行為。(請參閱 指標 (p. 601) 和 CreateSecurityProfile (p. 620)。)

以下說明一些用於 `behavior` 定義的欄位：

`name`

行為的名稱。

`metric`

使用的指標名稱 (也就是，行為所測量的項目)。

`criteria`

可判斷裝置對於 `metric` 的操作是否正確的條件。

`comparisonOperator`

將測量的實物 (`metric`) 關聯到條件 (`value` or `statisticalThreshold`) 的運算子。

可能值為："less-than"、"less-than-equals"、"greater-than"、"greater-than-equals"、"in-cidr-set"、"not-in-cidr-set"、"in-port-set" 和 "not-in-port-set"。並不是所有運算子都對每個指標有效。CIDR 集和連接埠的運算子只適用於與包含這類實體的指標搭配使用。

`value`

相較於 `metric` 的值。根據指標的類型，這應該包含 `count` (一個值)、`cidrs` (CIDR 清單) 或 `ports` (連接埠清單)。

`statisticalThreshold`

判定行為違規的統計閾值。此欄位包含的 `statistic` 欄位具有下列可能的值：「p0」、「p0.1」、「p0.01」、「p1」、「p10」、「p50」、「p90」、「p99」、「p99.9」、「p99.99」或「p100」。

此 `statistic` 以百分比表示。它可以解析成用來判斷該行為符合哪個合規的值。系統會在指定的持續時間 (`durationSeconds`)，從與此安全性設定檔關聯的所有報告裝置一或多次收集指標，並會根據該資料計算百分比。之後，系統會收集裝置的衡量值並在相同的持續時間累積這些值。如果裝置產生的值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。

百分比會指出被視為落在關聯值以下的所有衡量值百分比。例如，如果與「p90」 (第 90 個百分比) 的值是 123，則 90% 的所有衡量值低於 123。

`durationSeconds`

針對具有時間維度的這些條件，使用此項來指定評估行為的期間 (例如，`NUM_MESSAGES_SENT`)。針對 `statisticalThreshhold` 指標比較，這是收集所有裝置衡量值以判斷 `statisticalThreshold` 值，然後收集每個裝置衡量值以判斷其行為在比較中排名的所經期間。

`consecutiveDatapointsToAlarm`

如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。(這與 AWS IoT 主控台不同，在該主控台中預設值為 3，但可以覆寫。)

`consecutiveDatapointsToClear`

如果發生警示，而違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。(這與 AWS IoT 主控台不同，在該主控台中預設值為 3，但可以覆寫。)

指標

aws:message-byte-size

訊息中的位元組數。

More info (1)

使用此指標，以指定從裝置傳輸到 AWS IoT 的每則訊息的最大或最小大小 (以位元組為單位)。

來源：雲端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：位元組

範例：

```
{
  "name": "Max Message Size",
  "metric": "aws:message-byte-size",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 1024
    },
    "consecutiveDatapointsToAlarm": 3,
    "consecutiveDatapointsToClear": 3
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "Large Message Size",
  "metric": "aws:message-byte-size",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 3,
    "consecutiveDatapointsToClear": 3
  }
}
```

如果在三個連續 5 分鐘的期間內，某裝置傳輸的訊息累積大小超過針對 90% 的所有其他裝置而量測的訊息累積大小，所有其他裝置會為此安全性描述檔行為進行報告，而且出現警示。

aws:num-messages-received/aws:num-messages-sent

裝置在特定期間接收或傳送的訊息數量。

More info (2)

使用此指標，以指定在指定期間內可在每個裝置和 AWS IoT 之間傳送或接收的最大或最小訊息數量。

來源：雲端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：訊息

持續時間：非負整數、有效值為 300、600、900、1800 或 3600 秒

範例：

```
{
  "name": "Out bound message count",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 50
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 2
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "Out bound message rate",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p99"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 2
  }
}
```

aws:all-bytes-out

在特定期間內從裝置傳出位元組的數量。

More info (3)

使用此指標，以指定裝置應該傳送的傳出流量的最大或最小數量，其在指定期間內以位元組為單位測量。

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：位元組

持續時間：非負整數、有效值為 300、600、900、1800 或 3600 秒

範例：

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 4096
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 5,
    "consecutiveDatapointsToClear": 4
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 900,
    "consecutiveDatapointsToAlarm": 5,
    "consecutiveDatapointsToClear": 4
  }
}
```

aws:all-bytes-in

在特定期間內傳入裝置的位元組數量。

More info (4)

使用此指標，以指定裝置應該接收的傳入流量的最大或最小數量，其在指定期間內以位元組為單位測量。

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：位元組

持續時間：非負整數、有效值為 300、600、900、1800 或 3600 秒

範例：

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
```

```
      "count": 4096
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 3
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 3
  }
}
```

aws:all-packets-out

在特定期間內從裝置傳出封包的數量。

More info (5)

使用此指標，以指定裝置應該在指定期間內傳送的總傳出流量的最大或最小數量。

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：封包數

持續時間：非負整數。有效值為 300、600、900、1800 或 3600 秒。

範例：

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-packets-out",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 100
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 3
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-packets-out",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 3
  }
}
```

aws:all-packets-in

在特定期間內傳入裝置的封包數量。

More info (6)

使用此指標，以指定裝置應該在指定期間內接收的總傳入流量的最大或最小數量。

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：封包數

持續時間：非負整數。有效值為 300、600、900、1800 或 3600 秒。

範例：

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-packets-in",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 100
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 1
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-packets-in",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
```

```
    "consecutiveDatapointsToClear": 1
  }
}
```

aws:num-authorization-failures

在特定時間內的授權數。

More info (7)

使用此指標，以指定允許在指定期間內用於每一裝置的最大授權失敗數。當定義從裝置到 AWS IoT 的請求時發生授權失敗，例如，如果裝置嘗試發佈到一個不具備足夠許可的主題時。

來源：雲端

單位：失敗數

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：失敗數

持續時間：非負整數。有效值為 300、600、900、1800 或 3600 秒。

範例：

```
{
  "name": "Authorization Failures",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 5
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 1
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "Authorization Failures",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 1
  }
}
```

aws:source-ip-address

裝置連接到 AWS IoT 的來源 IP 地址。

More info (8)

使用此指標來指定一組允許 (之前稱為白名單) 或拒絕 (之前稱為 blacklisted) 的 CIDR，每個裝置必須或不得連接到 AWS IoT。

來源：雲端

運算子：in-cidr-set | not-in-cidr-set

值：CIDR 清單

單位：N/A

範例：

```
{
  "name": "Blacklisted source IPs",
  "metric": "aws:source-ip-address",
  "criteria": {
    "comparisonOperator": "not-in-cidr-set",
    "value": {
      "cidrs": [ "12.8.0.0/16", "15.102.16.0/24" ]
    }
  }
}
```

aws:destination-ip-addresses

一組 IP 目的地。

More info (9)

使用此指標來指定一組允許 (之前稱為白名單) 或拒絕 (之前稱為黑名單) 的 CIDR，每個裝置必須或不得進行通訊。

來源：裝置端

運算子：in-cidr-set | not-in-cidr-set

值：CIDR 清單

單位：N/A

範例：

```
{
  "name": "Blacklisted destination IPs",
  "metric": "aws:destination-ip-addresses",
  "criteria": {
    "comparisonOperator": "not-in-cidr-set",
    "value": {
      "cidrs": [ "12.8.0.0/16", "15.102.16.0/24" ]
    }
  }
}
```

aws:listening-tcp-ports / aws:listening-udp-ports

該裝置接聽的 TCP 或 UDP 連接埠。

More info (10)

使用此指標來指定一組允許 (之前稱為白名單) 或拒絕 (之前稱為黑名單) 的 TCP/UDP 連接埠，每個裝置必須或不得監聽。

來源：裝置端

運算子：in-port-set | not-in-port-set

值：一組連接埠

單位：N/A

範例：

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 443, 80 ]
    }
  }
}
```

aws:num-listening-tcp-ports / aws:num-listening-udp-ports

該裝置正在接聽的 TCP 或 UDP 數。

More info (11)

使用此指標，以指定每個裝置應該接聽的最大或最小數量的 TCP 或 UDP 連接埠數。

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：連接埠

範例：

```
{
  "name": "Max TCP Ports",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 4
    },
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 1
```

```
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "Max TCP Ports",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 2,
    "consecutiveDatapointsToClear": 1
  }
}
```

aws:num-established-tcp-connections

適用於裝置的 TCP 連線數。

More info (12)

使用此指標，以指定每個裝置應該擁有的最大或最小數量的作用中 TCP 連線數。(所有 TCP 狀態)

來源：裝置端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：連線數

範例：

```
{
  "name": "TCP Connection Count",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 3
    },
    "consecutiveDatapointsToAlarm": 3,
    "consecutiveDatapointsToClear": 3
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "TCP Connection Count",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "comparisonOperator": "less-than",
    "statisticalThreshold": {
      "statistic": "p90"
```

```
    },
    "durationSeconds": 900,
    "consecutiveDatapointsToAlarm": 3,
    "consecutiveDatapointsToClear": 3
  }
}
```

aws:num-connection-attempts

裝置已在指定時段嘗試建立連線的次數。

More info (13)

使用此指標來指定每個裝置的連線嘗試次數上限或下限。成功和失敗的嘗試都會列入計算。

來源：雲端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：嘗試連線的次數

持續時間：非負整數。有效值為 300、600、900、1800 或 3600 秒。

範例：

```
{
  "name": "Connection Attempts",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "comparisonOperator": "greater-than",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 2
  }
}
```

使用 `statisticalThreshold` 的範例：

```
{
  "name": "Connection Attempts",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "comparisonOperator": "greater-than",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 2
  }
}
```

aws:num-disconnects

裝置在特定時段與 AWS IoT 中斷連線的次數。

More info (14)

使用此指標來指定裝置在特定時段與 AWS IoT 中斷連線的次數上限或下限。

來源：雲端

運算子：less-than | less-than-equals | greater-than | greater-than-equals

值：是非負整數。

單位：中斷連線數

持續時間：非負整數。有效值為 300、600、900、1800 或 3600 秒。

範例：

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "greater-than",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 2
  }
}
```

使用 statisticalThreshold 的範例：

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "greater-than",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 2
  }
}
```

監控未註冊裝置的行為

對於未在 AWS IoT 登錄中註冊的裝置，AWS IoT Device Defender 能夠識別異常行為。您可以針對下列其中一個目標類型，定義特有的安全性描述檔：

- 所有裝置
- 所有已註冊的裝置 (AWS IoT 登錄中的實物)
- 所有未註冊的裝置
- 實物群組中的裝置

安全性描述檔會為您帳戶中的裝置定義一組預期的行為，並指定偵測到異常時所要採取的動作。安全性描述檔應該附加到最特定的目標，讓您能夠精確控制目前根據該描述檔評估哪些裝置。

未註冊的裝置必須在裝置生命週期中提供一致的 MQTT 用戶端識別符或實物名稱 (適用於報告裝置指標的裝置)，讓所有違規和指標歸屬於相同的裝置。

> Important
>
> 如果實物名稱包含控制字元，或無誤名稱超過 128 個位元組的 UTF-8 編碼字元，則會拒絕裝置所回報的訊息。

如何使用 AWS IoT Device Defender Detect

1. 您可以僅透過雲端指標來使用 AWS IoT Device Defender Detect，但如果打算使用裝置報告指標，則必須先在您的 AWS IoT 連線裝置或裝置閘道上部署 AWS IoT SDK。如需詳細資訊，請參閱 從裝置傳送指標 (p. 614)。
2. 請考慮在定義行為及建立警示之前，檢視您的裝置所產生的指標。AWS IoT 可以從您的裝置收集指標，以便您先針對一組裝置或您帳戶中的所有裝置識別正常或異常行為。使用 CreateSecurityProfile (p. 620)，但僅指定您感興趣的那些 `additionalMetricsToRetain`。請勿在此時指定任何 `behaviors`。

   使用 AWS IoT 主控台來查看您的裝置指標，以查看何者構成您裝置的典型行為。
3. 為您的安全性描述檔建立一組行為。包含指標的行為，其為您帳戶中的一組裝置或所有裝置指定正常行為。如需詳細資訊 (包括範例)，請參閱 指標 (p. 601)。在建立一組行為之後，您可以使用 ValidateSecurityProfileBehaviors (p. 648) 來進行驗證。
4. 使用 CreateSecurityProfile (p. 620) 建立包含您行為的安全性描述檔。當裝置違反行為時，您可以使用 `alertTargets` 參數將提醒傳送到目標 (SNS 主題)。(如果您使用 SNS 傳送提醒，這些都會計入您的帳戶的 SNS 限制。大量突增違規可能耗盡您的產能)。 您也可以使用 CloudWatch 指標來檢查違規。如需詳細資訊，請參閱 AWS IoT 指標 (p. 202)。
5. 使用 AttachSecurityProfile (p. 619) 將安全性描述檔附加到一組裝置 (實物群組)、您帳戶中所有已註冊的實物，所有未註冊的實物，或所有裝置。AWS IoT Device Defender Detect 會開始檢查異常行為，若偵測到違規行為，便會傳送提醒。如果您預期與不在您帳戶的實物登錄中的行動裝置互動，則可將安全性描述檔附加到所有未註冊的實物。您可以定義適用於不同裝置群組的不同行為，以符合您的需求。

   若要將安全性描述檔附加到一組裝置，您必須指定包含該裝置的實物群組的 ARN。此實物群組 ARN 的格式如下：

   ```
   arn:aws:iot:<region>:<accountid>:thinggroup/<thing-group-name>
   ```

   若要將安全性描述檔附加到帳戶中所有已註冊的實物 (忽略未註冊的實物)，您必須指定具下列格式的 ARN：

   ```
   arn:aws:iot:<region>:<accountid>:all/registered-things
   ```

   若要將安全性描述檔附加到所有未註冊的實物，您必須指定具下列格式的 ARN：

   ```
   arn:aws:iot:<region>:<accountid>:all/unregistered-things
   ```

   若要將安全性描述檔附加到所有裝置，您必須指定具下列格式的 ARN：

   ```
   arn:aws:iot:<region>:<accountid>:all/things
   ```
6. 您也可以使用 ListActiveViolations (p. 629) 追蹤違規，其可讓您查看某特定安全性描述檔或目標裝置被偵測到的違規。

使用 ListViolationEvents (p. 638) 查看特定時段內偵測到哪些違規。您可以透過安全性描述檔或裝置篩選這些結果。

7. 如果您的裝置違反所定義行為的次數太過頻繁，或是不夠頻繁，您應該微調行為定義。
8. 若要檢閱已設定的安全描述檔，以及正在監控中的裝置，請使用 ListSecurityProfiles (p. 633)、ListSecurityProfilesForTarget (p. 635) 和 ListTargetsForSecurityProfile (p. 636)。

   使用 DescribeSecurityProfile (p. 625) 取得更多有關安全性描述檔的詳細資料。
9. 若要變更安全描述檔，請使用 UpdateSecurityProfile (p. 642)。使用 DetachSecurityProfile (p. 628) 來分離安全性描述檔與帳戶或目標實物群組。使用 DeleteSecurityProfile (p. 624) 來完全刪除安全性描述檔。

許可

本節包含有關如何設定管理 AWS IoT Device Defender Detect 所需 IAM 角色和政策的資訊。如需詳細資訊，請參閱 AWS Identity and Access Management User Guide (AWS 識別和存取使用者指南)。

給予 AWS IoT Device Defender Detect 許可，將提醒發佈到 SNS 主題

如果在 CreateSecurityProfile (p. 620) 中使用該 `alertTargets` 參數，您必須指定 IAM 角色使用兩個政策、許可政策和信任政策。政策授予 AWS IoT Device Defender 許可的權限，以發佈通知到您的 SNS 主題。信任政策授予 AWS IoT Device Defender 擔任所需角色的許可。

許可政策

```
{
  "Version":"2012-10-17",
  "Statement":[
      {
        "Effect":"Allow",
        "Action":[
          "sns:Publish"
        ],
        "Resource":[
          "arn:aws:sns:region:account-id:your-topic-name"
        ]
    }
  ]
}
```

信任政策

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

傳遞角色政策

您也需要一個連接到 IAM 使用者的 IAM 許可政策，其允許使用者傳遞角色。請參閱授予使用者將角色傳遞至 AWS 服務的許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": [
          "iam:GetRole",
          "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::>account-id<:role/Role_To_Pass"
    }
  ]
}
```

服務限制

- 每個目標 (實物群組或使用者帳戶) 的安全性描述檔數目上限為 5。
- 每個安全性描述檔的行為數上限為 100。
- 每個安全性描述檔的 `value` 元素 (計數、IP 地址、連接埠) 數目上限為 1000。
- 可能會將裝置指標報告調節為每個裝置每 5 分鐘報告一個指標。因此，將您的裝置報告限制為每 5 分鐘報告一個指標，以避免進行調節。
- AWS IoT Device Defender Detect 違反情況經產生後將存放 30 天。

從裝置傳送指標

AWS IoT Device Defender Detect 可收集、彙總及監控 AWS IoT 裝置產生的指標資料，以識別呈現異常行為的裝置。本節包含有關如何將指標從裝置傳送至 AWS IoT Device Defender 的資訊。

您必須在 AWS IoT 連線裝置或裝置閘道上安全地部署 AWS IoT SKD，以收集裝置端指標。AWS IoT Device Defender 提供範例代理程式，可在建立您自己的代理程式時用做範例。如果您無法提供裝置指標，仍然可以依照雲端指標取得有限的功能。

請注意以下內容：

- 目前在 C 中有提供範本裝置指標報告代理程式，網址為 https://github.com/aws-samples/aws-iot-device-defender-agent-c。在 GitHub 的 Python 中也有提供範本裝置指標報告代理程式，網址為 https://github.com/aws-samples/aws-iot-device-defender-agent-sdk-python (具體位置在此)。
- 若要使用範例代理程式或建立自己的自訂代理程式，您必須安裝 AWS IoT Device SDK。若要查看開發語言的連結，請移至 AWS IoT 核心資源。
- 所有代理程式必須建立與 AWS IoT 的連線，並將指標發佈到其中一個預留 Device Defender MQTT 的主題：

  ```
  $aws/things/THING_NAME/defender/metrics/json
  ```

  或

  ```
  $aws/things/THING_NAME/defender/metrics/cbor
  ```

  Device Defender 會使用下列其中一個主題來回覆指標報告的接收狀態：

```
$aws/things/THING_NAME/defender/metrics/json/accepted
```

```
$aws/things/THING_NAME/defender/metrics/cbor/accepted
```

```
$aws/things/THING_NAME/defender/metrics/json/rejected
```

```
$aws/things/THING_NAME/defender/metrics/cbor/rejected
```

- 為了報告指標，裝置必須註冊為 AWS IoT 中的實物。
- 裝置通常須每 5 分鐘傳送指標報告一次。裝置會調節為每 5 分鐘一個指標報告。(裝置無法進行每 5 分鐘一個以上的指標報告。)
- 裝置必須報告目前的指標。不支援歷史指標報告。
- 您可以使用 任務 (p. 363)，設定裝置指標報告代理程式執行個體的指標報告間隔。Device Defender 代理程式 C 範例隨附一個範例。如需詳細資訊，請參閱 README.md。

裝置指標文件規格

整體結構：

長名稱	短名稱	必要	類型	約束	備註
標題	hed	Y	物件		正確格式的報告需要完整的區塊。
指標	met	Y	物件		正確格式的報告需要完整的區塊。

標題區塊

長名稱	短名稱	必要	類型	約束	備註
report_id	rid	Y	整數		單調增加的值。建議使用 Epoch 時間戳記。
version	v	Y	字串	Major.Minor	新增欄位時的次要增量。移除指標後的主要增量。

指標區塊：

TCP 連線：

長名稱	短名稱	父元素	必要	類型	約束	備註
tcp_connections	tc	指標	N	物件		

長名稱	短名稱	父元素	必要	類型	約束	備註
established_connections	ec	tcp_connections	N	List<Connection>		ESTABLISHED TCP 狀態
連線	cs	established_connections	N	List<Object>		
remote_addr	rad	連線	Y	數字	ip:port	ip 可為 IPv6 或 IPv4
local_port	lp	連線	N	數字	>= 0	
local_interface	li	連線	N	字串		介面名稱
總計	t	established_connections	N	數字	>= 0	已建立連線數量

偵聽 TCP 連接埠：

長名稱	短名稱	父元素	必要	類型	約束	備註
listening_tcp_ports	lpts	指標	N	物件		
連接埠	pts	listening_tcp_ports	N	List<Port>	> 0	
port	pt	連接埠	N	數字	> 0	連接埠號碼應該 > 0
interface	如	連接埠	N	字串		介面名稱
總計	t	listening_tcp_ports	N	數字	>= 0	

偵聽 UDP 連接埠：

長名稱	短名稱	父元素	必要	類型	約束	備註
listening_udp_ports	lpts	指標	N	物件		
連接埠	pts	listening_udp_ports	N	List<Port>	> 0	
port	pt	連接埠	N	數字	> 0	連接埠號碼應該 > 0
interface	如	連接埠	N	字串		介面名稱
總計	t	listening_udp_ports	N	數字	>= 0	

網路統計：

長名稱	短名稱	父元素	必要	類型	約束	備註
network_stats	ns	指標	N	物件		
bytes_in	bi	network_stats	N	數字	差異指標，>= 0	
bytes_out	bo	network_stats	N	數字	差異指標，>= 0	

長名稱	短名稱	父元素	必要	類型	約束	備註
packets_in	pi	network_stats	N	數字	差異指標，>= 0	
packets_out	po	network_stats	N	數字	差異指標，>= 0	

使用長名稱範例 JSON 結構：

```
{
    "header": {
        "report_id": 1530304554,
        "version": "1.0"
    },
    "metrics": {
        "listening_tcp_ports": {
            "ports": [
                {
                    "interface": "eth0",
                    "port": 24800
                },
                {
                    "interface": "eth0",
                    "port": 22
                },
                {
                    "interface": "eth0",
                    "port": 53
                }
            ],
            "total": 3
        },
        "listening_udp_ports": {
            "ports": [
                {
                    "interface": "eth0",
                    "port": 5353
                },
                {
                    "interface": "eth0",
                    "port": 67
                }
            ],
            "total": 2
        },
        "network_stats": {
            "bytes_in": 29358693495,
            "bytes_out": 26485035,
            "packets_in": 10013573555,
            "packets_out": 11382615
        },
        "tcp_connections": {
            "established_connections": {
                "connections": [
                    {
                        "local_interface": "eth0",
                        "local_port": 80,
                        "remote_addr": "192.168.0.1:8000"
                    },
                    {
                        "local_interface": "eth0",
                        "local_port": 80,
```

```
                        "remote_addr": "192.168.0.1:8000"
                    }
                ],
                "total": 2
            }
        }
    }
}
```

使用短名稱範例 JSON 結構：

```
{
    "hed": {
        "rid": 1530305228,
        "v": "1.0"
    },
    "met": {
        "tp": {
            "pts": [
                {
                    "if": "eth0",
                    "pt": 24800
                },
                {
                    "if": "eth0",
                    "pt": 22
                },
                {
                    "if": "eth0",
                    "pt": 53
                }
            ],
            "t": 3
        },
        "up": {
            "pts": [
                {
                    "if": "eth0",
                    "pt": 5353
                },
                {
                    "if": "eth0",
                    "pt": 67
                }
            ],
            "t": 2
        },
        "ns": {
            "bi": 29359307173,
            "bo": 26490711,
            "pi": 10014614051,
            "po": 11387620
        },
        "tc": {
            "ec" : {
                "cs": [
                    {
                        "li": "eth0",
                        "lp": 80,
                        "rad": "192.168.0.1:8000"
                    },
                    {
                        "li": "eth0",
                        "lp": 80,
                        "rad": "192.168.0.1:8000"
```

```
                    }
                ],
                "t": 2
            }
        }
    }
}
```

偵測命令

AttachSecurityProfile

將 AWS IoT Device Defender 安全性描述檔與下列其中一個目標類型產生關聯：

- 所有裝置
- 所有已註冊的裝置 (AWS IoT 登錄中的實物)
- 所有未註冊的裝置
- 實物群組中的裝置

每個目標類型最多可有 5 個安全性描述檔與其相關聯。

概要：

```
aws iot  attach-security-profile \
    --security-profile-name <value> \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "securityProfileTargetArn": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	連接的安全性描述檔。
securityProfileTargetArn	字串	安全性描述檔連接的目標 (實物群組) 的 ARN。

若要將安全性描述檔附加到一組裝置，您必須指定包含該裝置的實物群組的 ARN。此實物群組 ARN 的格式如下：

```
arn:aws:iot:<region>:<accountid>:thinggroup/<thing-group-name>
```

若要將安全性描述檔附加到帳戶中所有已註冊的實物 (忽略未註冊的實物)，您必須指定具下列格式的 ARN：

```
arn:aws:iot:<region>:<accountid>:all/registered-things
```

若要將安全性描述檔附加到所有未註冊的實物，您必須指定具下列格式的 ARN：

```
arn:aws:iot:<region>:<accountid>:all/unregistered-things
```

若要將安全性描述檔附加到所有裝置，您必須指定具下列格式的 ARN：

```
arn:aws:iot:<region>:<accountid>:all/things
```

輸出：

無

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`LimitExceededException`

已超出限制。

`VersionConflictException`

當傳送至命令的實物版本與 `--version` 參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CreateSecurityProfile

建立 Device Defender 安全性描述檔。

概要：

```
aws iot  create-security-profile \
    --security-profile-name <value> \
    [--security-profile-description <value>] \
    [--behaviors <value>] \
    [--alert-targets <value>] \
    [--additional-metrics-to-retain <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "securityProfileDescription": "string",
```

```
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

cli-input-json 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供的安全性描述檔的名稱。
securityProfileDescription	字串 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的說明。
行為	清單 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	字串 長度上限：128；下限：1	您已指定給行為的名稱。

名稱	類型	敘述
	模式：[a-zA-Z0-9:_-]+	
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。

名稱	類型	敘述
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。
alertTargets	對應	指定傳送提醒的目的地。(提醒始終傳送到主控台)。 當裝置 (實物) 違反行為時產生提醒。
alertTargetArn	字串	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	清單 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 `behaviors` 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理安全性描述檔的中繼資料。
金鑰	字串	標籤的金鑰。
數值	字串	標籤的值。

輸出：

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string"
}
```

CLI 輸出欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供的安全性描述檔的名稱。

名稱	類型	敘述
securityProfileArn	字串	安全性描述檔的 ARN。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteSecurityProfile

刪除 Device Defender 安全性描述檔。

概要：

```
aws iot  delete-security-profile \
    --security-profile-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	要刪除的安全性描述檔的名稱。
expectedVersion	long	安全性描述檔的預期版本。只要更新安全性描述檔便產生新版本。如果指定的值與實際版本不同，便擲出 `VersionConflictException`。

輸出：

無

錯誤：

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`VersionConflictException`

當傳送至命令的實物版本與 `--version` 參數所指定的不符，將擲出此例外狀況。

DescribeSecurityProfile

取得有關 Device Defender 的安全性描述檔的資訊。

概要：

```
aws iot  describe-security-profile \
    --security-profile-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您需要取得其資訊的安全性描述檔的名稱。

輸出：

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
```

```
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "version": "long",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	安全性描述檔的名稱。
securityProfileArn	字串	安全性描述檔的 ARN。
securityProfileDescription	字串 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的描述 (與安全性描述檔建立或更新時間相關聯)。
行為	清單 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。

名稱	類型	敘述
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。

名稱	類型	敘述
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為。否則就是違反行為。
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。
alertTargetArn	字串	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	清單 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 `behaviors` 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
version	long	安全性描述檔的版本。只要更新安全性描述檔便產生新版本。
creationDate	timestamp	安全性描述檔的建立時間。
lastModifiedDate	timestamp	安全性描述檔的最後建立時間。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DetachSecurityProfile

將 Device Defender 安全性描述檔與實物群組或此帳戶的關聯取消。

概要：

```
aws iot  detach-security-profile \
    --security-profile-name <value> \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "securityProfileTargetArn": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	分離的安全性描述檔。
securityProfileTargetArn	字串	安全性描述檔與之分離的實物群組的 ARN。

輸出：

無

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListActiveViolations

列出指定之 Device Defender 安全性描述檔的作用中違規。

概要：

```
aws iot  list-active-violations \
    [--thing-name <value>] \
    [--security-profile-name <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
```

```
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "thingName": "string",
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位：

名稱	類型	描述
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	已列出作用中違規的實物的名稱。
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	已列出違規之 Device Defender 安全性描述檔的名稱。
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出：

```
{
  "activeViolations": [
    {
      "violationId": "string",
      "thingName": "string",
      "securityProfileName": "string",
      "behavior": {
        "name": "string",
        "metric": "string",
        "criteria": {
          "comparisonOperator": "string",
          "value": {
            "count": "long",
            "cidrs": [
              "string"
            ],
            "ports": [
              "integer"
            ]
          },
          "durationSeconds": "integer",
          "consecutiveDatapointsToAlarm": "integer",
          "consecutiveDatapointsToClear": "integer",
          "statisticalThreshold": {
            "statistic": "string"
```

```
          }
        }
      },
      "lastViolationValue": {
        "count": "long",
        "cidrs": [
          "string"
        ],
        "ports": [
          "integer"
        ]
      },
      "lastViolationTime": "timestamp",
      "violationStartTime": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	敘述
activeViolations	清單 成員：ActiveViolation	作用中違規的清單。
violationId	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	作用中違規的 ID。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	負責作用中違規的實物的名稱。
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	行為違規的安全性描述檔。
行為	Behavior	違反的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。

名稱	類型	敘述
		列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。

名稱	類型	敘述
lastViolationValue	MetricValue	造成最新違規的指標值 (測量)。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
lastViolationTime	timestamp	最新違規的發生時間。
violationStartTime	timestamp	違規開始的時間。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListSecurityProfiles

列出您已建立的 Device Defender 安全性描述檔。您可以使用篩選功能，只列出僅與實物群組，或僅與您的帳戶關聯的這些安全性描述檔。

概要：

```
aws iot  list-security-profiles \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "nextToken": "string",
```

```
  "maxResults": "integer"
}
```

cli-input-json 欄位：

名稱	類型	說明
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出：

```
{
  "securityProfileIdentifiers": [
    {
      "name": "string",
      "arn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	敘述
securityProfileIdentifiers	清單 成員：SecurityProfileIdentifier Java 類別：java.util.List	安全性描述檔識別符的清單 (名稱和 ARN)。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供給安全性描述檔的名稱。
arn	字串	安全性描述檔的 ARN。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListSecurityProfilesForTarget

列出連接目標 (實物群組) 的 Device Defender 安全性描述檔。

概要：

```
aws iot  list-security-profiles-for-target \
    [--next-token <value>] \
    [--max-results <value>] \
    [--recursive | --no-recursive] \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "recursive": "boolean",
  "securityProfileTargetArn": "string"
}
```

`cli-input-json` 欄位：

名稱	類型	說明
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
recursive	布林值	若此值為 true，亦將回傳子群組。
securityProfileTargetArn	字串	您要取得其連接之安全性描述檔的目標 (實物群組) 的 ARN。

輸出：

```
{
  "securityProfileTargetMappings": [
    {
      "securityProfileIdentifier": {
        "name": "string",
        "arn": "string"
      },
      "target": {
        "arn": "string"
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	敘述
securityProfileTargetMappings	清單 成員：SecurityProfileTargetMapping Java 類別：java.util.List	安全性描述檔及其關聯目標的清單。
securityProfileIdentifier	SecurityProfileIdentifier	識別安全性描述檔的資訊。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供給安全性描述檔的名稱。
arn	字串	安全性描述檔的 ARN。
target	SecurityProfileTarget	有關與安全設定檔相關聯之目標 (實物群組) 的資訊。
arn	字串	安全性描述檔的 ARN。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListTargetsForSecurityProfile

列出與指定 Device Defender 安全性描述檔相關聯的目標 (實物群組)。

概要：

```
aws iot  list-targets-for-security-profile \
    --security-profile-name <value> \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	安全性描述檔。
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出：

```
{
  "securityProfileTargets": [
    {
      "arn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	描述
`securityProfileTargets`	清單 成員：SecurityProfileTarget Java 類別：java.util.List	安全性描述檔連接的實物群組。
`arn`	字串	安全性描述檔的 ARN。
`nextToken`	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListViolationEvents

列出在特定期間內發現之 Device Defender 安全性描述檔違規。您可以針對特別的安全性描述檔、行為或實物 (裝置)，使用篩選功能來限制發出這些提醒的結果。

概要：

```
aws iot  list-violation-events \
    --start-time <value> \
    --end-time <value> \
    [--thing-name <value>] \
    [--security-profile-name <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "startTime": "timestamp",
  "endTime": "timestamp",
  "thingName": "string",
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位：

名稱	類型	敘述
startTime	timestamp	列出之提醒的開始時間。
endTime	timestamp	列出之提醒的結束時間。
thingName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	一個篩選功能，其能限制指定之實物所引起的提醒的結果。
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	一個篩選功能，其能限制指定之安全性描述檔所產生的提醒的結果。
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出：

```
{
  "violationEvents": [
    {
      "violationId": "string",
      "thingName": "string",
      "securityProfileName": "string",
      "behavior": {
        "name": "string",
        "metric": "string",
        "criteria": {
          "comparisonOperator": "string",
          "value": {
            "count": "long",
            "cidrs": [
              "string"
            ],
            "ports": [
              "integer"
            ]
          },
          "durationSeconds": "integer",
          "consecutiveDatapointsToAlarm": "integer",
          "consecutiveDatapointsToClear": "integer",
          "statisticalThreshold": {
            "statistic": "string"
          }
        }
      },
      "metricValue": {
        "count": "long",
        "cidrs": [
          "string"
        ],
        "ports": [
          "integer"
        ]
      },
      "violationEventType": "string",
      "violationEventTime": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位：

名稱	類型	敘述
violationEvents	清單 成員：ViolationEvent	在指定時段內針對此帳戶發出的安全性描述檔違規提醒，可能由安全性描述檔、違規的行為或實物 (裝置) 違規所篩選出來。
violationId	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	違規事件的 ID。
thingName	字串	負責違規事件的實物的名稱。

名稱	類型	敘述
	長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	行為違規之安全性描述檔的名稱。
行為	Behavior	所違反的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。

名稱	類型	敘述
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。
metricValue	MetricValue	指標 (測量) 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
violationEventType	字串	違規事件的類型。 列舉：in-alarm | alarm-cleared | alarm-invalidated

名稱	類型	敘述
violationEventTime	timestamp	發生違規事件的時間。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

UpdateSecurityProfile

更新 Device Defender 安全性描述檔。

概要：

```
aws iot  update-security-profile \
    --security-profile-name <value> \
    [--security-profile-description <value>] \
    [--behaviors <value>] \
    [--alert-targets <value>] \
    [--additional-metrics-to-retain <value>] \
    [--delete-behaviors | --no-delete-behaviors] \
    [--delete-alert-targets | --no-delete-alert-targets] \
    [--delete-additional-metrics-to-retain | --no-delete-additional-metrics-to-retain] \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
  "securityProfileName": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
```

```
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "deleteBehaviors": "boolean",
  "deleteAlertTargets": "boolean",
  "deleteAdditionalMetricsToRetain": "boolean",
  "expectedVersion": "long"
}
```

cli-input-json 欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您要更新之安全性描述檔的名稱。
securityProfileDescription	字串 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的說明。
行為	清單 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-

名稱	類型	敘述
		cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。

名稱	類型	敘述
alertTargetArn	字串	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	清單 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 `behaviors` 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
deleteBehaviors	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 `behaviors`。如果在目前叫用中定義任何 `behaviors`，則會發生例外狀況。
deleteAlertTargets	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 `alertTargets`。如果在目前叫用中定義任何 `alertTargets`，則會發生例外狀況。
deleteAdditionalMetricsToRetain	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 `additionalMetricsToRetain`。如果在目前叫用中定義任何 `additionalMetricsToRetain`，則會發生例外狀況。
expectedVersion	long	安全性描述檔的預期版本。只要更新安全性描述檔便產生新版本。如果指定的值與實際版本不同，便擲出 `VersionConflictException`。

輸出：

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
```

```
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "version": "long",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位：

名稱	類型	敘述
securityProfileName	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	已更新之安全性描述檔的名稱。
securityProfileArn	字串	已更新之安全性描述檔的 ARN。
securityProfileDescription	字串 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的描述。
行為	清單 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。

名稱	類型	敘述
		列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為。否則就是違反行為。

名稱	類型	敘述
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。
alertTargetArn	字串	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	清單 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在安全性描述檔 `behaviors` 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
version	long	安全性描述檔的更新版本。
creationDate	timestamp	安全性描述檔的建立時間。
lastModifiedDate	timestamp	安全性描述檔的最後建立時間。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`VersionConflictException`

當傳送至命令的實物版本與 `--version` 參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ValidateSecurityProfileBehaviors

驗證 Device Defender 安全性描述檔行為規格。

概要：

```
aws iot  validate-security-profile-behaviors \
    --behaviors <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式：

```
{
```

```
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ]
}
```

cli-input-json 欄位：

名稱	類型	敘述
行為	清單 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	字串 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
指標	字串	行為測量的內容。
條件	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
comparisonOperator	字串	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
值	MetricValue	相較於 `metric` 的值。
計數	long 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。

名稱	類型	敘述
cidrs	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
連接埠	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	字串 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為。否則就是違反行為。

輸出：

```
{
  "valid": "boolean",
  "validationErrors": [
    {
      "errorMessage": "string"
    }
  ]
```

```
}
```

CLI 輸出欄位：

名稱	類型	描述
`valid`	布林值	行為若有效則為 True。
`validationErrors`	清單 成員：ValidationError	行為中找到的任何錯誤清單。
`errorMessage`	字串 長度上限：2048	行為中找到對錯誤的描述。

錯誤：

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

裝置代理程式與 AWS IoT Greengrass 整合

AWS IoT Device Defender 可以與 AWS IoT Greengrass 結合使用。裝置代理程式整合遵循標準式 AWS IoT Greengrass Lambda 函數的部署模型，讓您能夠將 AWS IoT Device Defender 安全性加入到 AWS IoT Greengrass 核心裝置中。若要整合裝置代理程式，請遵循本節中概述的步驟。

事前準備：

- 設定您的 AWS IoT Greengrass 環境。
- 設定和執行您的 AWS IoT Greengrass 核心。
- 確保您可以在 AWS IoT Greengrass 核心成功部署和執行您的 Lambda 函數。

一般而言，這裡所述的程序乃遵循 AWS IoT Greengrass 開發人員指南中的建立和封裝 Lambda 函數一節。

建立 Lambda 套件

1. 複製 AWS IoT Device Defender Python 範例儲存庫。

```
git clone https://github.com/aws-samples/aws-iot-device-defender-agent-sdk-python.git
```

2. 建立和啟動虛擬環境 (建議選用)。

```
pip install virtualenv
virtualenv metrics_lambda_environment
source metrics_lambda_environment/bin/activate
```

3. 在虛擬環境中安裝 AWS IoT Device Defender 範例代理程式。從 PyPi 安裝。

```
pip install AWSIoTDeviceDefenderAgentSDK
```

4. 安裝已下載的來源。

```
cd aws-iot-device-defender-agent-sdk-python
#This must be run from the same directory as setup.py
pip install .
```

5. 建立空的目錄來集合您的 Lambda 函數。這是您的 Lambda 目錄。

```
mkdir metrics_lambda
cd metrics_lambda
```

6. 完成建立和套件 Lambda 函數中的步驟 1-4。
7. 將 AWS IoT Greengrass Python SDK 解壓縮到您的 Lambda 目錄。

```
unzip ../aws_greengrass_core_sdk/sdk/python_sdk_1_1_0.zip
cp -R ../aws_greengrass_core_sdk/examples/HelloWorld/greengrass_common .
cp -R ../aws_greengrass_core_sdk/examples/HelloWorld/greengrasssdk .
cp -R ../aws_greengrass_core_sdk/examples/HelloWorld/greengrass_ipc_python_sdk .
```

8. 將 AWSIoTDeviceDefenderAgentSDK 模組複製到 Lambda 目錄的根層級。

```
cp -R ../aws-iot-device-defender-agent-sdk-python/AWSIoTDeviceDefenderAgentSDK .
```

9. 將 AWS IoT Greengrass 代理程式複製到 Lambda 目錄的根層級。

```
cp ../aws-iot-device-defender-agent-sdk-python/samples/greengrass/
greengrass_core_metrics_agent/greengrass_defender_agent.py .
```

10. 自訂 AWS IoT Greengrass 代理程式以包含 AWS IoT Greengrass 核心裝置的名稱，以及所需的指標採樣率：
    - 以您的 AWS IoT Greengrass 核心名稱取代 `GREENGRASS_CORENAME`。
    - 將 `SAMPLE_RATE_SECONDS` 設定為所需的指標報告間隔。AWS IoT Device Defender 支援的最短報告間隔是 5 分鐘 (300 秒)。
11. 將您的虛擬環境 (或您的系統) 的相依性複製到 Lambda 目錄的根層級。

```
cp -R ../metrics_lambda_environment/lib/python2.7/site-packages/psutil .
cp -R ../metrics_lambda_environment/lib/python2.7/site-packages/cbor .
```

12. 建立您的 Lambda 函數 Zip 檔。在 Lambda 目錄的根層級執行此命令。

```
rm *.zip
zip -r greengrass_defender_metrics_lambda.zip *
```

設定和部署 AWS IoT Greengrass Lambda 函數

1. 上傳您的 Lambda zip 檔。
2. 選取 Python 2.7 執行時間，並在 Handler 欄位中輸入 `greengrass_defender_agent.function_handler`。
3. 將您的 Lambda 函數設定為長效 Lambda 函數。
4. 將您的 Lambda 函數的訂閱設定為 AWS IoT 雲端。對於 AWS IoT Device Defender，不需要從 AWS IoT 雲端到您 Lambda 函數的訂閱。

5. 建立本機資源，讓您的 Lambda 函數從 AWS IoT Greengrass 核心主機收集指標：
   - 按照使用 Lambda 函數存取本機資源中的指示操作。使用下列參數：
     - 資源名稱：`Core Proc`
     - 類型：`Volume`
     - 來源路徑：`/proc`
     - 目的地路徑：`/host_proc`
     - 群組擁有者檔案存取許可：為擁有資源的 Linux 群組自動新增作業系統群組許可
     - 將資源與您的指標 Lambda 函數關聯在一起。
6. 將 Lambda 函數部署至您的 AWS IoT Greengrass 群組。

使用 AWS IoT 主控台檢閱您的 AWS IoT Device Defender 裝置指標

1. 將 AWS IoT Greengrass Lambda 函數的發佈主題暫時修改為「指標/測試」。
2. 部署 Lambda 函數。
3. 若要查看您的 AWS IoT Greengrass 核心所發出的指標，請在 AWS IoT 主控台的 Test (測試) 頁面上，新增對臨時主題 (「指標/測試」) 的訂閱。

裝置代理程式的安全性最佳實務

最低權限

代理程式程序應該只能獲得履行其職責所需的最低許可。

基本機制

- 代理程式應以非根使用者身分來執行。
- 代理程式應以專用使用者之方式在其自己之群組執行。
- 使用者/群組應獲得所需資源的唯讀許可，以收集和傳輸指標。
- 範例：範例代理程式的 /proc/sys 唯讀。
- 如需有關如何將程序設定為以降低許可執行的範例，請參閱 Python 範例代理程式隨附的設定說明。

有多種知名 Linux 機制，可協助您進一步限制或隔離您的代理程式處理序：

進階機制

- CGroups
- SELinux
- Chroot
- Linux 命名空間

運作彈性

代理程式處理序必須能迅速從意外運作錯誤和例外狀況中復原，不得當機或永久結束。程式碼需要從容地處理例外狀況，並且必須設定為在發生意外終止時自動重新啟動 (例如，由於系統重新啟動或未擷取到的例外狀況) 做為預防措施。

最少相依性

代理程式在其實作中必須使用最少的相依性 (也就是第三方程式庫)。如果因為任務的複雜性 (例如，Transport Layer Security) 而使程式庫的使用成為正當，只能使用維護良好的相依性並建立機制，以

保持最新的相依性。如果新增的相依性包含代理程式未使用的功能，且預設為作用中 (例如，開放連接埠網域通訊端)，請在您的程式碼中停用，或透過程式庫的組態檔案停用。

程序隔離

代理程式處理序必須只包含執行裝置指標收集和傳輸所需的功能。它不得裝載在其他系統處理序做為容器，或為其他範圍外的使用案例實作功能。此外，代理程式處理序必須避免建立傳入通訊管道，例如網域通訊端和網路服務連接埠，允許本機或遠端程序干擾其操作，並影響其完整性和孤立性。

隱匿性

代理程式處理序不得以關鍵字命名，例如指出用途和安全價值的 security (安全性)、monitoring (監控) 或 audit (稽核)。偏好使用一般程式碼名稱或隨機和每個裝置獨有的處理序名稱。為代理程式二進位程式碼所在目錄，以及處理序引數的任何名稱和值命名時，必須遵循相同的原則。

最少共用資訊

任何部署到裝置的代理程式成品，不得包含敏感資訊，如特許的登入資料、偵錯和無效的程式碼，或是內嵌評論或文件檔案，其揭露有關伺服器端處理代理程式收集指標的細節，或其他後端系統的詳細資訊。

Transport Layer Security

若要建立 TLS 安全管道來處理資料傳輸，代理程式處理序必須強制執行所有用戶端的驗證，例如憑證鏈和網域名稱在應用程式層級驗證，如果沒有預設啟用的話。此外，代理程式必須使用根憑證存放區，其中包含信任憑證授權單位，但不包含屬於被入侵的憑證發行者的憑證。

安全部署

任何代理程式的部署機制，例如程式碼推送或同步，及包含二進位程式碼的儲存庫、原始碼及任何組態檔案 (包括信任的根憑證)，必須控制存取權以防止未經授權的程式碼插入或竄改。如果部署機制倚賴網路通訊，則使用加密方法來保護傳輸之部署成品的完整性。

深入閱讀

- Security and Identity for AWS IoT
- Understanding the AWS IoT Security Model
- Redhat: A Bite of Python
- 10 common security gotchas in Python and how to avoid them
- What Is Least Privilege & Why Do You Need It?
- OWASP Embedded Security Top 10
- OWASP IoT Project

AWS IoT Device Defender 故障診斷指南

一般

問：使用 AWS IoT Device Defender 有無任何事前準備？

答：如果您想要使用裝置報告的指標，則必須先在 AWS IoT 連線裝置或裝置閘道上部署代理程式。裝置必須提供一致的用戶端識別符或實物名稱。

稽核

問：我已啟用檢查，我的稽核顯示「進行中」一段時間了。有問題嗎？ 何時能得到結果？

答：啟用檢查後，就立即開始收集資料。不過，如果您的帳戶中有大量資料需要收集 (憑證、實物、政策等)，那麼在您啟用檢查後，該檢查結果可能會有一段時間無法使用。

偵測

問：如何知道要在 AWS IoT Device Defender 安全性描述檔行為中設定哪些閾值？

答：首先以低閾值建立安全性描述檔行為，並將它連接到包含一組代表性裝置的實物群組。您可以使用 AWS IoT Device Defender 來檢視目前的指標，然後微調裝置行為閾值，以符合您的使用案例。

問：我建立行為，但未如預期觸發違規。應如何修正這個問題？

答：當您定義行為時，您會指定您希望裝置正常運作的方式。例如，如果您有一個安全相機僅連接到 TCP 連接埠 8888 的一個中央伺服器，您未預期其他任何連線。若要相機在其他連接埠連線時獲得提醒，您可以定義的行為如下：

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 8888 ]
    }
  }
}
```

如果相機在 TCP 連接埠 443 進行 TCP 連線，則裝置行為會違規並且觸發提醒。

問：我的一或多個行為違規。如何清除違規？

答：如行為描述檔所定義，警示會在裝置回到預期的行為後清除。在收到您裝置的指標資料時，會進行行為描述檔的評估。

問：我已刪除違規的行為，但是要如何停止提醒？

答：刪除行為會停止所有該行為未來的違規和提醒。事前提醒必須從您的通知機制排除。當您刪除行為時，該行為的違規記錄保留在您帳戶中的時間週期如同所有其他違規。

裝置指標

問：我提交指標報告，我知道違反我的行為但未觸發違規。怎麼回事？

答：透過訂閱以下 MQTT 主題確認是否正在受理您的指標報告：

```
$aws/things/THING_NAME/defender/metrics/FORMAT/rejected
$aws/things/THING_NAME/defender/metrics/FORMAT/accepted
```

其中 `THING_NAME` 是報告指標的實物名稱，`FORMAT` 為「json」或「cbor」，取決於實物所提交的指標報告格式。

在訂閱息之後，您應該針對每個提交的指標報告收到這些主題的訊息。`rejected` 訊息表示剖析指標報告時發生問題。錯誤訊息會包含在訊息承載內，以協助您修正指標報告中的任何錯誤。`accepted` 訊息，指示指標報告已正確剖析。

問：如果我在指標報告中傳送空的指標會怎麼樣？

答：空的連接埠或 IP 地址清單一律視為符合對應的行為。如果對應的行為是違規，則會清除違規。

問：為何我的裝置指標報告包含不在 AWS IoT 登錄中的裝置訊息？

如果您有一或多個安全性描述檔附加到所有實物或所有未註冊的實物，則 AWS IoT Device Defender 包含來自未註冊實物的指標。如果您想要排除來自未註冊實物的指標，您可以將描述檔附加到所有已註冊的裝置，而不是所有裝置。

問：即使我將安全性描述檔套用到所有未註冊的裝置或所有裝置，還是沒看到來自一或多個未註冊裝置的訊息。可以如何修正這個問題？

請確認您傳送的是格式正確的指標報告，且使用其中一個支援的格式。如需相關資訊，請參閱裝置指標文件規格 (p. 615)。確認未註冊的裝置使用一致的用戶端識別符或實物名稱。如果實物名稱包含控制字元，或無誤名稱超過 128 個位元組的 UTF-8 編碼字元，則會拒絕裝置所回報的訊息。

問：如果未註冊的裝置新增到登錄，或已註冊的裝置變成未註冊，會發生什麼狀況？

答：如果在登錄中新增或移除裝置：

- 如果持續發佈違規指標，您會看到裝置有兩項不同的違規 (一項針對其已註冊的實物名稱，一項針對其未註冊的身分)。舊身分的有效違規會在兩天後停止出現，但會出現在違規歷史記錄中長達 14 天。

問：我應該在我的裝置指標報告的報告 ID 欄位中提供哪個值？

答：使用每個指標報告的唯一值，以正整數表示。常見的做法是使用 Unix epoch 時間戳記。

問：我應該為 AWS IoT Device Defender 指標建立專用的 MQTT 連線嗎？

答：通常不需要獨立的 MQTT 連線。

問：連接時我應該使用哪個用戶端 ID 來發佈裝置指標？

對於 AWS IoT 登錄中的裝置 (實物)，請使用已註冊的實物名稱。對於不在 AWS IoT 登錄中的裝置，請在連接到 AWS IoT 時使用一致的識別符。此做法有助於比對違規與實物名稱。

問：是否可以發佈具不同用戶端 ID 的裝置指標？

您可以代表其他實物發佈指標。只要將指標發佈到該裝置的 AWS IoT Device Defender 預留主題，即可完成。例如，`Thing-1` 會自行發佈指標，也可代表 `Thing-2`。`Thing-1` 收集自己的指標，並且在 MQTT 主題上發佈這些指標：

```
$aws/things/Thing-1/defender/metrics/json
```

`Thing-1` 會接著從 `Thing-2` 取得指標，並且在 MQTT 主題上發佈這些指標：

```
$aws/things/Thing-2/defender/metrics/json
```

問：我在我的帳戶中可以擁有多少個安全性描述檔和行為？

答：請參閱 服務限制 (p. 614)。

問：適用於提醒目標的原型目標角色是什麼樣子？

答：可讓 AWS IoT Device Defender 對提醒目標 (SNS 主題) 發佈提醒的角色需要 2 項條件：

- 可將 iot.amazonaws.com 指定為信任實體的信任關係。
- 附加的政策，授予 AWS IoT 在指定的 SNS 主題上發佈的許可。例如：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sns:Publish",
            "Resource": "<sns-topic-arn>"
        }
    ]
}
```

事件訊息

> **Note**
>
> 此節包含了有關事物或任務更新或變更時由 AWS IoT 所發佈之資訊。有關讓您建立偵測器，以在操作中監控裝置故障或變更情況，和在這種情況發生時觸發動作的 AWS IoT Events 服務相關資訊，請參閱 AWS IoT Events。

當發生特定事件，AWS IoT 就會發佈事件訊息。例如，在新增、更新或刪除事物時，由登錄檔產生的事件。每個事件都會觸發傳送一則事件訊息。事件訊息會透過具 JSON 承載的 MQTT 發佈。承載內容取決於事件的類型。

> **Note**
>
> 事件訊息保證會發佈一次。事件訊息也有可能發佈超過一次。事件訊息的順序無法保證。

若要接收事件訊息，您的裝置必須使用適當的政策，以允許該裝置連接至 AWS IoT 裝置閘道，並訂閱 MQTT 事件主題。您也必須訂閱合適的主題篩選條件。

以下為接收生命週期事件所需的政策範例：

```
{
    "Version":"2012-10-17",
    "Statement":[{
        "Effect":"Allow",
        "Action":[
            "iot:Subscribe",
            "iot:Receive"
        ],
        "Resource":[
            "arn:aws:iot:region:account:/$aws/events/*"
        ]
    }]
}
```

您可透過呼叫 UpdateEventConfigurations API 或使用 update-event-configurations CLI 指令來控制要發佈的事件類型。例如：

aws iot update-event-configurations --event-configurations "{\"THING\":{\"Enabled\": true}}"

> **Note**
>
> 所有引號 (") 都會與反斜線 (\) 一起逸出。

您可以透過呼叫 DescribeEventConfigurations API 或使用 describe-event-configurations CLI 指令來取得目前的事件設定。例如：

aws iot describe-event-configurations

describe-event-configurations 命令的輸出如下所示：

```
{
    "lastModifiedDate": 1552671347.841,
    "eventConfigurations": {
        "THING_TYPE": {
            "Enabled": false
        },
        "JOB_EXECUTION": {
```

```
            "Enabled": false
        },
        "THING_GROUP_HIERARCHY": {
            "Enabled": false
        },
        "CERTIFICATE": {
            "Enabled": false
        },
        "THING_TYPE_ASSOCIATION": {
            "Enabled": false
        },
        "THING_GROUP_MEMBERSHIP": {
            "Enabled": false
        },
        "CA_CERTIFICATE": {
            "Enabled": false
        },
        "THING": {
            "Enabled": true
        },
        "JOB": {
            "Enabled": false
        },
        "POLICY": {
            "Enabled": false
        },
        "THING_GROUP": {
            "Enabled": false
        }
    },
    "creationDate": 1552671347.84
}
```

登錄檔事件

該登錄檔會在建立、更新或刪除事物、事物類型和事物群組時發佈事件訊息。該登錄檔目前支援以下事件類型：

建立/更新/刪除物件

該登錄檔會在建立、更新或刪除事物時發佈以下事件訊息：

- $aws/events/thing/<thingName>/created
- $aws/events/thing/<thingName>/updated
- $aws/events/thing/<thingName>/deleted

訊息包含的承載範例如下：

```
{
    "eventType" : "THING_EVENT",
    "eventId" : "f5ae9b94-8b8e-4d8e-8c8f-b3266dd89853",
    "timestamp" : 1234567890123,
    "operation" : "CREATED|UPDATED|DELETED",
    "accountId" : "123456789012",
    "thingId" : "b604f69c-aa9a-4d4a-829e-c480e958a0b5",
    "thingName" : "MyThing",
    "versionNumber" : 1,
    "thingTypeName" : null,
    "attributes": {
                "attribute3": "value3",
                "attribute1": "value1",
```

```
                "attribute2": "value2"
    }
}
```

承載包含以下屬性：

eventType

設定為「THING_EVENT」。

eventId

獨特的事件 ID (字串)。

timestamp

事件發生時的 UNIX 時間戳記。

operation

觸發事件的操作。有效值為：

- CREATED (已建立)
- UPDATED (已更新)
- DELETED (已刪除)

accountId

您的 AWS 帳戶 ID。

thingId

建立、更新或刪除的物件之 ID。

thingName

建立、更新或刪除的物件之名稱。

versionNumber

建立、更新或刪除的物件之版本。物件建立時，此值設定為 1。物件更新時，此值會增加 1。

thingTypeName

與物件關聯的物件類型 (若存在)。否則即為 `null`。

attributes

與物件關聯的名稱/值對之集合。

建立/棄用/取消棄用/刪除物件類型

該登錄檔會在建立、棄用、取消棄用或刪除事物類型時發佈以下事件訊息：

- `$aws/events/thingType/<thingTypeName>/created`
- `$aws/events/thingType/<thingTypeName>/updated`
- `$aws/events/thingType/<thingTypeName>/deleted`

訊息包含的承載範例如下：

```
{
    "eventType" : "THING_TYPE_EVENT",
    "eventId" : "8827376c-4b05-49a3-9b3b-733729df7ed5",
    "timestamp" : 1234567890123,
    "operation" : "CREATED|UPDATED|DELETED",
    "accountId" : "123456789012",
    "thingTypeId" : "c530ae83-32aa-4592-94d3-da29879d1aac",
    "thingTypeName" : "MyThingType",
    "isDeprecated" : false|true,
```

```
    "deprecationDate" : null,
    "searchableAttributes" : [ "attribute1", "attribute2", "attribute3" ],
    "description" : "My thing type"
}
```

承載包含以下屬性：

eventType

設定為「THING_TYPE_EVENT」。

eventId

獨特的事件 ID (字串)。

timestamp

事件發生時的 UNIX 時間戳記。

operation

觸發事件的操作。有效值為：

- CREATED (已建立)
- UPDATED (已更新)
- DELETED (已刪除)

accountId

您的 AWS 帳戶 ID。

thingTypeId

建立、棄用或刪除的物件類型之 ID。

thingTypeName

建立、棄用或刪除的物件類型之名稱。

isDeprecated

`true`指定物件類型已棄用。否則即為 `false`。

deprecationDate

棄用物件類型的 UNIX 時間戳記。

searchableAttributes

與可用於搜尋的物件類型關聯之名稱/值對集合。

說明

物件類型描述。

關聯事物類型或取消關聯事物

當事物類型與事物關聯或取消關聯時，登錄檔會發佈以下事件訊息。

- `$aws/events/thingTypeAssociation/thing/`*`<thingName>`*`/`*`<typeName>`*

訊息包含的承載範例如下：

```
{
    "eventId" : "87f8e095-531c-47b3-aab5-5171364d138d",
    "eventType" : "THING_TYPE_ASSOCIATION_EVENT",
    "operation" : "CREATED|DELETED",
    "thingId" : "b604f69c-aa9a-4d4a-829e-c480e958a0b5",
    "thingName": "myThing",
    "thingTypeName" : "MyThingType",
```

```
    "timestamp" : 1234567890123,
}
```

承載包含以下屬性：

eventId

獨特的事件 ID (字串)。

eventType

設定為 "THING_TYPE_ASSOCIATION_EVENT"。

operation

觸發事件的操作。有效值為：

- CREATED (已建立)
- DELETED (已刪除)

thingId

類型關聯已變更的物件 ID。

thingName

類型關聯已變更的物件名稱。

thingTypeName

與物件相關聯或不再相關聯的物件類型。

timestamp

事件發生時的 UNIX 時間戳記。

建立/更新/刪除物件群組

登錄檔會在建立、更新或刪除事物群組時發佈以下事件訊息。

- `$aws/events/thingGroup/<groupName>/created`
- `$aws/events/thingGroup/<groupName>/updated`
- `$aws/events/thingGroup/<groupName>/deleted`

訊息包含的承載範例如下：

```
{
    "eventType" : "THING_GROUP_EVENT",
    "eventId" : "87f8e095-531c-47b3-aab5-5171364d138d",
    "timestamp" : 1234567890123,
    "operation" : "CREATED|UPDATED|DELETED",
    "accountId" : "123456789012",
    "thingGroupId" : "8f82a106-6b1d-4331-8984-a84db5f6f8cb",
    "thingGroupName" : "MyRootThingGroup",
    "versionNumber" : 1,
    "parentGroupName" : null,
    "parentGroupId" : null,
    "description" : "My root thing group",
    "rootToParentThingGroups" : null,
    "attributes" : {
        "attribute1" : "value1",
        "attribute3" : "value3",
        "attribute2" : "value2"
    }
}
```

承載包含以下屬性：

eventType

設定為「THING_GROUP_EVENT」。

eventId

獨特的事件 ID (字串)。

timestamp

事件發生時的 UNIX 時間戳記。

operation

觸發事件的操作。有效值為：

- CREATED (已建立)
- UPDATED (已更新)
- DELETED (已刪除)

accountId

您的 AWS 帳戶 ID。

thingGroupId

建立、更新或刪除的物件群組之 ID。

thingGroupName

建立、更新或刪除的物件群組之名稱。

versionNumber

物件群組版本。物件群組建立時，此值設定為 1。物件群組更新時，此值會增加 1。

parentGroupName

父事物群組名稱 (若有的話)。

parentGroupId

父事物群組 ID (若有的話)。

說明

物件群組描述。

rootToParentThingGroups

關於父物件群組的一系列資訊。每個父物件群組都有一個項目，從目前物件群組的父項開始，一直到根物件群組。每個項目都包含了物件群組名稱和物件群組 ARN。

attributes

與物件群組關聯的名稱/值對之集合。

從事物群組新增或移除的事物

該登錄檔會在事物新增至事物群組，或從事物群組移除時發佈以下事件訊息。

- `$aws/events/thingGroupMembership/thingGroup/<thingGroupName>/thing/<thingName>/added`
- `$aws/events/thingGroupMembership/thingGroup/<thingGroupName>/thing/<thingName>/removed`

訊息包含的承載範例如下：

```
{
    "eventType" : "THING_GROUP_MEMBERSHIP_EVENT",
    "eventId" : "d684bd5f-6f6e-48e1-950c-766ac7f02fd1",
```

```
    "timestamp" : 1234567890123,
    "operation" : "ADDED|REMOVED",
    "accountId" : "123456789012",
    "groupArn" : "arn:aws:iot:ap-northeast-2:123456789012:thinggroup/
MyChildThingGroup",
    "groupId" : "06838589-373f-4312-b1f2-53f2192291c4",
    "thingArn" : "arn:aws:iot:ap-northeast-2:123456789012:thing/MyThing",
    "thingId" : "b604f69c-aa9a-4d4a-829e-c480e958a0b5",
    "membershipId" : "8505ebf8-4d32-4286-80e9-c23a4a16bbd8"
}
```

承載包含以下屬性：

eventType

設定為「THING_GROUP_MEMBERSHIP_EVENT」。

eventId

事件 ID。

timestamp

事件發生時的 UNIX 時間戳記。

operation

當實物新增到實物群組時，`ADDED`。當實物從實物群組移除時，`REMOVED`。

accountId

您的 AWS 帳戶 ID。

groupArn

物件群組的 ARN。

groupId

群組的 ID。

thingArn

物件群組新增或移除之物件的 ARN。

thingId

物件群組新增或移除之物件的 ID。

membershipId

代表物件和物件群組之間關係的 ID。當您在物件群組新增物件，就會產生此值。

從事物群組新增或移除的事物群組

當一個事物群組新增至另一個事物群組，或從另一個事物群組中移除事物群組時，登錄檔都會發佈以下事件訊息。

- `$aws/events/thingGroupHierarchy/thingGroup/<parentThingGroupName>/childThingGroup/<childThingGroupName>/added`
- `$aws/events/thingGroupHierarchy/thingGroup/<parentThingGroupName>/childThingGroup/<childThingGroupName>/removed`

訊息包含的承載範例如下：

```
{
    "eventType" : "THING_GROUP_HIERARCHY_EVENT",
    "eventId" : "264192c7-b573-46ef-ab7b-489fcd47da41",
    "timestamp" : 1234567890123,
```

```
    "operation" : "ADDED|REMOVED",
    "accountId" : "123456789012",
    "thingGroupId" : "8f82a106-6b1d-4331-8984-a84db5f6f8cb",
    "thingGroupName" : "MyRootThingGroup",
    "childGroupId" : "06838589-373f-4312-b1f2-53f2192291c4",
    "childGroupName" : "MyChildThingGroup"
}
```

承載包含以下屬性：

eventType

設定為「THING_GROUP_HIERARCHY_EVENT」。

eventId

事件 ID。

timestamp

事件發生時的 UNIX 時間戳記。

operation

當實物新增到實物群組時，`ADDED`。當實物從實物群組移除時，`REMOVED`。

accountId

您的 AWS 帳戶 ID。

thingGroupId

父物件群組的 ID。

thingGroupName

父物件群組的名稱。

childGroupId

子物件群組的 ID。

childGroupName

子物件群組的名稱。

Jobs 事件

當工作為待定、完成或取消，以及當裝置回報執行工作成功或失敗，AWS IoT 任務服務會以 MQTT 通訊協定發佈至預留主題。藉由訂閱這些主題，裝置或管理及監控應用程式就能夠追蹤工作的狀態。使用 UpdateEventConfigurations API 來控制您想要收到的任務事件類型。

由於取消和刪除任務可能需要一些時間，系統會傳送兩個訊息來表示請求開始和結束。例如，當取消請求開始時，系統會將訊息傳送到 `$aws/events/job/jobID/cancellation_in_progress` 主題。當取消請求完成時，系統會將訊息傳送到 `$aws/events/job/jobID/canceled` 主題。任務刪除請求的程序亦同。管理及監控應用程式可以藉由訂閱這些主題，以追蹤任務的狀態。

如需發佈與訂閱 MQTT 主題的詳細資訊，請參閱 AWS IoT 的訊息中介裝置 (p. 242)。

任務完成/取消/刪除

當任務完成、取消、刪除後，或該任務正在進行取消、刪除時，AWS IoT 任務服務會發佈訊息到 MQTT 主題上：

- `$aws/events/job/`*`jobID`*`/completed`
- `$aws/events/job/`*`jobID`*`/canceled`

- $aws/events/job/*jobID*/deleted
- $aws/events/job/*jobID*/cancellation_in_progress
- $aws/events/job/*jobID*/deletion_in_progress

completed 訊息包含的承載範例如下：

```
{
  "eventType": "JOB",
  "eventId": "7364ffd1-8b65-4824-85d5-6c14686c97c6",
  "timestamp": 1234567890,
  "operation": "completed",
  "jobId": "27450507-bf6f-4012-92af-bb8a1c8c4484",
  "status": "COMPLETED",
  "targetSelection": "SNAPSHOT|CONTINUOUS",
  "targets": [
    "arn:aws:iot:us-east-1:123456789012:thing/a39f6f91-70cf-4bd2-a381-9c66df1a80d0",
    "arn:aws:iot:us-east-1:123456789012:thinggroup/2fc4c0a4-6e45-4525-
a238-0fe8d3dd21bb"
  ],
  "description": "My Job Description",
  "completedAt": 1234567890123,
  "createdAt": 1234567890123,
  "lastUpdatedAt": 1234567890123,
  "jobProcessDetails": {
    "numberOfCanceledThings": 0,
    "numberOfRejectedThings": 0,
    "numberOfFailedThings": 0,
    "numberOfRemovedThings": 0,
    "numberOfSucceededThings": 3
  }
}
```

canceled 訊息包含的承載範例如下：

```
{
  "eventType": "JOB",
  "eventId": "568d2ade-2e9c-46e6-a115-18afa1286b06",
  "timestamp": 1234567890,
  "operation": "canceled",
  "jobId": "4d2a531a-da2e-47bb-8b9e-ff5adcd53ef0",
  "status": "CANCELED",
  "targetSelection": "SNAPSHOT|CONTINUOUS",
  "targets": [
    "arn:aws:iot:us-east-1:123456789012:thing/Thing0-947b9c0c-ff10-4a80-b4b3-
cd33d0145a0f",
    "arn:aws:iot:us-east-1:123456789012:thinggroup/ThingGroup1-95c644d5-1621-41a6-9aa5-
ad2de581d18f"
  ],
  "description": "My job description",
  "createdAt": 1234567890123,
  "lastUpdatedAt": 1234567890123
}
```

deleted 訊息包含的承載範例如下：

```
{
      "eventType": "JOB",
      "eventId": "568d2ade-2e9c-46e6-a115-18afa1286b06",
      "timestamp": 1234567890,
      "operation": "deleted",
      "jobId": "4d2a531a-da2e-47bb-8b9e-ff5adcd53ef0",
      "status": "DELETED",
```

```
      "targetSelection": "SNAPSHOT|CONTINUOUS",
      "targets": [
        "arn:aws:iot:us-east-1:123456789012:thing/Thing0-947b9c0c-ff10-4a80-b4b3-
cd33d0145a0f",
        "arn:aws:iot:us-east-1:123456789012:thinggroup/
ThingGroup1-95c644d5-1621-41a6-9aa5-ad2de581d18f"
      ],
      "description": "My job description",
      "createdAt": 1234567890123,
      "lastUpdatedAt": 1234567890123,
      "comment": "Comment for this operation"
    }
```

`cancellation_in_progress` 訊息包含的承載範例如下：

```
{
      "eventType": "JOB",
      "eventId": "568d2ade-2e9c-46e6-a115-18afa1286b06",
      "timestamp": 1234567890,
      "operation": "cancellation_in_progress",
      "jobId": "4d2a531a-da2e-47bb-8b9e-ff5adcd53ef0",
      "status": "CANCELLATION_IN_PROGRESS",
      "targetSelection": "SNAPSHOT|CONTINUOUS",
      "targets": [
        "arn:aws:iot:us-east-1:123456789012:thing/Thing0-947b9c0c-ff10-4a80-b4b3-
cd33d0145a0f",
        "arn:aws:iot:us-east-1:123456789012:thinggroup/
ThingGroup1-95c644d5-1621-41a6-9aa5-ad2de581d18f"
      ],
      "description": "My job description",
      "createdAt": 1234567890123,
      "lastUpdatedAt": 1234567890123,
      "comment": "Comment for this operation"
    }
```

`deletion_in_progress` 訊息包含的承載範例如下：

```
{
      "eventType": "JOB",
      "eventId": "568d2ade-2e9c-46e6-a115-18afa1286b06",
      "timestamp": 1234567890,
      "operation": "deletion_in_progress",
      "jobId": "4d2a531a-da2e-47bb-8b9e-ff5adcd53ef0",
      "status": "DELETION_IN_PROGRESS",
      "targetSelection": "SNAPSHOT|CONTINUOUS",
      "targets": [
        "arn:aws:iot:us-east-1:123456789012:thing/Thing0-947b9c0c-ff10-4a80-b4b3-
cd33d0145a0f",
        "arn:aws:iot:us-east-1:123456789012:thinggroup/
ThingGroup1-95c644d5-1621-41a6-9aa5-ad2de581d18f"
      ],
      "description": "My job description",
      "createdAt": 1234567890123,
      "lastUpdatedAt": 1234567890123,
      "comment": "Comment for this operation"
    }
```

工作執行結束狀態

當裝置將任務執行更新為終端狀態時，AWS IoT 任務服務會發佈訊息：

- `$aws/events/jobExecution/`*`jobID`*`/succeeded`
- `$aws/events/jobExecution/`*`jobID`*`/failed`

- `$aws/events/jobExecution/jobID/rejected`
- `$aws/events/jobExecution/jobID/canceled`
- `$aws/events/jobExecution/jobID/timed_out`
- `$aws/events/jobExecution/jobID/removed`
- `$aws/events/jobExecution/jobID/deleted`

訊息包含的承載範例如下：

```
{
  "eventType": "JOB_EXECUTION",
  "eventId": "cca89fa5-8a7f-4ced-8c20-5e653afb3572",
  "timestamp": 1234567890,
  "operation": "succeeded|failed|rejected|canceled|removed|timed_out",
  "jobId": "154b39e5-60b0-48a4-9b73-f6f8dd032d27",
  "thingArn": "arn:aws:iot:us-east-1:123456789012:myThing/6d639fbc-8f85-4a90-924d-
a2867f8366a7",
  "status": "SUCCEEDED|FAILED|REJECTED|CANCELED|REMOVED|TIMED_OUT",
  "statusDetails": {
    "key": "value"
  }
}
```

生命週期事件

AWS IoT 會對以下討論的 MQTT 主題發佈生命週期事件。這些訊息讓您能從訊息中介裝置獲得生命週期事件的通知。

> **Note**
>
> 生命週期的訊息可能不會按照順序傳送。您可能會收到重複的訊息。

連線/中斷連線事件

當用戶端連線或中斷連線，AWS IoT 就會發佈訊息至以下 MQTT 主題：

- `$aws/events/presence/connected/clientId`：用戶端已連接到訊息中介裝置。
- `$aws/events/presence/disconnected/clientId`：用戶端已與訊息中介裝置中斷連線。

以下為發佈至 `$aws/events/presence/connected/clientId`主題之連線/中斷連線訊息所包含的 JSON 元素。

clientId

連線或中斷連線之用戶端的用戶端 ID。

> **Note**
>
> 包含 # 或 + 的用戶端 ID 不會收到生命週期事件。

clientInitiatedDisconnect

僅在連線中斷訊息中發現。如果用戶端已啟動中斷連線，則為 True。否則為 False。

eventType

事件的類型。有效值為 `connected`或 `disconnected`。

principalIdentifier

用於身分驗證的登入資料。對於 TLS 交互身分驗證憑證，此為憑證 ID。對於其他連線，此為 IAM 登入資料。

sessionIdentifier

在 AWS IoT 中全域唯一的識別碼，存在於工作階段期間。

timestamp

事件發生的大致時間，以自 Unix epoch 以來的毫秒表示。時間戳記的準確性為 +/- 2 分鐘。

versionNumber

生命週期事件的版本號碼。這是每個用戶端 ID 連線的依序遞增長整數值。訂閱者戶可以使用版本號碼來推斷生命週期事件的順序。

> **Note**
>
> 用戶端連線的 Connect (連線) 和 Disconnect (中斷連線) 訊息具有相同的版本號碼。
> 版本號碼可能會略過值，並且每個事件並不保證會以 1 為值持續增加。
> 如果用戶端約一小時未連線，則版本號碼將重設為 0。若為持久工作階段，當用戶端中斷連線時間超過為持久工作階段所設的存留時間 (TTL) 後，版本號碼將重設為 0。

處理用戶端中斷連線

最佳實務是一律為生命週期事件實作等待狀態，包括「臨終遺囑」(LWT) 訊息。收到中斷連線的訊息時，您的程式碼會等待一段時間並驗證裝置是否仍然離線，才採取動作。使用 SQS 延遲佇列是執行此操作的其中一種方法。當用戶端收到 LWT 或生命週期事件時，您可以將訊息加入佇列 (例如，5 秒鐘)。當該訊息變為可用並由 Lambda 或其他服務進行處理時，您可以在進行下一個動作前，先檢查該裝置是否仍處於離線狀態。

Subscribe/Unsubscribe Events

當用戶端訂閱或取消訂閱 MQTT 主題，AWS IoT 就會發佈訊息至以下 MQTT 主題。

```
$aws/events/subscriptions/subscribed/clientId
```

或

```
$aws/events/subscriptions/unsubscribed/clientId
```

其中 `clientId` 是連線至 AWS IoT 訊息中介裝置的 MQTT 用戶端 ID。

發佈至此主題的訊息具有以下結構：

```
{
    "clientId": "186b5",
    "timestamp": 1460065214626,
    "eventType": "subscribed" | "unsubscribed",
    "sessionIdentifier": "00000000-0000-0000-0000-000000000000",
    "principalIdentifier": "000000000000/ABCDEFGHIJKLMNOPQRSTU:some-user/
ABCDEFGHIJKLMNOPQRSTU:some-user"
    "topics" : ["foo/bar","device/data","dog/cat"]
}
```

以下為發佈至 `$aws/events/subscriptions/subscribed/clientId`和 `$aws/events/subscriptions/unsubscribed/clientId` 主題之訂閱與取消訂閱訊息所包含的 JSON 元素。

clientId

訂閱或取消訂閱之用戶端的用戶端 ID。

> **Note**
>
> 包含 # 或 + 的用戶端 ID 不會收到生命週期事件。

eventType

事件的類型。有效值為 `subscribed`或 `unsubscribed`。

principalIdentifier

用於身分驗證的登入資料。對於 TLS 交互身分驗證憑證，此為憑證 ID。對於其他連線，此為 IAM 登入資料。

sessionIdentifier

在 AWS IoT 中全域唯一的識別碼，存在於工作階段期間。

timestamp

事件發生的大致時間，以自 Unix epoch 以來的毫秒表示。時間戳記的準確性為 +/- 2 分鐘。

topics

用戶端已訂閱的一系列 MQTT 主題。

> **Note**
>
> 生命週期的訊息可能不會按照順序傳送。您可能會收到重複的訊息。

適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合

適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 是 AWS IoT 的新功能，它會以經濟實惠的方式將 Alexa Voice 帶至任何連線裝置。適用於 AWS IoT 的 AVS 可降低整合 Alexa 的成本和複雜性。此功能運用 AWS IoT 將裝置中的密集運算和記憶體音訊任務卸載至雲端。由於工程用料表 (eBoM) 的成本降低，裝置製造商現在可以經濟實惠的方式將 Alexa 帶到資源有限的 IoT 裝置，讓消費者可以局部在其住家、辦公室或旅館房間直接與 Alexa 通話，以獲得環境體驗。

目前，智慧家庭 IoT 裝置採用低成本微控制器 (MCU) 建置，其記憶體有限，無法執行即時作業系統。以前，Alexa 內建產品的 AVS 解決方案需要昂貴的應用程式處理器型裝置在 Linux 或 Android 上執行，其中具有超過 50 MB 的記憶體。這些昂貴的硬體需求使得在資源有限的 IoT 裝置上整合 Alexa Voice 成本過高。適用於 AWS IoT 的 AVS 可啟用 MCU 上的 Alexa 內建功能，例如具有少於 1 MB 內嵌 RAM 的 ARM Cortex M 類別。若要這樣做，AVS 會將記憶體和運算任務卸載至雲端中的虛擬 Alexa 內建裝置。如此可以降低高達 50% 的 eBoM 成本。

> **Note**
>
> 適用於 AWS IoT 的 AVS 適用於所有提供 AWS IoT 的 AWS 區域，但中國 (北京和寧夏)、亞太區域 (香港) 和中東 (巴林) 區域除外。如需 AWS 目前的區域清單，請參閱 AWS 區域表。

適用於 AWS IoT 的 AVS 有三個元件：

- 一組保留的 MQTT 主題，可在啟用 Alexa 的裝置與 AVS 之間傳輸音訊訊息。
- 雲端中啟用虛擬 Alexa 的裝置，可將與媒體擷取、音訊解碼、音訊混合和狀態管理相關的任務從實體裝置移至虛擬裝置。
- 一組 API，可支援透過保留主題接收和傳送訊息、與裝置麥克風和喇叭連接，以及管理裝置狀態。

下圖說明這些元件如何搭配運作。它還示範了裝置製造商如何使用 Login with Amazon 服務來驗證 AVS。

裝置製造商有兩個選項可以開始使用 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合。

- 開發套件 – 我們的合作夥伴推出的開發套件可讓您輕鬆上手。NXP i.MX RT 106 A 和 適用於 Amazon AVS 的 Qualcomm Home Hub 100 開發套件是最先上市的兩個套件。你可以在適用於 AVS 的開發套件上找到它們。套件中包括與 AWS IoT 的立即可用連線能力、用於遠場音訊收取、回音消除和 Alexa Wake Word 的 AVS 認證音訊演算法，以及適用於 AWS IoT 的 AVS 應用程式碼。您可以使用功能應用程式碼，快速製作裝置原型，並在準備就緒時將實作移植到所選的 MCU 設計中，以在準備就緒時進行測試和裝置生產。
- 自訂裝置端應用程式碼 – 開發人員也可以使用公開提供的 API 來撰寫自訂 適用於 AWS IoT 的 AVS 應用程式。此 API 的文件可在 AVS 開發人員頁面上取得。您可以從 Amazon FreeRTOS 主控台 (https://console.aws.amazon.com/freertos/) 或 GitHub 下載 Amazon FreeRTOS 和 AWS IoT 裝置開發套件。

若要開始使用 NXP i.MX 1060RT 開發套件，請參閱在 NXP 裝置上開始使用 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合。

在 NXP 裝置上開始使用 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合

NXP i.MX 1060RT 開發套件可讓您使用預先設定的 NXP 帳戶預覽 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合。使用 NXP 帳戶預覽功能之後，您需要自訂套件隨附的韌體、應用程式原始碼和 NXP 行動應用程式，才能使用您自己的帳戶。本主題會逐步解說您使用預先設定的帳戶進行預覽，以及使用您自己帳戶自訂裝置的步驟。

主題

- 使用預先設定的 NXP 帳戶預覽 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合 (p. 672)

- 使用您的 AWS 和 Alexa Voice Service 開發人員帳戶來設定適用於 AWS IoT 的 AVS 整合 (p. 675)

使用預先設定的 NXP 帳戶預覽 適用於 AWS IoT 的 Alexa Voice Service (AVS) 整合

先決條件

若要遵循這些步驟，您需要下列資源。

- NXP i.MX 1060RT 開發套件
- Mac 電腦、Windows 7/10 或 Linux 電腦
- 與您的電腦搭配使用的序列驅動程式
- Android 或 iOS 行動裝置
- Android Debug Bridge (ADB)
- Amazon Alexa 帳戶

安裝和啟動開發套件

1. 啟動裝置套件。此套件隨附一張 Get Started Onboarding (入門) 卡。此卡包含啟動套件和取得必要軟體套件和參考設計檔案的說明。軟體套件包含開發套件原始碼和隨附的 Android 應用程式 (`VoiceCompanionApp.apk`)。如果您使用的是 iOS 裝置，則您必須向當地 NXP 代表請求存取 TestFlight iOS 應用程式的權限。

   下載並解壓縮軟體套件之後，您會看到下列檔案結構。

   MobileApplication 資料夾包含 Android 行動應用程式。

   Tools 資料夾包含您用來設定裝置的指令碼。

   您也可以找到下列文件：

   - SLN-ALEXA-IOT-DG.pdf – NXP 開發人員指南
   - SLN-ALEXA-IOT-MG.pdf – NXP 遷移指南
   - SLN-ALEXA-IOT-UG.pdf – NXP 使用者指南
2. 啟動裝置套件。套件隨附的 USB 分割器纜線具有電源和資料連線。

a. 將兩個 USB-A 接頭連接至您的電腦。

b. 將 USB-C 接頭連接至您的套件。您的組態可能看起來如下圖所示。

當主機板有電源時，D1 LED 會亮起並顯示綠燈。D2 LED (最接近喇叭) 表示裝置的狀態。當裝置開啟時，這個 LED 會閃爍多種顏色。當裝置執行應用程式碼時，D2 LED 閃爍黃色。裝置初始化完成時，D2 LED 會顯示橘色，表示裝置正在等待 Wi-Fi 登入資料新增至裝置。

《NXP 使用者指南》 包含裝置實體控制項的描述。

3. 使用終端機應用程式了解裝置上的用戶端應用程式如何運作。您也可以使用它來進行偵錯，並確保裝置處於正確狀態。

   請輸入下列命令來開始使用應用程式：

   - `help` – 顯示可用命令的清單。
   - `enable_usb_log` – 在裝置上啟用記錄功能。這可協助您了解應用程式正在執行的動作。
   - `logs` – 顯示日誌的最後幾行。此時，日誌應指出裝置正在等候 Wi-Fi 登入資料。

將裝置連線至 AWS IoT 和 Amazon Alexa

1. 安裝適當的行動應用程式。如果您使用的是 Android 裝置，則可以在終端機中使用 ADB 來安裝 `VoiceCompanionApp.apk` 檔案。如果您使用的是 iOS 裝置，請使用 TestFlight 應用程式。
2. 佈建 Wi-Fi 登入資料。您可以使用隨附的行動應用程式或終端機來佈建 Wi-Fi 登入資料。

   a. 請依照下列步驟使用行動應用程式來佈建 Wi-Fi 登入資料。

      i. 當裝置從其出廠的新狀態開機時，它會建立 Wi-Fi 存取點。開啟隨附的行動應用程式，然後選擇 WIFI PROVISION (WIFI 佈建) 來連線至此存取點。

      ii. 從可用清單中選擇網路。

      iii. 輸入您的密碼，然後選擇 Send (傳送)，將登入資料傳輸至裝置套件。

   b. 在終端機應用程式中，輸入以下命令。

   ```
   setup YourWiFiNetworkName YourWiFiNetworkPassword
   ```

   c. 選擇 Enter (輸入)。
3. 驗證裝置套件並連線至 AWS IoT 和 Amazon Alexa。確定您的行動裝置和裝置套件使用相同的 Wi-Fi 網路，讓這兩部裝置能夠通訊。

若要這麼做，請按下行動應用程式中的 Discover (探索) 按鈕。探索完成後，您會看到附近可用 device kits (裝置套件) 的序號清單。如果有多個裝置套件可用，您可以在終端機應用程式中輸入 **`serial_number`**，以確認套件的唯一序號。

4. 選擇要使用的裝置套件。行動應用程式會使用 Login with Amazon 服務，提示您輸入 Amazon 使用者登入資料。裝置套件會開始向 AVS 註冊裝置。D2 LED 顯示紫色，表示裝置註冊已經開始。

   Note

   如果您的行動裝置已經安裝了 Amazon 購物應用程式，則隨附的行動應用程式會自動使用登入購物應用程式的 Amazon 帳戶。

登入您的 Amazon 帳戶後，隨附的應用程式會執行以下步驟：

1. 裝置套件會從 Login with Amazon 服務接收存取字符，並開始將裝置連線至 AWS IoT 和 AVS。行動應用程式會顯示完成百分比。裝置上的 D2 LED 顯示橘色。
2. D1 LED 每隔 500 毫秒閃爍綠色，直到裝置連線至 AWS IoT 為止。
3. 當裝置連線至 AWS IoT 時，裝置套件會開始將裝置連線至 AVS。D1 LED 每隔 250 毫秒閃爍綠色。
4. 當裝置連線至 AVS 時，行動應用程式中裝置套件序號的顏色會變成黃色。行動應用程式會顯示 Complete (完成) 訊息。D1 LED 關閉，並且裝置會播放鈴聲。裝置現在會連線至 NXP 的 AWS IoT 帳戶。

您可以嘗試一些 Alexa Voice 命令，如下列範例所示：

- 「Alexa，天氣怎樣？」
- 「Alexa，播放新聞簡報。」
- 「Alexa，播放音樂。」

使用您的 AWS 和 Alexa Voice Service 開發人員帳戶來設定適用於 AWS IoT 的 AVS 整合

先決條件

對於此程序，您需要下列資源。

- NXP i.MX 1060RT 開發套件
- MCUXPresso v10.3.1
- Segger J-Link 偵錯探針
- Mac 電腦
- Android 行動裝置
- Android Studio
- Android Debug Bridge
- Amazon Alexa 帳戶
- AWS 帳戶

設定您的 AWS IoT 帳戶和佈建您的裝置

1. 產生要使用 AWS IoT 進行身分驗證的登入資料。所有裝置都必須在其韌體中安裝裝置憑證、私有金鑰和根 CA 憑證，才能與 AWS IoT 通訊。請遵循在登錄中註冊裝置中的指示，以向 AWS IoT 註冊您的裝置。如需 X.509 憑證的相關資訊，請參閱 X.509 用戶端憑證。

2. 打開《NXP 開發人員指南》。導覽至根資料夾，並打開《NXP 開發人員指南》。請遵循 Section 9 (Filesystem) (第 9 節 (檔案系統)) 中的指示，將您的用戶端憑證和金鑰轉換成 NXP 檔案系統可以讀取的二進位格式。

   使用下列指令碼和命令來轉換憑證和金鑰。`bin_dump.py` 指令碼位於 `Tools\Ivaldi.zip\Scripts\sln_alexa_iot_utils` 資料夾中。

```
python3 bin_dump.py CertificateName-certificate.pem.crt CertificateName-
certificate.pem.crt.bin
```

```
python3 bin_dump.py CertificateName-private.pem.key CertificateName-private.pem.key.bin
```

如需針對生產裝置大規模產生登入資料的相關資訊，請參閱裝置佈建。

在 Alexa Voice Service 開發人員主控台中設定您的裝置

1. 打開《NXP 遷移指南》。導覽至根資料夾，並打開《NXP 遷移指南》。
2. 若要取得您建立 AVS API 金鑰所需的 MD5 和 SHA256 簽章，請依照《NXP 遷移指南》中的 Section 2.Creating a Keystore for Android AVS Companion Application (第 2 節建立 Android AVS 配套應用程式的金鑰存放區)。(這包括導覽至 Alexa Voice Service 開發人員主控台)。
3. 若要建立 Login with Amazon 安全性設定檔，並建立 AVS 產品，請遵循《 NXP 遷移指南》Section 3 (第 3 節) 中的指示。

重建 Android 行動應用程式

1. 在 Android Studio 中開啟 VoiceCompanionApp 專案。`VoiceCompanionApp.apk` 檔案包含在 NXP 軟件套件中。
2. 在 Android Studio 中，將您在前一個程序中產生的 API 金鑰新增至 assets (資產) 資料夾中的 `api_key.txt` 檔案。這會將配套行動應用程式與 AVS 安全性設定檔同步。當您使用 Login with Amazon 服務登入時，應用程式會使用 API 金鑰。當應用程式從 Amazon 取得授權碼時，它會將此授權碼傳輸到裝置韌體。然後，裝置會從 Login with Amazon 服務取得存取和重新整理字符，以呼叫 AVS。
3. 在 Android Studio 中，選擇 Build Project (建置專案) 和 Generate a Signed Bundle/APK (產生簽署的套件/APK)。如需詳細資訊，請參閱《NXP 遷移指南》Section 4.Building and Generating the Mobile Application (第 4 節建置和產生行動應用程式)。如需以這種方式利用 Alexa 進行授權的詳細資訊，請參閱從配套應用程式進行授權 (Android/iOS)。
4. 將更新後的行動應用程式安裝至您的行動裝置。在您的 Mac 電腦上設定 ADB，並使用終端機主控台，將更新後的 Android APK 檔案安裝至您的手機。如果您已使用預先設定的 NXP 帳戶，安裝行動應用程式來預覽 適用於 AWS IoT 的 AVS 服務，則必須先解除安裝再重新安裝此應用程式。

使用您的 AWS 登入資料更新用戶端應用程式

1. 請遵循《NXP 開發人員指南》中的 Section 3 (第 3 節) 和 Section 4 (第 4 節) 中的指示，進行必要的 Segger J-Link 驅動程式修改。這些指示也會說明如何使用 MCUXPresso IDE，來匯入 Bootloader、ais_demo 和 bootstrap 專案。
2. 請遵循《NXP 遷移指南》Section 7.RT106A Firmware Changes (第 7 節 RT106A 韌體變更) 中的指示，更新原始碼並重建應用程式。建議您使用 MCUXPresso v10.3.1。

   這些原始碼變更可讓裝置韌體與您的 AWS 帳戶和 AVS 產品通訊，而非預設的 NXP 帳戶。

設定並連接您的裝置至 AWS IoT 和 Amazon Alexa

1. 將裝置重設為出廠預設值。如果您使用預先設定的 NXP 帳戶預覽 適用於 AWS IoT 的 AVS 服務，則必須在使用配套行動應用程式再次登入之前重設裝置韌體。這可確保行動應用程式和裝置使用您的安全性設定檔。請確定裝置已連線至您的電腦。按住 SW1 10 秒，以啟動原廠重設。

   NXP 軟體套件中隨附的《NXP 使用者指南》 包含裝置實體控制項的描述。
2. 重新編寫韌體的程式，以使用您為裝置產生的憑證和金鑰。將金鑰和憑證新增至您在先前程序中建立並更新的 Bootstrap 和 Ais_demo 二進位檔案。我們也建議使用預設的 Intelligent toolbox (智慧型工具箱)、app_crt 和 CA_crt 二進位檔，重新編寫韌體的程式。

   如需如何完成此任務的指示，請參閱《NXP 開發人員指南》Section 5.Building and Programming (第 5 節建置和編寫程式)。
3. 遵循上述將裝置連線至 AWS 和 Amazon Alexa (p. 674)程序中的指示。這會使用您自己的安全性設定檔和登入資料來連線至 適用於 AWS IoT 的 AVS。

AWS IoT 軟體開發套件

內容

- 適用於 Android 的 AWS Mobile SDK (p. 678)
- Arduino Yún SDK (p. 678)
- 適用於 Embedded C 的 AWS IoT 裝置 SDK (p. 678)
- AWS IoT C++ 裝置 SDK (p. 679)
- 適用於 iOS 的 AWS Mobile SDK (p. 679)
- 適用於 Java 的 AWS IoT 裝置 SDK (p. 679)
- 適用於 JavaScript 的 AWS IoT 裝置 SDK (p. 679)
- 適用於 Python 的 AWS IoT 裝置 SDK (p. 679)

AWS IoT 裝置 SDK 可協助您將裝置輕鬆快速連接至 AWS IoT。AWS IoT 裝置 SDK 包含開放原始碼程式庫、開發人員指南與範例以及移植指南，因此您能夠在自選硬體平台上，建置創新的 IoT 產品或解決方案。

> **Note**
>
> 目前僅適用於 Embedded C 的 AWS IoT 裝置 SDK 支援機群佈建 (Beta)。

適用於 Android 的 AWS Mobile SDK

適用於 Android 的 AWS SDK 內含程式庫、範本和文件，可供開發人員使用 AWS 建立連線行動應用程式。本 SDK 也包含對於呼叫 AWS IoT API 的支援。如需詳細資訊，請參閱下列內容：

- GitHub 上適用於 Android 的 AWS Mobile SDK
- 適用於 Android 的 AWS Mobile SDK 讀我檔案
- 適用於 Android 的 AWS Mobile SDK 範本

Arduino Yún SDK

開發人員可使用 AWS IoT Arduino Yún SDK 將 Arduino Yún 相容的電路板連接至 AWS IoT。將裝置連線至 AWS IoT 後，使用者便可安全地使用 AWS IoT 提供的訊息中介裝置、規則和影子，以及可使用其他 AWS 服務，例如 AWS Lambda、Kinesis 與 Amazon S3。如需詳細資訊，請參閱下列內容：

- GitHub 上的 Arduino Yún SDK
- Arduino Yún SDK 讀我檔案

適用於 Embedded C 的 AWS IoT 裝置 SDK

適用於 Embedded C 的 AWS IoT 裝置 SDK 為 C 原始檔的集合，可在嵌入式應用程式內使用，安全地連線至 AWS IoT 平台。內含傳輸用戶端、TLS 實作、使用範本。亦可支援 AWS IoT 專屬之功能，例如可存取 Device Shadow 服務的 API。其會以原始碼的形式發行，應搭配應用程式的程式碼、其他程式庫和 RTOS 等，內建在客戶的韌體中。如需詳細資訊，請參閱下列內容：

- 適用於 Embedded C 的 AWS IoT 裝置 SDK GitHub

- 適用於 Embedded C 的 AWS IoT 裝置 SDK 讀我檔案
- 適用於 Embedded C 的 AWS IoT 裝置 SDK 移植指南

AWS IoT C++ 裝置 SDK

AWS IoT C++ 裝置 SDK 可讓開發人員使用 AWS 和 AWS IoT API 建立連線應用程式。此 SDK 特別是為並未受限於資源的裝置所設計，需要訊佇列、多重執行緒支援、最新語言功能等進階功能。如需詳細資訊，請參閱下列內容：

- AWS IoT C++ 裝置 SDK GitHub
- AWS IoT C++ 裝置 SDK 讀我檔案

適用於 iOS 的 AWS Mobile SDK

適用於 iOS 的 AWS SDK 是一種開放原始碼軟體開發套件，在 Apache 開放原始碼授權下發行。適用於 iOS 的 SDK 內含程式庫、程式碼範本和文件，可協助開發人員使用 AWS 建立連線行動應用程式。本 SDK 也包含對呼叫 AWS IoT API 的支援。

- GitHub 上的適用於 iOS 的 AWS SDK
- 適用於 iOS 的 AWS SDK 讀我檔案
- 適用於 iOS 的 AWS SDK 範本

適用於 Java 的 AWS IoT 裝置 SDK

適用於 Java 的 AWS IoT 裝置 SDK 可讓 Java 開發人員透過 MQTT 或是經 WebSocket 通訊協定的 MQTT 存取 AWS IoT 平台。此 SDK 內建有影子支援。您可以使用 HTTP 方法來存取影子，包括 GET、UPDATE 與 DELETE。此 SDK 亦支援簡化的影子存取模式，開發人員只需使用 getter 和 setter 方法，即可與影子交換資料，而無需將任何 JSON 文件序列化或還原序列化。如需詳細資訊，請參閱下列內容：

- GitHub 上的適用於 Java 的 AWS IoT 裝置 SDK
- 適用於 Java 的 AWS IoT 裝置 SDK 讀我檔案

適用於 JavaScript 的 AWS IoT 裝置 SDK

aws-iot-device-sdk.js 套件可讓開發人員編寫出可使用 MQTT 或經 WebSocket 通訊協定的 MQTT 來存取 AWS IoT 的 JavaScript 應用程式。可用於 Node.js 環境和瀏覽器應用程式中。如需詳細資訊，請參閱下列內容：

- GitHub 上的適用於 JavaScript 的 AWS IoT 裝置 SDK
- 適用於 JavaScript 的 AWS IoT 裝置 SDK 讀我檔案

適用於 Python 的 AWS IoT 裝置 SDK

適用於 Python 的 AWS IoT 裝置 SDK 可讓開發人員編寫出以裝置透過 MQTT 或是經 WebSocket 通訊協定的 MQTT 存取 AWS IoT 平台的 Python 指令碼。將其裝置連線至 AWS IoT 後，使用者便可安全地使用 AWS IoT 提供的訊息中介裝置、規則和影子，以及可使用其他 AWS 服務，例如 AWS Lambda、Kinesis 與 Amazon S3 等。

- GitHub 上的適用於 Python 的 AWS IoT 裝置 SDK
- 適用於 Python 的 AWS IoT 裝置 SDK 讀我檔案

故障診斷 AWS IoT

以下資訊可能有助於診斷 AWS IoT 內的常見問題。

任務

- 診斷連線問題 (p. 681)
- 診斷規則問題 (p. 681)
- 診斷影子的問題 (p. 682)
- 診斷 Salesforce IoT 輸入串流動作問題 (p. 683)
- AWS IoT 錯誤 (p. 684)

診斷連線問題

身分驗證

我的裝置該如何驗證 AWS IoT 端點的身分？

請將 AWS IoT CA 憑證新增至用戶端信任的存放區。請檢閱 AWS IoT Core 中的伺服器驗證文件，然後遵循連結來下載適當的 CA 憑證。

如何驗證憑證是否正確設定？

請使用 OpenSSL `s_client`命令來測試至 AWS IoT 端點的連線：

```
openssl s_client -connect custom_endpoint.iot.us-east-1.amazonaws.com:8443 -
CAfile CA.pem -cert cert.pem -key privateKey.pem
```

如需使用 `openssl s_client` 的詳細資訊，請參閱 OpenSSL s_client 文件。

授權

我從中介裝置收到 `PUBNACK`或 `SUBNACK` 的回應？我要怎麼做？

請確認您所使用的憑證是否有連接用以呼叫 AWS IoT 的政策。根據預設，所有發佈/訂閱的操作都會遭拒。

診斷規則問題

CloudWatch Logs 是對規則所發生的問題進行除錯的最佳方式。如需將 CloudWatch Logs 與 AWS IoT 一起使用的詳細資訊，請參閱 使用 CloudWatch Logs 進行監控 (p. 210)。當您為 AWS IoT 啟用 CloudWatch Logs 時，您可以查看已觸發哪些規則，以及該規則為成功或失敗。也會獲得 WHERE 子句條件是否符合的資訊。

最常見的問題在於授權。若您的角色未獲得在該資源上執行 AssumeRole 的授權，即會顯示在日誌中。以下為精細記錄 (p. 213)所產生的日誌範例之一：

```
{
    "timestamp": "2017-12-09 22:49:17.954",
    "logLevel": "ERROR",
    "traceId": "ff563525-6469-506a-e141-78d40375fc4e",
    "accountId": "123456789012",
    "status": "Failure",
    "eventType": "RuleExecution",
    "clientId": "iotconsole-123456789012-3",
    "topicName": "test-topic",
    "ruleName": "rule1",
    "ruleAction": "DynamoAction",
    "resources": {
        "ItemHashKeyField": "id",
        "Table": "trashbin",
        "Operation": "Insert",
        "ItemHashKeyValue": "id",
        "IsPayloadJSON": "true"
    },
    "principalId": "ABCDEFG1234567ABCD890:outis",
    "details": "User: arn:aws:sts::123456789012:assumed-role/dynamo-testbin/5aUMInJH
 is not authorized to perform: dynamodb:PutItem on resource: arn:aws:dynamodb:us-
east-1:123456789012:table/testbin (Service: AmazonDynamoDBv2; Status Code: 400; Error Code:
 AccessDeniedException; Request ID: AKQJ987654321AKQJ123456789AKQJ987654321AKQJ987654321)"
}
```

以下為全域記錄 (p. 212)所產生的類似日誌範例：

```
2017-12-09 22:49:17.954 TRACEID:ff562535-6964-506a-e141-78d40375fc4e
PRINCIPALID:ABCDEFG1234567ABCD890:outis [ERROR] EVENT:DynamoActionFailure
TOPICNAME:test-topic CLIENTID:iotconsole-123456789012-3
MESSAGE:Dynamo Insert record failed. The error received was User:
 arn:aws:sts::123456789012:assumed-role/dynamo-testbin/5aUMInJI is not authorized to
 perform: dynamodb:PutItem on resource: arn:aws:dynamodb:us-east-1:123456789012:table/
testbin
(Service: AmazonDynamoDBv2; Status Code: 400; Error Code: AccessDeniedException; Request
 ID: AKQJ987654321AKQJ987654321AKQJ987654321AKQJ987654321).
Message arrived on: test-topic, Action: dynamo, Table: trashbin, HashKeyField: id,
 HashKeyValue: id, RangeKeyField: None, RangeKeyValue: 123456789012
No newer events found at the moment. Retry.
```

如需更多詳細資訊，請參閱 the section called “檢視日誌” (p. 230)。

外部服務受最終使用者的控制。在執行規則之前，請確認已為外部服務設定足夠的傳輸量和容量單位。

診斷影子的問題

診斷影子

問題	故障診斷指南
裝置影子文件被拒絕，並出現 "Invalid JSON document." (JSON 文件無效。)	如果您不熟悉 JSON，請修改此指南中提供的範本供自行使用。如需詳細資訊，請參閱「Device Shadow 文件語法」。
我已提交正確的 JSON，但該檔案完全未存放於裝置的影子文件中，或僅存放了部分。	請確認您是否有按照 JSON 格式編排準則進行。僅有 `desired`和 `reported` 內的 JSON 欄位會儲存。在這些部分之外的 JSON 內容 (即使格式正確) 會被忽略。

問題	故障診斷指南
我收到裝置影子超出允許大小的錯誤。	裝置影子僅支援 8 KB 的資料。請嘗試縮短您 JSON 文件內的欄位名稱，或直接建立更多「實物」以建立更多的影子。可與裝置相關聯的物件/影子數目，並不受限制。唯一的要求是，每個物件名稱在您的帳戶中皆不得重複。
當我收到裝置的影子時，其大於 8 KB。怎麼會出現這種情況？	收到裝置影子時，AWS IoT 服務對裝置影子新增中繼資料。該服務會在回應中包含此資料，而不會計入 8 KB 的限制之中。只有傳送至裝置影子的狀態文件內之 `desired`與 `reported` 狀態的資料，才會計入限制。
我的請求因為版本錯誤而遭拒。我該怎麼辦？	請執行 GET 操作，以同步到文件的最新版本。當使用 MQTT 時，訂閱 ./update/accepted 主題，以接收狀態變更的通知和 JSON 文件的最新版本。
時間戳記出現幾秒鐘的差距。	當 AWS IoT 服務收到整份文件，或將狀態文件發佈至 ./update/accepted and ./update/delta 訊息時，個別欄位的時間戳記和整個 JSON 文件均會更新。網路傳遞的訊息可能會延遲幾秒，而導致時間戳記有幾秒鐘的差距。
我的裝置可發佈於對應的影子主題且已訂閱該主題，但當我嘗試透過 HTTP REST API 更新影子文件時，卻出現 HTTP 403。	請確認已在 IAM 建立政策，以存取這類主題以及您使用的登入資料相應的動作 (UPDATE/GET/DELETE)。IAM 政策和憑證政策是獨立的。
其他問題。	Device Shadow 服務會將錯誤記錄至 CloudWatch Logs。若要識別裝置和組態的問題，啟用 CloudWatch Logs 並檢視日誌即可取得除錯資訊。

診斷 Salesforce IoT 輸入串流動作問題

執行追蹤

如何查看 Salesforce 動作的執行追蹤？

若尚未設定 CloudWatch Logs，請參閱 使用 CloudWatch Logs 進行監控 (p. 210) 一節。啟動日誌後，即可看見 Salesforce 動作的執行追蹤。

作業成功和故障

該如何確認訊息已成功傳送至 Salesforce IoT input 串流？

請於 CloudWatch Logs 檢視執行 Salesforce 動作所產生的日誌。若您看見「`Action executed successfully`」，即代表 AWS IoT 規則引擎已從 Salesforce IoT 收到確認，該訊息已成功推送至目標的輸入串流。

如果您遭遇到 Salesforce IoT 平台的任何問題，請聯絡 Salesforce IoT 支援。

若訊息並未成功傳送至 Salesforce IoT input 串流，我該怎麼辦？

請於 CloudWatch Logs 檢視執行 Salesforce 動作所產生的日誌。您可以根據日誌項嘗試以下動作：

`Failed to locate the host`

請確認該動作的 `url`參數是否正確，以及您的 Salesforce IoT 輸入串流是否存在。

`Received Internal Server Error from Salesforce`

請重試。如果問題仍存在，請聯絡 Salesforce IoT 支援部門。

`Received Bad Request Exception from Salesforce`

請檢查您傳送的承載是否有錯誤。

`Received Unsupported Media Type Exception from Salesforce`

Salesforce IoT 目前不支援二位元承載。請確認您傳送得是否為 JSON 承載。

`Received Unauthorized Exception from Salesforce`

請確認該動作的 `token`參數是否正確，以及您的字符是否仍有效。

`Received Not Found Exception from Salesforce`

請確認該動作的 `url`參數是否正確，以及您的 Salesforce IoT 輸入串流是否存在。

若您收到的錯誤未列在此處，請聯絡 AWS Support。

AWS IoT 錯誤

本節會列出 AWS IoT 傳送的錯誤代碼。

訊息中介裝置錯誤代碼

錯誤代碼	錯誤說明
400	錯誤的請求。
401	未經授權。
403	禁止。
503	服務無法使用。

身分與安全性錯誤代碼

錯誤代碼	錯誤說明
401	未經授權。

Device Shadow 錯誤代碼

錯誤代碼	錯誤說明
400	錯誤的請求。
401	未經授權。
403	禁止。
404	未找到。

錯誤代碼	錯誤說明
409	衝突。
413	請求過大。
422	無法處理請求。
429	請求過多。
500	內部錯誤。
503	服務無法使用。

AWS IoT 限制

本節說明 AWS IoT 中目前的限制。各項限制除非另有說明，否則都是區域特定規定。

物件限制

資源	限制
物件名稱大小	128 個位元組的 UTF-8 編碼字元。此限制同時適用於物件登錄和物件影子服務。
針對具有物件類型的某物件，其物件屬性的數量上限	50
針對沒有物件類型的某物件，其物件屬性的數量上限	3
可與物件建立關聯的物件類型數量	1
AWS 帳戶中物件類型的數量上限	無限制

物件群組限制

資源	描述	限制	可調整
直接子群組的數量上限	直接子群組的數量上限。	100	否
動態群組的數量上限	動態群組的數量上限。	100	否
物件群組階層	物件群組階層的最大深度。	7	否
物件群組屬性	與物件群組相關聯的屬性數量上限。	50	否
物件群組屬性名稱	物件群組屬性名稱的大小上限 (以字元為單位)。	128	否
物件群組屬性值	物件群組屬性值的大小上限 (以字元為單位)。	800	否

訊息中介限制

資源	描述	限制	可調整
每個帳戶的並行用戶端連線數上限	每個帳戶允許的並行連線數上限。	500,000	是

資源	描述	限制	可調整
每個帳戶每秒的連接請求數	AWS IoT 會限制帳戶每秒 MQTT `CONNECT` 請求的上限數。	500	是
每個用戶端 ID 秒的連接請求數	AWS IoT 會限制 MQTT `CONNECT` 請求從相同的 `accountId` 和 `clientId` 為每秒 1 MQTT `CONNECT` 操作。	1	否
每個帳戶的訂閱數	AWS IoT 限制跨所有作用中連線的訂閱數上限。	500,000	是
每個帳戶每秒訂閱數	AWS IoT 限制帳戶每秒訂閱數上限。例如，如果一秒內有兩個 MQTT `SUBSCRIBE` 請求，每個具有 3 個訂閱 (主題篩選條件)，則 AWS IoT 對此限制會將其計數為 6 個訂閱。	500	是
每個連接的訂閱數	AWS IoT 支援每個連線 50 個訂閱數。超出此數量的相同連線上的訂閱請求可能會遭到 AWS IoT 拒絕，且連線將會關閉。用戶端應驗證 `SUBACK` 訊息以確認其訂閱請求已成功處理。	50	否
每個連接每秒發佈的請求數	AWS IoT 限制每個用戶端每秒連接傳入和傳出發佈請求數上限。超過該限制的發佈請求將會遭到捨棄。	100	否

資源	描述	限制	可調整
每個帳戶每秒傳入發佈請求	訊息中介在對訂閱用戶端或規則引擎路由訊息之前，AWS IoT 所處理的所有訊息中，傳入發佈請求的計數。例如，在 `$aws/things/device/shadow/update` 主題發佈一則訊息，這樣會發佈三個額外的訊息到 `$aws/things/device/shadow/update/accepted` `$aws/things/device/shadow/update/documents` 和 `$aws/things/device/shadow/delta` 主題。在這種情況下，AWS IoT 會對此限制計成 4 個傳入發佈請求。不過，未預留主題的單一訊息，例如 `a/b`，只會計成單一傳入發佈請求。	20,000	是
每個帳戶每秒傳出發佈請求	對於符合用戶端訂閱或符合規則引擎訂閱的每個訊息，傳出的發佈請求計數。例如，兩個用戶端訂閱了主題篩選條件 `a/b`，而規則訂閱了主題篩選條件 `a/#`。主題 `a/b` 的傳入發佈請求會產生總共 3 個傳出發佈請求。	20,000	是
每個連線每秒的傳輸量	透過用戶端連線接收或傳送資料以最高傳輸率處理。超出最大傳輸量的資料的處理速度將會延遲。	512 KiB	否
傳入未確認的 QoS 1 發佈請求數上限	AWS IoT 限制每個用戶端未確認之傳入發佈請求數。當達到此限制時，此用戶端就不會再接受新的發佈請求，直到伺服器傳回 `PUBACK` 訊息為止。	100	否

資源	描述	限制	可調整
傳出未確認的 QoS 1 發佈請求數上限	AWS IoT 限制每個用戶端未確認之傳出發佈請求數。當達到此限制時，用戶端就不會再接收新的發佈請求，直到用戶端確認發佈請求為止。	100	否
遞送 QoS 1 訊息的重試間隔上限	AWS IoT 將會重新嘗試傳送未確認的服務品質 1 (QoS 1) 發佈請求至用戶端最長達一個小時。如果一小時過後，AWS IoT 未從用戶端接到 `PUBACK` 訊息，它將會放棄發佈請求。	1 小時	否
持久性工作階段到期期間	訊息中介裝置將存放 MQTT 持久性工作階段的持續時間。到期期間會在訊息中介裝置偵測到工作階段已中斷連線時開始。一旦經過到期期間，訊息中介裝置會終止工作階段並捨棄任何已排入佇列的關聯訊息。	1 小時	是

通訊協定限制

資源	描述
連接閒置 (持續作用間隔)	對於 MQTT (或 MQTT 透過 WebSocket) 連線，用戶端可以請求持續作用間隔時間為介於 30 – 1200 秒，做為 MQTT `CONNECT` 訊息的一部分。AWS IoT 會在傳送 `CONNACK` 以回應 `CONNECT` 訊息時，啟動用戶端的持續作用計時器。每當 AWS IoT 從用戶端收到 `PUBLISH`、`SUBSCRIBE`、`PING` 或 `PUBACK` 訊息時，就會重設此計時器。AWS IoT 會在用戶端的持續作用計時器已經達到 1.5x 的指定持續作用間隔 (即 1.5 的倍數) 時中斷與用戶端的連線。預設持續作用間隔時間為 1200 秒。如果用戶端要求持續作用間隔為零，則會使用預設的持續作用間隔。如果用戶端要求持續作用間隔大於 1200 秒，則會使用預設的持續作用間隔。如果用戶端請求的持續作用間隔不到 30 秒，但大於零，則伺服器會視用戶端為請求了 30 秒的持續作用間隔。
WebSocket 連接持續時間	WebSocket 連接限制為 24 小時。如果超過限制，當用戶端或伺服器嘗試傳送訊息時，WebSocket 連接就會自動關閉。
每個訂閱請求的訂閱數上限	單一 `SUBSCRIBE` 請求僅限於最多八個訂閱請求。

資源	描述
訊息大小	每個發佈請求承載限制為 128 KB。AWS IoT 服務會拒絕大於此大小的發佈請求和連接請求。
用戶端 ID 大小	128 個位元組的 UTF-8 編碼字元。
受限制的用戶端 ID 字首	`$` 會保留給 AWS IoT 產生的用戶端 ID。
主題大小	當傳送發佈請求限制為 256 個位元組的 UTF-8 編碼字元，主題會傳遞到 AWS IoT。這會排除基本擷取主題 (`$AWS/rules/`*`rule-name`*`/`) 的前三個強制區段。
受限制的主題字首	任何以 `$` 開頭的主題均由 AWS IoT 預留，而且不支援發佈和訂閱，但使用 AWS IoT 服務 (亦即物件影子) 所定義的特定主題名稱除外。
主題或主題篩選條件的斜線數量上限	發佈或訂閱請求中的主題限制為 7 個正斜線 (`/`)。這會排除基本擷取主題 (`$AWS/rules/`*`rule-name`*`/`) 強制區段中的前三個斜線。

裝置影子限制

JSON 裝置狀態文件的深度上限	JSON 裝置狀態文件的 `desired` 或 `reported` 區段層級數量上限為 5。例如： `"desired": { "one": { "two": { "three": { "four": { "five":{ } } } } } } }`
每一物件傳送中未確認訊息的數量上限。	物件影子服務針對每一物件支援最多 10 個傳送中未確認訊息。當達到此限制時，會拒絕所有新影子請求，錯誤碼為 429。
每個 AWS 帳戶的 JSON 物件數量上限。	每個 AWS 帳戶的 JSON 物件數量沒有任何限制。
JSON 狀態文件的大小上限。	8 KB。請注意，中繼資料不會影響服務限制或定價的文件大小。
物件名稱的大小上限。	128 個位元組的 UTF-8 編碼字元。
AWS 帳戶中影子的數量上限。	無限制。
每一物件每秒的請求數。	物件影子服務針對每一物件支援最多 20 個請求。請注意，這個限制是按照物件，不是按照 API。

Note

在刪除所建立的帳戶之後，或依每個客戶請求，AWS IoT 會刪除物件影子。基於運作目的，AWS IoT 服務備份保留 6 個月。

安全性和身分限制

每個區域每個 AWS 帳戶允許主題欄位相同的 CA 憑證數量上限	10
可連接至憑證或 Amazon Cognito 身分的政策數量上限。	10
命名政策版本的數量上限	5
政策文件大小上限	2048 個字元 (不含空格)
每秒可註冊的裝置憑證數量上限	15
AWS IoT 角色別名數量上限	100
自訂驗證方的數量上限	10

AWS IoT 調節限制

API	每秒交易次數
AcceptCertificateTransfer	10
AddThingToBillingGroup	60
AddThingToThingGroup	60
AssociateTargetsWithJob	10
AttachPrincipalPolicy	15
AttachPolicy	15
AttachThingPrincipal	15
CancelCertificateTransfer	10
CancelJob	10
CancelJobExecution	10
ClearDefaultAuthorizer	10
CloseTunnel	1
CreateAuthorizer	10
CreateBillingGroup	25
CreateCertificateFromCsr	15

API	每秒交易次數
CreateDynamicThingGroup	5
CreateJob	10
CreatePolicy	10
CreatePolicyVersion	10
CreateRoleAlias	10
CreateThing	15
CreateThingGroup	25
CreateThingType	15
DeleteAuthorizer	10
DeleteBillingGroup	15
DeleteCertificate	10
DeleteCACertificate	10
DeleteDynamicThingGroup	5
DeleteJob	10
DeleteJobExecution	10
DeletePolicy	10
DeletePolicyVersion	10
DeleteRegistrationCode	10
DeleteRoleAlias	10
DeleteThing	15
DeleteThingGroup	15
DeleteThingType	15
DeprecateThingType	15
DescribeAuthorizer	10
DescribeBillingGroup	100
DescribeCertificate	10
DescribeCACertificate	10
DescribeDefaultAuthorizer	10
DescribeJob	10
DescribeJobExecution	10
DescribeRoleAlias	10

API	每秒交易次數
DescribeThing	350
DescribeThingGroup	100
DescribeThingType	10
DetachThingPrincipal	15
DescribeTunnel	10
DetachPrincipalPolicy	15
DetachPolicy	15
GetEffectivePolicies	50
GetJobDocument	10
GetPolicy	10
GetPolicyVersion	15
GetRegistrationCode	10
ListAttachedPolicies	15
ListAuthorizers	10
ListBillingGroups	10
ListCACertificates	10
ListCertificates	10
ListChildThingGroups	15
ListCertificatesByCA	10
ListJobExecutionsForJob	10
ListJobExecutionsForThing	10
ListJobs	10
ListOutgoingCertificates	10
ListPolicies	10
ListPolicyPrincipals	10
ListPolicyVersions	10
ListPrincipalPolicies	15
ListPrincipalThings	10
ListRoleAliases	10
ListTagsForResource	10
ListTargetsForPolicy	10

API	每秒交易次數
ListThingGroups	10
ListThingGroupsForThing	10
ListThingPrincipals	10
ListThings	10
ListThingsInBillingGroup	25
ListThingsInThingGroup	25
ListThingTypes	10
ListTunnels	10
OpenTunnel	1
RegisterCertificate	10
RegisterCACertificate	10
RegisterThing	10
RejectCertificateTransfer	10
RemoveThingFromBillingGroup	15
RemoveThingFromThingGroup	15
SetDefaultAuthorizer	10
SetDefaultPolicyVersion	10
TagResource	10
TestAuthorization	10
TestInvokeAuthorizer	10
TransferCertificate	10
UntagResource	10
UpdateAuthorizer	10
UpdateBillingGroup	15
UpdateCertificate	10
UpdateCACertificate	10
UpdateDynamicThingGroup	5
UpdateJob	10
UpdateRoleAlias	10
UpdateThing	10
UpdateThingGroup	15

AWS IoT 規則引擎限制

每個 AWS 帳戶的規則上限數目	1000
每個規則的動作數	每個規則最多可定義 10 個動作。
規則大小	256 KB
每個帳戶每秒基本導入傳入發佈請求	高達 256 KB 的 UTF-8 編碼字元 (包括空格)。
每個帳戶每秒傳入發佈請求	20,000
基本導入傳入發佈請求會計算在基本導入主題 (以 `$aws/rules/<rule-name>` 開頭) 上發佈的所有訊息計數	20,000

AWS IoT 規則引擎 HTTP 動作限制

HTTPS 通訊協定需要由公有憑證授權單位 (CA) 所發出的有效憑證	無
允許進行 HTTP 動作的通訊埠	443 和 8443
每個帳戶的主題規則目的地數量上限	1,000
請求逾時	3,000 毫秒
每個動作的標頭數量上限	100
標頭索引鍵的大小上限	256 位元組
端點 URL 的長度上限	2 KiB
支援的 HTTP 方法	POST

AWS IoT 工作限制

資源	最小值	最大值	注意
`JobId`	1 個字元	64 個字元	`JobId` 長度不得超過 64 個字元。
`Document`	無	32768 位元組	可傳送到 AWS IoT 裝置的文件大小上限為 32 KB。
`DocumentSource`	無	1350 個字元	工作文件來源大小上限為 1350 個字元。
`Description`	無	2028 個字元	工作描述大小上限為 2028 個字元。

資源	最小值	最大值	注意
`Targets`	1	100	一項工作可擁有的目標數量。
`ExpiresInSec`	60 秒	3600 秒	預先簽章 URL 的生命週期必須設定為超過 60 秒而少於 1 小時。
`Comment`	無	2028 個字元	註解大小上限為 2028 個字元。
`MaxResults`	1	250	每頁列出結果上限為 250 筆。
`MaximumJobExecutionsPerMinute`	1	1000	設定工作推展速度。
作用中快照工作	0	100	作用中快照工作的最大數量為 100 個 (無論作用中持續工作的數量為何)。
作用中持續工作	0	100	作用中持續工作的最大數量為 100 個 (無論作用中快照任務的數量為何)。
工作文件變數替換	0	10	工作文件最多可以有 10 個變數替換，包括預先簽章的 URL。
資料保留	無	730 天	非作用中任務 (任務不在 IN_PROGRESS 中) 的任務資料和任務執行資料將在 730 天後清除。
`StatusDetail` 對應金鑰值對	1 組金鑰值對	10 組金鑰值對	
`StatusDetail` 對應金鑰大小	1 個字元	128 個字元	
`StatusDetail` 對應值大小	1 個字元	128 個字元	
`DescribeJobExecution` 和 `GetPendingJobExectuions`	無	每個帳戶 200 TPS	如果在資料平面† 叫用這些的一或多個「讀取」API 造成相關聯的 AWS 帳戶超出總計每秒 200 個讀取交易量 (TPS)，那麼將會調節造成問題的 API 呼叫數，以維持每個 AWS 帳戶最多 200 個讀取 TPS。請留意控制平面†，DescribeJobExecution 限制在每一呼叫 10 個 TPS。

資源	最小值	最大值	注意
`StartNextPendingJobExecution` 和 `UpdateJobExecution`	無	每個帳戶 200 TPS	如果在資料平面† 叫用這些的一或多個「寫入」API 造成相關聯的 AWS 帳戶超出總計每秒 200 個寫入交易量 (TPS)，那麼將會調節造成問題的 API 呼叫數，以維持每個 AWS 帳戶最多 200 個寫入 TPS。
`TimeoutConfig` 的 `inProgressTimeoutInMinutes` 屬性	1	10080	值會以分鐘計 (1 分鐘到 7 天)。
以 `UpdateJobExecution` 和 `StartNextPendingJobExecution` 傳遞 `stepTimeoutInMinutes` 值	1	10080	值會以分鐘計 (1 分鐘到 7 天)。使用 `UpdateJobExecution` API 時 -1 的值也屬有效，並捨棄先前設定的計時器。

† 如需資料平面及控制平面的定義，請參閱 存取 AWS IoT Core 的方法為何

AWS IoT 安全通道限制

描述	限制
每個通道的頻寬上限	800 Kbps
連線速率上限	10 TPS
通道生命週期逾時上限	12 小時
標記限制	如需詳細資訊，請參閱標記限制。

AWS IoT 裝置機群索引限制

資源	限制	注意
每條查詢的查詢詞彙數量上限	5	每條查詢最多可以有 5 個詞彙。
查詢長度上限	1000	您的查詢長度最多可達 1000 個位元組 UTF-8 編碼字元。
查詢結果數量上限	500	裝置機群索引服務在每條查詢將傳回最多 500 筆結果。
每個查詢詞彙的 * 萬用字元運算子數量上限	2	每個查詢詞彙可以有最多 2 個多字元萬用字元 (*)。

資源	限制	注意
每個查詢詞彙的 ? 萬用字元運算子數量上限	5	每個查詢詞彙可以有最多 5 個單一字元萬用字元 (?)。
每秒的查詢數量上限	15	每秒可執行最多 15 個搜尋查詢。
裝置機群索引中物件數量上限	無限制	可建立索引的物件數無限制。
機群索引中動態群組的數量上限	100	最多 100 個動態群組可以編製索引。
自訂欄位的數量上限。	5	您最多可以設定 5 個自訂欄位。
自訂欄位名稱的長度上限	1024	自訂欄位名稱的長度最多可有 1024 個字元。

AWS IoT 調節限制

API	每秒呼叫數上限
UpdateIndexingConfiguration	1
GetIndexingConfiguration	20
DescribeIndex	10
ListIndices	5
SearchIndex	15
GetStatistics	15
GetCardinality	15
GetPercentiles	15

AWS IoT 機群佈建限制

描述	限制
機群佈建範本的大小上限	10 KiB

AWS IoT 大量實物註冊限制

資源	限制	注意
註冊任務終止	30 天	任何待處理/未完成的大量註冊任務都會在 30 天後終止。

資源	限制	注意
資料保留政策	30 天	相關聯的大量註冊任務完成後 (可長期存留)，大量實物註冊相關資料便會在 30 天後永久刪除。
允許的註冊任務	1	對於任何特定的 AWS 帳戶，一次僅可執行一個大量註冊任務。
一行的長度上限	256K	Amazon S3 輸入 JSON 檔案中每一行的長度不得超過 256K。

AWS IoT Device Defender 限制

稽核限制

資源	限制	描述
排程稽核	上限為 5。	在 `LimitExceeded` 例外狀況發生前最多可建立 5 項排程稽核。
同時進行中的「隨需」稽核	上限為 10。	在 `LimitExceeded` 例外狀況發生前最多可建立 10 項「隨需」稽核。

偵測限制

- 每個目標 (實物群組或使用者帳戶) 的安全性描述檔數目上限為 5。
- 每個安全性描述檔的行為數上限為 100。
- 每個安全性描述檔的 `value` 元素 (計數、IP 地址、連接埠) 數目上限為 1000。
- 裝置指標報告限制為每部裝置每 5 分鐘一項指標 (各裝置每 5 分鐘不得回報超過一項指標)。
- Device Defender Detect 違反情況經產生後將存放 30 天。

IOT 命令

本章包含下列部分：

- AcceptCertificateTransfer (p. 704)
- AddThingToBillingGroup (p. 705)
- AddThingToThingGroup (p. 706)
- AssociateTargetsWithJob (p. 708)
- AttachPolicy (p. 709)
- AttachPrincipalPolicy (p. 710)
- AttachSecurityProfile (p. 712)
- AttachThingPrincipal (p. 713)
- CancelAuditMitigationActionsTask (p. 714)
- CancelAuditTask (p. 714)
- CancelCertificateTransfer (p. 715)
- CancelJob (p. 716)
- CancelJobExecution (p. 718)
- ClearDefaultAuthorizer (p. 720)
- CreateAuthorizer (p. 721)
- CreateBillingGroup (p. 722)
- CreateCertificateFromCsr (p. 724)
- CreateDynamicThingGroup (p. 726)
- CreateJob (p. 729)
- CreateKeysAndCertificate (p. 734)
- CreateMitigationAction (p. 735)
- CreateOTAUpdate (p. 738)
- CreatePolicy (p. 743)
- CreatePolicyVersion (p. 745)
- CreateRoleAlias (p. 746)
- CreateScheduledAudit (p. 748)
- CreateSecurityProfile (p. 750)
- CreateStream (p. 753)
- CreateThing (p. 756)
- CreateThingGroup (p. 758)
- CreateThingType (p. 761)
- CreateTopicRule (p. 763)
- DeleteAccountAuditConfiguration (p. 776)
- DeleteAuthorizer (p. 777)
- DeleteBillingGroup (p. 778)

- DeleteCACertificate (p. 779)
- DeleteCertificate (p. 780)
- DeleteDynamicThingGroup (p. 781)
- DeleteJob (p. 782)
- DeleteJobExecution (p. 783)
- DeleteMitigationAction (p. 785)
- DeleteOTAUpdate (p. 786)
- DeletePolicy (p. 787)
- DeletePolicyVersion (p. 788)
- DeleteRegistrationCode (p. 789)
- DeleteRoleAlias (p. 790)
- DeleteScheduledAudit (p. 791)
- DeleteSecurityProfile (p. 792)
- DeleteStream (p. 793)
- DeleteThing (p. 794)
- DeleteThingGroup (p. 795)
- DeleteThingShadow (p. 796)
- DeleteThingType (p. 797)
- DeleteTopicRule (p. 798)
- DeleteV2LoggingLevel (p. 799)
- DeprecateThingType (p. 800)
- DescribeAccountAuditConfiguration (p. 801)
- DescribeAuditFinding (p. 802)
- DescribeAuditMitigationActionsTask (p. 806)
- DescribeAuditTask (p. 810)
- DescribeAuthorizer (p. 813)
- DescribeBillingGroup (p. 815)
- DescribeCACertificate (p. 816)
- DescribeCertificate (p. 819)
- DescribeDefaultAuthorizer (p. 821)
- DescribeEndpoint (p. 823)
- DescribeEventConfigurations (p. 824)
- DescribeIndex (p. 825)
- DescribeJob (p. 826)
- DescribeJobExecution (p. 831)
- DescribeJobExecution (p. 834)
- DescribeMitigationAction (p. 836)
- DescribeRoleAlias (p. 839)
- DescribeScheduledAudit (p. 841)
- DescribeSecurityProfile (p. 843)
- DescribeStream (p. 846)
- DescribeThing (p. 848)

- DescribeThingGroup (p. 850)
- DescribeThingRegistrationTask (p. 853)
- DescribeThingType (p. 855)
- DetachPolicy (p. 857)
- DetachPrincipalPolicy (p. 858)
- DetachSecurityProfile (p. 859)
- DetachThingPrincipal (p. 860)
- DisableTopicRule (p. 861)
- EnableTopicRule (p. 862)
- GetEffectivePolicies (p. 863)
- GetIndexingConfiguration (p. 864)
- GetJobDocument (p. 866)
- GetLoggingOptions (p. 867)
- GetOTAUpdate (p. 868)
- GetPendingJobExecutions (p. 873)
- GetPolicy (p. 875)
- GetPolicyVersion (p. 876)
- GetRegistrationCode (p. 878)
- GetStatistics (p. 879)
- GetThingShadow (p. 881)
- GetTopicRule (p. 882)
- GetV2LoggingOptions (p. 895)
- ListActiveViolations (p. 896)
- ListAttachedPolicies (p. 900)
- ListAuditFindings (p. 902)
- ListAuditMitigationActionsExecutions (p. 907)
- ListAuditMitigationActionsTasks (p. 909)
- ListAuditTasks (p. 911)
- ListAuthorizers (p. 913)
- ListBillingGroups (p. 915)
- ListCACertificates (p. 917)
- ListCertificates (p. 918)
- ListCertificatesByCA (p. 920)
- ListIndices (p. 922)
- ListJobExecutionsForJob (p. 923)
- ListJobExecutionsForThing (p. 925)
- ListJobs (p. 928)
- ListMitigationActions (p. 930)
- ListOTAUpdates (p. 932)
- ListOutgoingCertificates (p. 933)
- ListPolicies (p. 935)
- ListPolicyPrincipals (p. 937)

- ListPolicyVersions (p. 938)
- ListPrincipalPolicies (p. 940)
- ListPrincipalThings (p. 941)
- ListRoleAliases (p. 943)
- ListScheduledAudits (p. 944)
- ListSecurityProfiles (p. 946)
- ListSecurityProfilesForTarget (p. 947)
- ListStreams (p. 949)
- ListTagsForResource (p. 951)
- ListTargetsForPolicy (p. 952)
- ListTargetsForSecurityProfile (p. 953)
- ListThingGroups (p. 955)
- ListThingGroupsForThing (p. 956)
- ListThingPrincipals (p. 958)
- ListThingRegistrationTaskReports (p. 959)
- ListThingRegistrationTasks (p. 961)
- ListThingTypes (p. 962)
- ListThings (p. 964)
- ListThingsInBillingGroup (p. 966)
- ListThingsInThingGroup (p. 968)
- ListTopicRules (p. 969)
- ListV2LoggingLevels (p. 970)
- ListViolationEvents (p. 972)
- 發佈 (p. 976)
- RegisterCACertificate (p. 977)
- RegisterCertificate (p. 979)
- RegisterThing (p. 981)
- RejectCertificateTransfer (p. 982)
- RemoveThingFromBillingGroup (p. 983)
- RemoveThingFromThingGroup (p. 984)
- ReplaceTopicRule (p. 985)
- SearchIndex (p. 998)
- SetDefaultAuthorizer (p. 1001)
- SetDefaultPolicyVersion (p. 1003)
- SetLoggingOptions (p. 1004)
- SetV2LoggingLevel (p. 1005)
- SetV2LoggingOptions (p. 1006)
- StartAuditMitigationActionsTask (p. 1007)
- StartNextPendingJobExecution (p. 1009)
- StartOnDemandAuditTask (p. 1011)
- StartThingRegistrationTask (p. 1013)
- StopThingRegistrationTask (p. 1014)

- TagResource (p. 1015)
- TestAuthorization (p. 1016)
- TestInvokeAuthorizer (p. 1020)
- TransferCertificate (p. 1022)
- UntagResource (p. 1023)
- UpdateAccountAuditConfiguration (p. 1024)
- UpdateAuthorizer (p. 1026)
- UpdateBillingGroup (p. 1027)
- UpdateCACertificate (p. 1029)
- UpdateCertificate (p. 1030)
- UpdateDynamicThingGroup (p. 1032)
- UpdateEventConfigurations (p. 1034)
- UpdateIndexingConfiguration (p. 1035)
- UpdateJob (p. 1036)
- UpdateJobExecution (p. 1039)
- UpdateMitigationAction (p. 1042)
- UpdateRoleAlias (p. 1045)
- UpdateScheduledAudit (p. 1046)
- UpdateSecurityProfile (p. 1048)
- UpdateStream (p. 1055)
- UpdateThing (p. 1057)
- UpdateThingGroup (p. 1059)
- UpdateThingGroupsForThing (p. 1061)
- UpdateThingShadow (p. 1062)
- ValidateSecurityProfileBehaviors (p. 1063)

AcceptCertificateTransfer

接受待定憑證傳送。憑證的預設狀態為 INACTIVE。

若要檢查待定憑證傳送，請呼叫 ListCertificates 來列舉您的憑證。

概要

```
aws iot  accept-certificate-transfer \
    --certificate-id <value> \
    [--set-as-active | --no-set-as-active]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string",
  "setAsActive": "boolean"
}
```

cli-input-json 欄位

名稱	類型	描述
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)
setAsActive	布林值	指定憑證是否作用中。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

TransferAlreadyCompletedException

無法還原憑證傳送，因為傳送已完成。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

AddThingToBillingGroup

將物件加入帳單群組。

概要

```
aws iot  add-thing-to-billing-group \
    [--billing-group-name <value>] \
    [--billing-group-arn <value>] \
    [--thing-name <value>] \
    [--thing-arn <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "billingGroupName": "string",
  "billingGroupArn": "string",
  "thingName": "string",
  "thingArn": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
billingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
billingGroupArn	string	帳單群組的 ARN。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	要新增到帳單群組的物件名稱。
thingArn	string	要新增到帳單群組的物件 ARN。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

AddThingToThingGroup

將物件加入物件群組。

概要

```
aws iot  add-thing-to-thing-group \
    [--thing-group-name <value>] \
    [--thing-group-arn <value>] \
```

```
    [--thing-name <value>] \
    [--thing-arn <value>] \
    [--override-dynamic-groups | --no-override-dynamic-groups]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "thingGroupArn": "string",
  "thingName": "string",
  "thingArn": "string",
  "overrideDynamicGroups": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件欲加入的群組名稱。
`thingGroupArn`	string	物件欲加入的群組 ARN。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲加入群組的物件名稱。
`thingArn`	string	欲加入群組的物件 ARN。
`overrideDynamicGroups`	布林值	達到 10 個群組的上限時，以靜態物件群組覆寫動態物件群組。如果物件屬於 10 個物件群組，而其中有一或多個群組為動態物件群組，則對靜態群組新增物件會從最後一個動態群組移除物件。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

AssociateTargetsWithJob

將群組與持續的工作建立關聯。必須符合以下條件：

- 建立工作時必須將 `targetSelection`欄位設定為「CONTINUOUS」。
- 工作狀態目前必須是「IN_PROGRESS」。
- 與工作關聯的目標總數不得超過 100 個。

概要

```
aws iot  associate-targets-with-job \
    --targets <value> \
    --job-id <value> \
    [--comment <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "targets": [
    "string"
  ],
  "jobId": "string",
  "comment": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`targets`	列出 成員：TargetArn	定義工作之目標的物件群組 ARN 清單。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`comment`	string 長度上限：2028 模式：[^\\p{C}]+	描述工作為何與目標建立關聯的選用註解字串。

輸出

```
{
```

```
  "jobArn": "string",
  "jobId": "string",
  "description": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
jobArn	string	識別工作的 ARN。
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
description	string 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

AttachPolicy

將政策連接至指定目標。

概要

```
aws iot  attach-policy \
    --policy-name <value> \
    --target <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "target": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	欲連接的政策名稱。
target	string	政策欲連接的身分。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

LimitExceededException

已超出限制。

AttachPrincipalPolicy

將指定政策連接至指定委託人 (憑證或其他登入資料)。

備註：此 API 已作廢。請改用 AttachPolicy。

概要

```
aws iot  attach-principal-policy \
    --policy-name <value> \
    --principal <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "principal": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`principal`	string	委託人可能是憑證 ARN (自 CreateCertificate 操作所回傳) 或 Amazon Cognito ID。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

AttachSecurityProfile

將 Device Defender 安全性描述檔與實物群組或此帳戶建立關聯。每個實物群組或帳戶最多可有 5 個安全性描述檔與其相關聯。

概要

```
aws iot  attach-security-profile \
    --security-profile-name <value> \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "securityProfileTargetArn": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	連接的安全性描述檔。
`securityProfileTargetArn`	string	安全性描述檔連接的目標 (實物群組) 的 ARN。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`LimitExceededException`

已超出限制。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

AttachThingPrincipal

將指定委託人連接至指定物件。委託人可以是 X.509 憑證、IAM 使用者、群組和角色、Amazon Cognito 身分或聯合身分。

概要

```
aws iot  attach-thing-principal \
    --thing-name <value> \
    --principal <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "principal": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`principal`	string	委託人，例如憑證或其他登入資料。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CancelAuditMitigationActionsTask

取消進行中的緩解動作任務。如果任務不在進行中，則會發生 InvalidRequestException。

概要

```
aws iot  cancel-audit-mitigation-actions-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	您要取消之任務的唯一識別符。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CancelAuditTask

取消正在進行的稽核。稽核可以是排定或隨需。如果稽核不在進行中，會發生「InvalidRequestException」。

概要

```
aws iot  cancel-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您要取消的稽核的 ID。您只能取消「IN_PROGRESS」的稽核。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CancelCertificateTransfer

取消指定憑證的待定傳送。

備註 僅傳送來源帳戶可透過此操作取消傳送。(傳送目的地則可改用 RejectCertificateTransfer。) 傳送之後，AWS IoT 會將憑證以 INACTIVE 狀態回傳至來源帳戶。目的地帳戶接受傳送之後，傳送即無法取消。

憑證傳送取消之後，憑證狀態會自 PENDING_TRANSFER 變更為 INACTIVE。

概要

```
aws iot  cancel-certificate-transfer \
```

```
    --certificate-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`TransferAlreadyCompletedException`

無法還原憑證傳送，因為傳送已完成。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CancelJob

取消工作。

概要

```
aws iot  cancel-job \
    --job-id <value> \
    [--reason-code <value>] \
    [--comment <value>] \
    [--force | --no-force]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
  "reasonCode": "string",
  "comment": "string",
  "force": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`reasonCode`	string 長度上限：128 模式：[\\p{Upper}p *`Digit`*_]+	(選用) 說明任務取消原因的原因代碼。
`comment`	string 長度上限：2028 模式：[^\\p{C}]+	描述工作為何取消的選用註解字串。
`force`	布林值	(選用) 若狀態為「IN_PROGRESS」與「QUEUED」的 `true` 任務遭刪除，則只會刪除一個狀態為「QUEUED」的任務執行。預設值為 `false`。 取消狀態為「IN_PROGRESS」的任務將導致正在執行任務的裝置無法更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已取消任務是可復原至有效狀態的。

輸出

```
{
  "jobArn": "string",
  "jobId": "string",
```

```
  "description": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`jobArn`	string	工作 ARN。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`description`	string 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

CancelJobExecution

取消執行特定任務。

概要

```
aws iot  cancel-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--force | --no-force] \
    [--expected-version <value>] \
    [--status-details <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
  "jobId": "string",
  "thingName": "string",
  "force": "boolean",
  "expectedVersion": "long",
  "statusDetails": {
    "string": "string"
  }
}
```

cli-input-json 欄位

名稱	類型	描述
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	欲取消的任務 ID。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	任務執行將被取消的物件名稱。
force	布林值	(選用) 若狀態為「IN_PROGRESS」與「QUEUED」的 true 任務將遭刪除，則只會刪除一個狀態為「QUEUED」的任務執行。若您嘗試取消為「IN_PROGRESS」的任務執行，且您未將 force 設為 true，那麼將會丟出 InvalidStateTransitionException。預設值為 false。 取消狀態為「IN_PROGRESS」的任務執行將導致裝置無法更新任務執行狀態。請謹慎使用並確認裝置可還原為有效狀態。
expectedVersion	長整數	(選用) 預期的目前任務執行版本。每次您更新工作執行，其版本編號就會增加。如果存放於 Jobs 的工作執行版本不符，則更新會被 VersionMismatch 錯誤拒絕，並傳回包含目前工作執行狀態資料的 ErrorResponse。(這樣就不必另外執行 DescribeJobExecution 請求以獲得工作執行狀態資料。)
statusDetails	對應	名稱/值對的集合，能夠描述工作執行的狀態。如果未指定，則 statusDetails 不會改變。您最多可指定 10 個名稱/值組。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InvalidStateTransitionException`

由於工作執行的當前狀態而使嘗試變更工作執行狀態的更新無效 (例如，嘗試變更將 SUCCESS 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 executionState 欄位。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

ClearDefaultAuthorizer

清除預設授權方。

概要

```
aws iot  clear-default-authorizer  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CreateAuthorizer

建立授權方。

概要

```
aws iot  create-authorizer \
    --authorizer-name <value> \
    --authorizer-function-arn <value> \
    --token-key-name <value> \
    --token-signing-public-keys <value> \
    [--status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string",
  "authorizerFunctionArn": "string",
  "tokenKeyName": "string",
  "tokenSigningPublicKeys": {
    "string": "string"
  },
  "status": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerFunctionArn`	string	授權方 Lambda 函數的 ARN。
`tokenKeyName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	用於自 HTTP 標頭擷取符記的符記金鑰名稱。
`tokenSigningPublicKeys`	對應	針對從自訂身分驗證服務回傳的數位簽章進行驗證之公開金鑰。

名稱	類型	描述
status	string	所建立授權方的要求狀態。 列舉：ACTIVE | INACTIVE

輸出

```
{
  "authorizerName": "string",
  "authorizerArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
authorizerName	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
authorizerArn	string	授權方 ARN。

錯誤

ResourceAlreadyExistsException

此資源已存在。

InvalidRequestException

請求的內容無效。

LimitExceededException

已超出限制。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

CreateBillingGroup

建立帳單群組。

概要

```
aws iot  create-billing-group \
    --billing-group-name <value> \
    [--billing-group-properties <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "billingGroupName": "string",
  "billingGroupProperties": {
    "billingGroupDescription": "string"
  },
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您希望指定給帳單群組的名稱。
`billingGroupProperties`	BillingGroupProperties	帳單群組的屬性。
`billingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	帳單群組的描述。
`tags`	列出 成員：Tag Java 類別：java.util.List	用於管理帳單群組的中繼資料。
`Key`	string	標籤的金鑰。
`Value`	string	標籤的值。

輸出

```
{
  "billingGroupName": "string",
  "billingGroupArn": "string",
  "billingGroupId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您為帳單群組提供的名稱。
`billingGroupArn`	string	帳單群組的 ARN。
`billingGroupId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	帳單群組的 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CreateCertificateFromCsr

使用指定憑證簽署要求建立 X.509 憑證。

備註：CSR 必須包含 RSA 公開金鑰 (長度至少 2048 位元) 或 ECC 公開金鑰 (至少 NIST P-256 或 NIST P-384 曲線)。

備註：重複使用相同的憑證簽署要求 (CSR) 會產生不同的憑證。

您可以建立一個目錄、複製多個 .csr 檔案至該目錄，並於命令列指定該目錄，藉此在一個批次中建立多個憑證。下列命令說明如何以一批 CSR 建立一批憑證。

假設一組 CSR 位於目錄 my-csr-directory 內：

Linux 和 OS X 的命令為：

$ ls my-csr-directory/ | xargs -I aws iot create-certificate-from-csr --certificate-signing-request file://my-csr-directory/

此命令列出 my-csr-directory 的所有 CSR，並將每一 CSR 檔案名稱導入 aws iot create-certificate-from-csr AWS CLI 命令，藉此建立 CSR 所對應的憑證。

命令的 aws iot create-certificate-from-csr 部分亦可平行執行，以加速憑證建立流程：

$ ls my-csr-directory/ | xargs -P 10 -I aws iot create-certificate-from-csr --certificate-signing-request file:// my-csr-directory/

在 Windows PowerShell 中，於 my-csr-directory 建立所有 CSR 憑證的命令為：

> ls -Name my-csr-directory | % *aws iot create-certificate-from-csr --certificate-signing-request file://my-csr-directory/$_*

在 Windows 命令提示中，於 my-csr-directory 建立所有 CSR 憑證的命令為：

> forfiles /p my-csr-directory /c "cmd /c aws iot create-certificate-from-csr --certificate-signing-request file:// @path"

概要

```
aws iot  create-certificate-from-csr \
    --certificate-signing-request <value> \
    [--set-as-active | --no-set-as-active]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateSigningRequest": "string",
  "setAsActive": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateSigningRequest`	string 長度下限：1	憑證簽署要求 (CSR)。
`setAsActive`	布林值	指定憑證是否作用中。

輸出

```
{
  "certificateArn": "string",
  "certificateId": "string",
  "certificatePem": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`certificateArn`	string	憑證的 Amazon Resource Name (ARN)。您可以使用 ARN 做為政策操作的委託人。

名稱	類型	描述
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。僅需 certificateId 即可進行憑證管理操作。
certificatePem	string 長度上限：65536；下限：1	憑證資料 (PEM 格式)。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CreateDynamicThingGroup

建立動態物件群組。

概要

```
aws iot  create-dynamic-thing-group \
    --thing-group-name <value> \
    [--thing-group-properties <value>] \
    [--index-name <value>] \
    --query-string <value> \
    [--query-version <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "thingGroupProperties": {
    "thingGroupDescription": "string",
    "attributePayload": {
      "attributes": {
        "string": "string"
      },
```

```
      "merge": "boolean"
    }
  },
  "indexName": "string",
  "queryString": "string",
  "queryVersion": "string",
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲建立的動態物件群組名稱。
thingGroupProperties	ThingGroupProperties	動態物件群組屬性。
thingGroupDescription	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
attributePayload	AttributePayload	JSON 格式的物件群組屬性。
attributes	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： *\"attributes\": {\"string1\": \"string2\"}*
merge	布林值	指定 AttributePayload提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 UpdateThing 加上一個空白屬性值。 Note merge 屬性僅於呼叫 UpdateThing 或 UpdateThingGroup 時有效。
indexName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	動態物件群組索引名稱。 Note 目前僅支援一個索引："AWS_Things"。

名稱	類型	描述
queryString	string 長度下限：1	動態物件群組搜尋查詢字串。 若需查詢字串語法的詳細資訊，請參閱查詢語法。
queryVersion	string	動態物件群組查詢版本。 Note 目前僅支援一個查詢版本："2017-09-30"。如果未指定，則查詢版本預設為此值。
tags	列出 成員：Tag Java 類別：java.util.List	可以用來管理動態實物群組的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "thingGroupName": "string",
  "thingGroupArn": "string",
  "thingGroupId": "string",
  "indexName": "string",
  "queryString": "string",
  "queryVersion": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	動態物件群組名稱。
thingGroupArn	string	動態物件群組 ARN。
thingGroupId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	動態物件群組 ID。
indexName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	動態物件群組索引名稱。

名稱	類型	描述
queryString	string 長度下限：1	動態物件群組搜尋查詢字串。
queryVersion	string	動態物件群組查詢版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`InvalidQueryException`

查詢無效。

`LimitExceededException`

已超出限制。

CreateJob

建立工作。

概要

```
aws iot  create-job \
    --job-id <value> \
    --targets <value> \
    [--document-source <value>] \
    [--document <value>] \
    [--description <value>] \
    [--presigned-url-config <value>] \
    [--target-selection <value>] \
    [--job-executions-rollout-config <value>] \
    [--abort-config <value>] \
    [--timeout-config <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
```

```
  "targets": [
    "string"
  ],
  "documentSource": "string",
  "document": "string",
  "description": "string",
  "presignedUrlConfig": {
    "roleArn": "string",
    "expiresInSec": "long"
  },
  "targetSelection": "string",
  "jobExecutionsRolloutConfig": {
    "maximumPerMinute": "integer",
    "exponentialRate": {
      "baseRatePerMinute": "integer",
      "incrementFactor": "double",
      "rateIncreaseCriteria": {
        "numberOfNotifiedThings": "integer",
        "numberOfSucceededThings": "integer"
      }
    }
  },
  "abortConfig": {
    "criteriaList": [
      {
        "failureType": "string",
        "action": "string",
        "thresholdPercentage": "double",
        "minNumberOfExecutedThings": "integer"
      }
    ]
  },
  "timeoutConfig": {
    "inProgressTimeoutInMinutes": "long"
  },
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	工作識別符，這在您 AWS 帳戶中必須是獨特的。我們建議使用 UUID。文數字元、「-」和「_」於此處有效。
targets	列出 成員：TargetArn	應收到工作的物件及物件群組清單。
documentSource	string 長度上限：1350；下限：1	工作文件的 S3 連結。
document	string 長度上限：32768	工作文件。

名稱	類型	描述
		Note 如果任務文件存放在 S3 儲存貯體，您必須使用預留位置連結指定文件。預留位置連結的格式如下： `$ aws:iot:s3-presigned-url:https://s3.amazonaws.com/bucket/key` 其中，bucket 是您的儲存貯體名稱，而 key 是您所連結在儲存貯體中的物件。
`description`	string 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
`presignedUrlConfig`	PresignedUrlConfig	預先簽章的 S3 URL 之組態資訊。
`roleArn`	字串 長度上限：2048；下限：20	授予自工作資料/更新儲存位置 S3 儲存貯體下載檔案的許可之 IAM 角色 ARN。此角色也必須授權讓 IoT 下載檔案。
`expiresInSec`	長整數 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 - 3600，預設值為 3600 秒。當 Jobs 收到對工作文件的 MQTT 請求，就會產生預先簽章的 URL。
`targetSelection`	string	指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。 列舉：CONTINUOUS | SNAPSHOT
`jobExecutionsRolloutConfig`	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。
`maximumPerMinute`	integer 範圍 - 分鐘：1	每分鐘接收待定工作的物件之數量上限。此參數可讓您建立分階段推展的任務。

名稱	類型	描述
`exponentialRate`	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
`baseRatePerMinute`	integer 範圍 - 上限：1000；下限：1	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。
`rateIncreaseCriteria`	RateIncreaseCriteria	任務啟動增加推展速率的條件。 AWS IoT 最多支援小數點後一位數 (例如，1.5，但不支援 1.55)。
`numberOfNotifiedThings`	integer 範圍 - 分鐘：1	啟動增加推展速率通知物件數量的閾值。
`numberOfSucceededThings`	integer 範圍 - 分鐘：1	啟動增加推展速率成功物件數量的閾值。
`abortConfig`	AbortConfig	可讓您建立中止任務的條件。
`criteriaList`	列出 成員：AbortCriteria Java 類別：java.util.List	用於定義中止任務規則的中止條件清單。
`failureType`	string	用於定義初始化任務中止的任務執行失敗類型。 enum: FAILED | REJECTED | TIMED_OUT | ALL
`action`	string	用於初始化任務中止的中止動作類型。 enum: CANCEL
`minNumberOfExecutedThings`	integer 範圍 - 分鐘：1	評估中止規則之前的已執行物件最低數量。
`timeoutConfig`	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會自動設為 `TIMED_OUT`。

名稱	類型	描述
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 IN_PROGRESS 狀態超過此間隔時，任務執行就會失敗，並切換到結束 TIMED_OUT 狀態。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理任務的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "jobArn": "string",
  "jobId": "string",
  "description": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
jobArn	string	工作 ARN。
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您指派給此工作的獨特識別符。
description	string 長度上限：2028 模式：[^\\p{C}]+	工作描述。

錯誤

InvalidRequestException

請求的內容無效。

ResourceNotFoundException

指定的資源不存在。

`ResourceAlreadyExistsException`

此資源已存在。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

CreateKeysAndCertificate

使用已發行的公開金鑰，建立 2048 位元 RSA 金鑰對並發行 X.509 憑證。

備註 這是 AWS IoT 發行此憑證私有金鑰的唯一機會，因此必須妥善保存。

概要

```
aws iot  create-keys-and-certificate \
    [--set-as-active | --no-set-as-active]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "setAsActive": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`setAsActive`	布林值	指定憑證是否作用中。

輸出

```
{
  "certificateArn": "string",
  "certificateId": "string",
  "certificatePem": "string",
  "keyPair": {
    "PublicKey": "string",
    "PrivateKey": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`certificateArn`	string	憑證的 ARN。

名稱	類型	描述
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。AWS IoT 發行的憑證會預設主體名稱 (如 AWS IoT 憑證)。
certificatePem	string 長度上限：65536；下限：1	憑證資料 (PEM 格式)。
keyPair	KeyPair	所產生的金鑰對。
PublicKey	string 長度下限：1	公開金鑰。
PrivateKey	string 長度下限：1	私有金鑰。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CreateMitigationAction

定義可使用 StartAuditMitigationActionsTask 套用到稽核發現結果的動作。每個緩解行動只能套用一個類型的變更。

概要

```
aws iot  create-mitigation-action \
    --action-name <value> \
    --role-arn <value> \
    --action-params <value> \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "actionName": "string",
  "roleArn": "string",
  "actionParams": {
    "updateDeviceCertificateParams": {
      "action": "string"
    },
    "updateCACertificateParams": {
      "action": "string"
    },
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "string"
      ],
      "overrideDynamicGroups": "boolean"
    },
    "replaceDefaultPolicyVersionParams": {
      "templateName": "string"
    },
    "enableIoTLoggingParams": {
      "roleArnForLogging": "string",
      "logLevel": "string"
    },
    "publishFindingToSnsParams": {
      "topicArn": "string"
    }
  },
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
actionName	string 長度上限：128 模式：[a-zA-Z0-9_-]+	動作的易記名稱。選擇可正確描述動作的易記名稱 (例如，EnableLoggingAction)。
roleArn	字串 長度上限：2048；下限：20	用來套用緩解動作的 IAM 角色的 ARN。
actionParams	MitigationActionParams	定義動作的類型和該動作的參數。
updateDeviceCertificateParams	UpdateDeviceCertificateParams	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
action	string	您要套用到裝置憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
updateCACertificateParams	UpdateCACertificateParams	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。

名稱	類型	描述
action	string	您要套用到 CA 憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
addThingsToThingGroupParams	AddThingsToThingGroupParams	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
thingGroupNames	列出 成員：ThingGroupName	觸發緩解行動新增物件的群組清單。您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。
overrideDynamicGroups	布林值	指定此緩解動作是否可以移動觸發緩解動作的實物，即使該實物為一或多個動態物件群組的一部分。
replaceDefaultPolicyVersionParams	ReplaceDefaultPolicyVersionParams	參數定義緩解行動，新增空白政策來限制權限。
templateName	字串	要套用範本的名稱。唯一支援的值為 BLANK_POLICY。 列舉：BLANK_POLICY
enableIoTLoggingParams	EnableIoTLoggingParams	定義可以指定的細節層級啟用 AWS IoT 記錄的緩解動作的參數。
roleArnForLogging	字串 長度上限：2048；下限：20	用於記錄 IAM 角色的 ARN。
logLevel	string	指定要記錄的資訊類型。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
publishFindingToSnsParams	PublishFindingToSnsParams	用來定義發佈結果到 Amazon SNS 緩解動作的參數。您可以實作自訂的動作來回應 Amazon SNS 訊息。
topicArn	string 長度上限：350	您要發佈結果主題的 ARN。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理緩解行動的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "actionArn": "string",
  "actionId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
actionArn	字串	新的緩解行動的 ARN。
actionId	string	新的緩解動作的唯一識別符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CreateOTAUpdate

於物件或群組的目標群組建立 AWS IoT OTAUpdate。

概要

```
aws iot  create-ota-update \
    --ota-update-id <value> \
    [--description <value>] \
    --targets <value> \
    [--target-selection <value>] \
    [--aws-job-executions-rollout-config <value>] \
    --files <value> \
    --role-arn <value> \
    [--additional-parameters <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
"otaUpdateId": "string",
"description": "string",
"targets": [
  "string"
],
"targetSelection": "string",
"awsJobExecutionsRolloutConfig": {
  "maximumPerMinute": "integer"
},
"files": [
  {
    "fileName": "string",
    "fileVersion": "string",
    "fileLocation": {
      "stream": {
        "streamId": "string",
        "fileId": "integer"
      },
      "s3Location": {
        "bucket": "string",
        "key": "string",
        "version": "string"
      }
    },
    "codeSigning": {
      "awsSignerJobId": "string",
      "startSigningJobParameter": {
        "signingProfileParameter": {
          "certificateArn": "string",
          "platform": "string",
          "certificatePathOnDevice": "string"
        },
        "signingProfileName": "string",
        "destination": {
          "s3Destination": {
            "bucket": "string",
            "prefix": "string"
          }
        }
      },
      "customCodeSigning": {
        "signature": {
          "inlineDocument": "blob"
        },
        "certificateChain": {
          "certificateName": "string",
          "inlineDocument": "string"
        },
        "hashAlgorithm": "string",
        "signatureAlgorithm": "string"
      }
    },
    "attributes": {
      "string": "string"
    }
  }
],
"roleArn": "string",
"additionalParameters": {
  "string": "string"
},
"tags": [
  {
    "Key": "string",
    "Value": "string"
  }
```

```
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
otaUpdateId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	欲建立的 OTA 更新 ID。
description	string 長度上限：2028 模式：[^\\p{C}]+	OTA 更新的描述。
targets	列出 成員：Target	接收 OTA 更新的目標裝置。
targetSelection	string	指定更新是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成更新後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，更新也可能會運作於物件上。例如，當物件新增至目標群組，更新就會在物件上運作，即使該更新已被原本就在群組中的所有物件完成。有效值：CONTINUOUS | SNAPSHOT。 列舉：CONTINUOUS | SNAPSHOT
awsJobExecutionsRolloutConfig	AwsJobExecutionsRolloutConfig	OTA 更新的發行組態。
maximumPerMinute	integer 範圍 - 上限：1000；下限：1	每分鐘可啟動的 OTA 更新任務執行數上限。
files	列出 成員：OTAUpdateFile	OTA 更新所串流的檔案。
fileName	string	檔案名稱。
fileVersion	string	檔案版本。
fileLocation	FileLocation	已更新韌體的位置。
stream	串流	包含 OTA 更新的串流。
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。

名稱	類型	描述
fileId	integer 範圍 - 上限：255；下限：0	與串流建立關聯的檔案 ID。
s3Location	S3Location	S3 中已更新韌體的位置。
bucket	string 長度下限：1	S3 儲存貯體。
key	string 長度下限：1	S3 金鑰。
version	string	S3 儲存貯體版本。
codeSigning	CodeSigning	檔案的代碼簽署方式。
awsSignerJobId	string	用於簽署檔案所建立的 AWSSignerJob ID。
startSigningJobParameter	StartSigningJobParameter	描述程式碼簽署任務。
signingProfileParameter	SigningProfileParameter	描述程式碼簽署描述檔。
certificateArn	string	憑證 ARN。
platform	string	您裝置的硬體平台。
certificatePathOnDevice	string	程式碼簽署憑證在您裝置上的位置。
signingProfileName	string	程式碼簽署描述檔名稱。
destination	目的地	撰寫程式碼簽章檔案的位置。
s3Destination	S3Destination	描述 S3 中已更新韌體的位置。
bucket	string 長度下限：1	內含已更新韌體的 S3 儲存貯體。
prefix	string	S3 字首。
customCodeSigning	CustomCodeSigning	代碼簽署檔案的自訂方式。
signature	CodeSigningSignature	檔案的簽章。
inlineDocument	blob	以 base64 編碼呈現的二進位代碼簽署簽章。
certificateChain	CodeSigningCertificateChain	憑證鏈。
certificateName	string	憑證名稱。
inlineDocument	string	以 base64 編碼呈現的二進位代碼簽署憑證鏈。
hashAlgorithm	string	用於以代碼簽署檔案的雜湊演算法。

名稱	類型	描述
signatureAlgorithm	string	用於以代碼簽署檔案的簽章演算法。
attributes	對應	名稱/屬性對清單。
roleArn	字串 長度上限：2048；下限：20	允許存取 AWS IoT Jobs 服務的 IAM 角色。
additionalParameters	對應	其他 OTA 更新參數清單，以名稱/值對表示。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理更新的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "otaUpdateId": "string",
  "awsIotJobId": "string",
  "otaUpdateArn": "string",
  "awsIotJobArn": "string",
  "otaUpdateStatus": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
otaUpdateId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	OTA 更新 ID。
awsIotJobId	string	與 OTA 更新建立關聯的 AWS IoT 工作 ID。
otaUpdateArn	string	OTA 更新 ARN。
awsIotJobArn	string	與 OTA 更新建立關聯的 AWS IoT 工作 ARN。
otaUpdateStatus	string	OTA 更新狀態。 列舉：CREATE_PENDING | CREATE_IN_PROGRESS | CREATE_COMPLETE | CREATE_FAILED

錯誤

`InvalidRequestException`

請求的內容無效。

`LimitExceededException`

已超出限制。

`ResourceNotFoundException`

指定的資源不存在。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ServiceUnavailableException`

此服務暫時無法使用。

CreatePolicy

建立 AWS IoT 政策。

所建立的政策為政策的預設版本。本操作會建立一個版本識別碼為 1 的政策版本，並將 1 設為政策的預設版本。

概要

```
aws iot  create-policy \
    --policy-name <value> \
    --policy-document <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "policyDocument": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。

名稱	類型	描述
policyDocument	string	描述政策的 JSON 文件。policyDocument 長度至少為 1，上限為 2048，不含空白字元。

輸出

```
{
  "policyName": "string",
  "policyArn": "string",
  "policyDocument": "string",
  "policyVersionId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
policyDocument	string	描述政策的 JSON 文件。
policyVersionId	string 模式：[0-9]+	政策版本 ID。

錯誤

ResourceAlreadyExistsException

此資源已存在。

MalformedPolicyException

政策文件無效。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

CreatePolicyVersion

建立特定 AWS IoT 政策的新版本。若要更新政策，請建立新的政策版本。受管政策至多可有 5 個版本。若政策目前有五個版本，您必須使用 DeletePolicyVersion 刪除現有版本，之後才能建立新的版本。

您也可以將新版本設定為政策的預設版本。預設版本為實施版本 (亦即連接至憑證的政策之生效版本)。

概要

```
aws iot  create-policy-version \
    --policy-name <value> \
    --policy-document <value> \
    [--set-as-default | --no-set-as-default]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "policyDocument": "string",
  "setAsDefault": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`policyDocument`	string	描述政策的 JSON 文件。長度下限為 1。上限為 2048，不含空白字元。
`setAsDefault`	布林值	指定該政策版本是否設定為預設版本。若本參數為真，新的政策版本會成為實施版本 (亦即連接至憑證的政策之生效版本)。

輸出

```
{
  "policyArn": "string",
  "policyDocument": "string",
  "policyVersionId": "string",
  "isDefaultVersion": "boolean"
}
```

CLI 輸出欄位

名稱	類型	描述
`policyArn`	string	政策 ARN。

名稱	類型	描述
policyDocument	string	描述政策的 JSON 文件。
policyVersionId	string 模式：[0-9]+	政策版本 ID。
isDefaultVersion	布林值	指定該政策版本是否為預設版本。

錯誤

ResourceNotFoundException

指定的資源不存在。

MalformedPolicyException

政策文件無效。

VersionsLimitExceededException

政策版本數量超過限制。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

CreateRoleAlias

建立角色別名。

概要

```
aws iot  create-role-alias \
    --role-alias <value> \
    --role-arn <value> \
    [--credential-duration-seconds <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "roleAlias": "string",
  "roleArn": "string",
```

```
  "credentialDurationSeconds": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
roleAlias	string 長度上限：128；下限：1 模式：[w=,@-]+	指向角色 ARN 的角色別名。如此無需更新裝置即可變更角色。
roleArn	字串 長度上限：2048；下限：20	角色 ARN。
credentialDurationSeconds	integer 範圍 - 上限：3600；下限：900	登入資料有效期間 (秒)。

輸出

```
{
  "roleAlias": "string",
  "roleAliasArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
roleAlias	string 長度上限：128；下限：1 模式：[w=,@-]+	角色別名。
roleAliasArn	string	角色別名 ARN。

錯誤

ResourceAlreadyExistsException

此資源已存在。

InvalidRequestException

請求的內容無效。

LimitExceededException

已超出限制。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CreateScheduledAudit

建立已排程的稽核，其以指定的時間間隔執行。

概要

```
aws iot  create-scheduled-audit \
    --frequency <value> \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    --target-check-names <value> \
    [--tags <value>] \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`frequency`	string	已排程的稽核執行的頻率。可以是「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
`dayOfMonth`	string 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程稽核的日期。可以是「1」到「31」或「最後一天」。如果「頻率」參數設定為「每月」，則此欄位為必填。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。

名稱	類型	描述
dayOfWeek	string	當週執行已排程稽核的日期。可以是「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。如果「頻率」參數設定為「每週」或「每兩週」，則此欄位為必填。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
targetCheckNames	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 DescribeAccountAuditConfiguration 查看所有檢查清單，包括已啟用的檢查，或使用 UpdateAccountAuditConfiguration 選取啟用哪些檢查。)
tags	列出 成員：Tag Java 類別：java.util.List	用於管理排程稽核的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。
scheduledAuditName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	您要指定給已排程的稽核的名稱。(上限128 個字元)

輸出

```
{
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
scheduledAuditArn	string	已排程的稽核的 ARN。

錯誤

InvalidRequestException

請求的內容無效。

ResourceAlreadyExistsException

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

CreateSecurityProfile

建立 Device Defender 安全性描述檔。

概要

```
aws iot  create-security-profile \
    --security-profile-name <value> \
    [--security-profile-description <value>] \
    [--behaviors <value>] \
    [--alert-targets <value>] \
    [--additional-metrics-to-retain <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
```

```
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
securityProfileName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供的安全性描述檔的名稱。
securityProfileDescription	string 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的說明。
behaviors	列出 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
metric	string	行為測量的內容。
criteria	BehaviorCriteria	可判斷裝置對於 metric 的操作是否正確的條件。
comparisonOperator	string	將測量的實物 (metric) 關聯到條件 (包含 value 或 statisticalThreshold) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
value	MetricValue	相較於 metric 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。

名稱	類型	描述
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，NUM_MESSAGES_SENT)。對於 statisticalThreshhold 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (durationSeconds) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (comparisonOperator)，則該裝置會被視為符合行為，否則就是違反行為。
alertTargets	對應	指定傳送提醒的目的地。(提醒始終傳送到主控台)。 當裝置 (實物) 違反行為時產生提醒。
alertTargetArn	string	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。

名稱	類型	描述
additionalMetricsToRetain	列出 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 behaviors 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理安全性描述檔的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
securityProfileName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供的安全性描述檔的名稱。
securityProfileArn	字串	安全性描述檔的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CreateStream

建立可透過 MQTT 交付一個或多個大型檔案區塊的串流。串流自 S3 等來源傳輸的資料位元組區塊，會封包為 MQTT 訊息。一個串流可與一個或多個檔案建立關聯。

概要

```
aws iot  create-stream \
    --stream-id <value> \
    [--description <value>] \
    --files <value> \
    --role-arn <value> \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "streamId": "string",
  "description": "string",
  "files": [
    {
      "fileId": "integer",
      "s3Location": {
        "bucket": "string",
        "key": "string",
        "version": "string"
      }
    }
  ],
  "roleArn": "string",
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`streamId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
`description`	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。
`files`	列出 成員：StreamFile	欲串流的檔案。
`fileId`	integer 範圍 - 上限：255；下限：0	檔案 ID。
`s3Location`	S3Location	檔案於 S3 中的位置。
`bucket`	string	S3 儲存貯體。

名稱	類型	描述
	長度下限：1	
key	string 長度下限：1	S3 金鑰。
version	string	S3 儲存貯體版本。
roleArn	字串 長度上限：2048；下限：20	此 IAM 角色允許 IoT 服務委託人存取您的 S3 檔案。
tags	列出 成員：Tag Java 類別：java.util.List	用於管理串流的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "streamId": "string",
  "streamArn": "string",
  "description": "string",
  "streamVersion": "integer"
}
```

CLI 輸出欄位

名稱	類型	描述
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
streamArn	string	串流 ARN。
description	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。
streamVersion	integer 範圍 - 上限：65535；下限：0	此串流的版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`LimitExceededException`

已超出限制。

`ResourceNotFoundException`

指定的資源不存在。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

CreateThing

在登錄檔內建立物件記錄。如果使用相同的實物名稱和組態進行多次呼叫，則呼叫將成功。如果使用相同的實物名稱但不同組態進行呼叫，則會擲回 `ResourceAlreadyExistsException`。

> **Note**
>
> 這是一個控制平面操作。如需有關授權控制平面動作的資訊，請參閱授權。

概要

```
aws iot  create-thing \
    --thing-name <value> \
    [--thing-type-name <value>] \
    [--attribute-payload <value>] \
    [--billing-group-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "thingTypeName": "string",
  "attributePayload": {
    "attributes": {
      "string": "string"
    },
    "merge": "boolean"
  },
  "billingGroupName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲建立的物件名稱。
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	與新物件建立關聯的物件類型名稱。
attributePayload	AttributePayload	JSON 文件中，包含至多三個名稱/值對的屬性承載。例如： *\"attributes\": {\"string1\": \"string2\"}*
attributes	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： *\"attributes\": {\"string1\": \"string2\"}*
merge	布林值	指定 AttributePayload提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 UpdateThing 加上一個空白屬性值。 Note merge 屬性僅於呼叫 UpdateThing 或 UpdateThingGroup 時有效。
billingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	會將物件新增到其中的帳單群組名稱。

輸出

```
{
  "thingName": "string",
  "thingArn": "string",
  "thingId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	新物件的名稱。
`thingArn`	string	新物件的 ARN。
`thingId`	string	物件 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceAlreadyExistsException`

此資源已存在。

`ResourceNotFoundException`

指定的資源不存在。

CreateThingGroup

建立物件群組。

Note

這是一個控制平面操作。如需有關授權控制平面動作的資訊，請參閱授權。

概要

```
aws iot  create-thing-group \
    --thing-group-name <value> \
    [--parent-group-name <value>] \
    [--thing-group-properties <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "parentGroupName": "string",
  "thingGroupProperties": {
    "thingGroupDescription": "string",
    "attributePayload": {
      "attributes": {
        "string": "string"
      },
      "merge": "boolean"
    }
  },
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲建立的物件群組名稱。
`parentGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	父物件群組的名稱。
`thingGroupProperties`	ThingGroupProperties	物件群組屬性。
`thingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
`attributePayload`	AttributePayload	JSON 格式的物件群組屬性。
`attributes`	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： `\"attributes\": {\"string1\": \"string2\"}`
`merge`	布林值	指定 `AttributePayload`提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 `UpdateThing` 加上一個空白屬性值。

名稱	類型	描述
		Note `merge` 屬性僅於呼叫 `UpdateThing` 或 `UpdateThingGroup` 時有效。
`tags`	列出 成員：Tag Java 類別：java.util.List	用於管理物件群組的中繼資料。
`Key`	string	標籤的金鑰。
`Value`	string	標籤的值。

輸出

```
{
  "thingGroupName": "string",
  "thingGroupArn": "string",
  "thingGroupId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件群組名稱。
`thingGroupArn`	string	物件群組 ARN。
`thingGroupId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	物件群組 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

CreateThingType

建立新物件類型。

概要

```
aws iot  create-thing-type \
    --thing-type-name <value> \
    [--thing-type-properties <value>] \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingTypeName": "string",
  "thingTypeProperties": {
    "thingTypeDescription": "string",
    "searchableAttributes": [
      "string"
    ]
  },
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingTypeName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
`thingTypeProperties`	ThingTypeProperties	ThingTypeProperties 用於欲建立的物件類型，內含新物件類型的資訊，包括描述和可搜尋物件屬性名稱清單。
`thingTypeDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件類型描述。
`searchableAttributes`	列出 成員：AttributeName	可搜尋物件屬性名稱清單。

名稱	類型	描述
	Java 類別：java.util.List	
tags	列出 成員：Tag Java 類別：java.util.List	用於管理物件類型的中繼資料。
Key	string	標籤的金鑰。
Value	string	標籤的值。

輸出

```
{
  "thingTypeName": "string",
  "thingTypeArn": "string",
  "thingTypeId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
thingTypeArn	string	物件類型的 Amazon Resource Name (ARN)。
thingTypeId	string	物件類型 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceAlreadyExistsException`

此資源已存在。

CreateTopicRule

建立規則。規則的建立屬於管理員層級的動作。任何擁有許可而能建立規則的使用者，將能夠存取規則所處理的資料。

概要

```
aws iot  create-topic-rule \
    --rule-name <value> \
    --topic-rule-payload <value> \
    [--tags <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "ruleName": "string",
  "topicRulePayload": {
    "sql": "string",
    "description": "string",
    "actions": [
      {
        "dynamoDB": {
          "tableName": "string",
          "roleArn": "string",
          "operation": "string",
          "hashKeyField": "string",
          "hashKeyValue": "string",
          "hashKeyType": "string",
          "rangeKeyField": "string",
          "rangeKeyValue": "string",
          "rangeKeyType": "string",
          "payloadField": "string"
        },
        "dynamoDBv2": {
          "roleArn": "string",
          "putItem": {
            "tableName": "string"
          }
        },
        "lambda": {
          "functionArn": "string"
        },
        "sns": {
          "targetArn": "string",
          "roleArn": "string",
          "messageFormat": "string"
        },
        "sqs": {
          "roleArn": "string",
          "queueUrl": "string",
          "useBase64": "boolean"
        },
        "kinesis": {
          "roleArn": "string",
          "streamName": "string",
          "partitionKey": "string"
        },
        "republish": {
          "roleArn": "string",
          "topic": "string",
          "qos": "integer"
```

```
        },
        "s3": {
          "roleArn": "string",
          "bucketName": "string",
          "key": "string",
          "cannedAcl": "string"
        },
        "firehose": {
          "roleArn": "string",
          "deliveryStreamName": "string",
          "separator": "string"
        },
        "cloudwatchMetric": {
          "roleArn": "string",
          "metricNamespace": "string",
          "metricName": "string",
          "metricValue": "string",
          "metricUnit": "string",
          "metricTimestamp": "string"
        },
        "cloudwatchAlarm": {
          "roleArn": "string",
          "alarmName": "string",
          "stateReason": "string",
          "stateValue": "string"
        },
        "elasticsearch": {
          "roleArn": "string",
          "endpoint": "string",
          "index": "string",
          "type": "string",
          "id": "string"
        },
        "salesforce": {
          "token": "string",
          "url": "string"
        },
        "iotAnalytics": {
          "channelArn": "string",
          "channelName": "string",
          "roleArn": "string"
        },
        "iotEvents": {
          "inputName": "string",
          "messageId": "string",
          "roleArn": "string"
        },
        "stepFunctions": {
          "executionNamePrefix": "string",
          "stateMachineName": "string",
          "roleArn": "string"
        }
      }
    ],
    "ruleDisabled": "boolean",
    "awsIotSqlVersion": "string",
    "errorAction": {
      "dynamoDB": {
        "tableName": "string",
        "roleArn": "string",
        "operation": "string",
        "hashKeyField": "string",
        "hashKeyValue": "string",
        "hashKeyType": "string",
        "rangeKeyField": "string",
        "rangeKeyValue": "string",
```

```
        "rangeKeyType": "string",
        "payloadField": "string"
      },
      "dynamoDBv2": {
        "roleArn": "string",
        "putItem": {
          "tableName": "string"
        }
      },
      "lambda": {
        "functionArn": "string"
      },
      "sns": {
        "targetArn": "string",
        "roleArn": "string",
        "messageFormat": "string"
      },
      "sqs": {
        "roleArn": "string",
        "queueUrl": "string",
        "useBase64": "boolean"
      },
      "kinesis": {
        "roleArn": "string",
        "streamName": "string",
        "partitionKey": "string"
      },
      "republish": {
        "roleArn": "string",
        "topic": "string",
        "qos": "integer"
      },
      "s3": {
        "roleArn": "string",
        "bucketName": "string",
        "key": "string",
        "cannedAcl": "string"
      },
      "firehose": {
        "roleArn": "string",
        "deliveryStreamName": "string",
        "separator": "string"
      },
      "cloudwatchMetric": {
        "roleArn": "string",
        "metricNamespace": "string",
        "metricName": "string",
        "metricValue": "string",
        "metricUnit": "string",
        "metricTimestamp": "string"
      },
      "cloudwatchAlarm": {
        "roleArn": "string",
        "alarmName": "string",
        "stateReason": "string",
        "stateValue": "string"
      },
      "elasticsearch": {
        "roleArn": "string",
        "endpoint": "string",
        "index": "string",
        "type": "string",
        "id": "string"
      },
      "salesforce": {
        "token": "string",
```

```
        "url": "string"
      },
      "iotAnalytics": {
        "channelArn": "string",
        "channelName": "string",
        "roleArn": "string"
      },
      "iotEvents": {
        "inputName": "string",
        "messageId": "string",
        "roleArn": "string"
      },
      "stepFunctions": {
        "executionNamePrefix": "string",
        "stateMachineName": "string",
        "roleArn": "string"
      }
    }
  },
  "tags": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
ruleName	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。
topicRulePayload	TopicRulePayload	規則承載。
sql	string	用於查詢主題的 SQL 陳述式。如需詳細資訊，請參閱 AWS IoT 開發人員指南的 AWS IoT SQL 參考。
description	string	規則的描述。
actions	列出 成員：Action	與規則建立關聯的動作。
dynamoDB	DynamoDBAction	寫入至 DynamoDB 資料表。
tableName	string	DynamoDB 資料表的名稱。
roleArn	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
operation	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。
hashKeyField	string	雜湊索引鍵名稱。
hashKeyValue	string	雜湊索引鍵值。

名稱	類型	描述
`hashKeyType`	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`rangeKeyField`	string	範圍索引鍵名稱。
`rangeKeyValue`	string	範圍索引鍵值。
`rangeKeyType`	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`payloadField`	string	動作承載。此名稱可自訂。
`dynamoDBv2`	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`putItem`	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
`tableName`	string	將寫入訊息資料的資料表。
`lambda`	LambdaAction	呼叫 Lambda 函數。
`functionArn`	string	Lambda 函式的 ARN。
`sns`	SnsAction	發佈至 Amazon SNS 主題。
`targetArn`	string	SNS 主題的 ARN。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。

名稱	類型	描述
messageFormat	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
sqs	SqsAction	發佈至 Amazon SQS 佇列。
roleArn	string	授予存取權限的 IAM 角色 ARN。
queueUrl	string	Amazon SQS 佇列的 URL。
useBase64	布林值	指定是否使用 Base64 編碼。
kinesis	KinesisAction	將資料寫入至 Amazon Kinesis 串流。
roleArn	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
streamName	string	Amazon Kinesis 串流的名稱。
partitionKey	string	分割區索引鍵。
republish	RepublishAction	發佈至另一個 MQTT 主題。
roleArn	string	授予存取權限的 IAM 角色 ARN。
topic	string	MQTT 主題名稱。
qos	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
s3	S3Action	寫入至 Amazon S3 儲存貯體。
roleArn	string	授予存取權限的 IAM 角色 ARN。
bucketName	string	Amazon S3 儲存貯體。
key	string	物件金鑰。

名稱	類型	描述
`cannedAcl`	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
`firehose`	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
`roleArn`	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
`deliveryStreamName`	string	交付串流名稱。
`separator`	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
`cloudwatchMetric`	CloudwatchMetricAction	擷取 CloudWatch 指標。
`roleArn`	string	允許存取 CloudWatch 指標的 IAM 角色。
`metricNamespace`	string	CloudWatch 指標命名空間名稱。
`metricName`	string	CloudWatch 指標名稱。
`metricValue`	string	CloudWatch 指標值。
`metricUnit`	string	CloudWatch 支援的指標單位。
`metricTimestamp`	string	Unix 時間戳記 (選用)。
`cloudwatchAlarm`	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
`roleArn`	string	允許存取 CloudWatch 警示的 IAM 角色。
`alarmName`	string	CloudWatch 警示名稱。
`stateReason`	string	警示變更的原因。
`stateValue`	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
`elasticsearch`	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
`roleArn`	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。

名稱	類型	描述
`endpoint`	string 模式：https?://.*	您 Elasticsearch 網域的端點。
`index`	string	欲用來存放資料的 Elasticsearch 索引。
`type`	string	欲存放文件的類型。
`id`	string	欲存放文件的唯一識別符。
`salesforce`	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
`token`	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
`url`	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w *1,20*/w *1,20*/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
`iotAnalytics`	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
`channelArn`	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。
`channelName`	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。
`roleArn`	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
`iotEvents`	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
`inputName`	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
`messageId`	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
`roleArn`	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。("Action":"iotevents:BatchPutMessage")。

名稱	類型	描述
`stepFunctions`	StepFunctionsAction	開始執行 Step Functions 狀態機器。
`executionNamePrefix`	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
`stateMachineName`	string	Step Functions 狀態機器的名稱會開始執行。
`roleArn`	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。
`ruleDisabled`	布林值	指定規則是否停用。
`awsIotSqlVersion`	string	評估規則時所用的 SQL 規則引擎版本。
`errorAction`	動作	發生錯誤時欲採取的動作。
`dynamoDB`	DynamoDBAction	寫入至 DynamoDB 資料表。
`tableName`	string	DynamoDB 資料表的名稱。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`operation`	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。
`hashKeyField`	string	雜湊索引鍵名稱。
`hashKeyValue`	string	雜湊索引鍵值。
`hashKeyType`	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`rangeKeyField`	string	範圍索引鍵名稱。
`rangeKeyValue`	string	範圍索引鍵值。
`rangeKeyType`	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`payloadField`	string	動作承載。此名稱可自訂。

名稱	類型	描述
`dynamoDBv2`	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`putItem`	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
`tableName`	string	將寫入訊息資料的資料表。
`lambda`	LambdaAction	呼叫 Lambda 函數。
`functionArn`	string	Lambda 函式的 ARN。
`sns`	SnsAction	發佈至 Amazon SNS 主題。
`targetArn`	string	SNS 主題的 ARN。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`messageFormat`	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
`sqs`	SqsAction	發佈至 Amazon SQS 佇列。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`queueUrl`	string	Amazon SQS 佇列的 URL。
`useBase64`	布林值	指定是否使用 Base64 編碼。
`kinesis`	KinesisAction	將資料寫入至 Amazon Kinesis 串流。

名稱	類型	描述
`roleArn`	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
`streamName`	string	Amazon Kinesis 串流的名稱。
`partitionKey`	string	分割區索引鍵。
`republish`	RepublishAction	發佈至另一個 MQTT 主題。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`topic`	string	MQTT 主題名稱。
`qos`	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
`s3`	S3Action	寫入至 Amazon S3 儲存貯體。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`bucketName`	string	Amazon S3 儲存貯體。
`key`	string	物件金鑰。
`cannedAcl`	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
`firehose`	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
`roleArn`	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
`deliveryStreamName`	string	交付串流名稱。
`separator`	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
`cloudwatchMetric`	CloudwatchMetricAction	擷取 CloudWatch 指標。
`roleArn`	string	允許存取 CloudWatch 指標的 IAM 角色。
`metricNamespace`	string	CloudWatch 指標命名空間名稱。
`metricName`	string	CloudWatch 指標名稱。
`metricValue`	string	CloudWatch 指標值。

名稱	類型	描述
metricUnit	string	CloudWatch 支援的指標單位。
metricTimestamp	string	Unix 時間戳記 (選用)。
cloudwatchAlarm	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
roleArn	string	允許存取 CloudWatch 警示的 IAM 角色。
alarmName	string	CloudWatch 警示名稱。
stateReason	string	警示變更的原因。
stateValue	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
elasticsearch	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
roleArn	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。
endpoint	string 模式：https?://.*	您 Elasticsearch 網域的端點。
index	string	欲用來存放資料的 Elasticsearch 索引。
type	string	欲存放文件的類型。
id	string	欲存放文件的唯一識別符。
salesforce	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
token	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
url	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w *1,20*/w *1,20*/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
iotAnalytics	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
channelArn	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。
channelName	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。

名稱	類型	描述
`roleArn`	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
`iotEvents`	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
`inputName`	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
`messageId`	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
`roleArn`	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。("Action":"iotevents:BatchPutMessage")。
`stepFunctions`	StepFunctionsAction	開始執行 Step Functions 狀態機器。
`executionNamePrefix`	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
`stateMachineName`	string	Step Functions 狀態機器的名稱會開始執行。
`roleArn`	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。
`tags`	string	可以用來管理主題規則的中繼資料。 Note 對於 URI 請求參數，請使用格式：...key1=value1&key2=value2... 對於 CLI 命令列參數，請使用格式：--tags "key1=value1&key2=value2..." 對於 cli-input-json 檔案，請使用格式："tags": "key1=value1&key2=value2..."

輸出

無

錯誤

`SqlParseException`

無法正確剖析 SQL 表達式規則。

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ResourceAlreadyExistsException`

此資源已存在。

`ServiceUnavailableException`

此服務暫時無法使用。

`ConflictingResourceUpdateException`

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

DeleteAccountAuditConfiguration

針對此帳戶為 Device Defender 稽核還原預設值。您輸入的任何組態資料會被刪除，而所有稽核檢查會重設為停用。

概要

```
aws iot  delete-account-audit-configuration \
    [--delete-scheduled-audits | --no-delete-scheduled-audits]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "deleteScheduledAudits": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`deleteScheduledAudits`	布林值	若為 true，則刪除所有排定的稽核。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteAuthorizer

刪除授權方。

概要

```
aws iot  delete-authorizer \
    --authorizer-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	欲刪除的授權方名稱。

輸出

無

錯誤

`DeleteConflictException`

您無法刪除此資源，因其連接至一個或多個資源。

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DeleteBillingGroup

刪除帳單群組。

概要

```
aws iot  delete-billing-group \
    --billing-group-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "billingGroupName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
`expectedVersion`	長整數	帳單群組的預期版本。若帳單群組版本與請求中指定的預期版本不符，則會使用 `VersionConflictException` 來拒絕 `DeleteBillingGroup` 請求。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteCACertificate

刪除已註冊的 CA 憑證。

概要

```
aws iot  delete-ca-certificate \
    --certificate-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	欲刪除憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`CertificateStateException`

不允許此憑證操作。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DeleteCertificate

刪除指定的憑證。

如果憑證連接了政策或 IoT 物件，或其狀態設定為 ACTIVE (作用中)，則該憑證無法刪除。若要刪除憑證，首先請透過 DetachPrincipalPolicy API 分離所有政策。接著，請使用 UpdateCertificate API，將憑證設定為 INACTIVE 狀態。

概要

```
aws iot  delete-certificate \
    --certificate-id <value> \
    [--force-delete | --no-force-delete]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string",
  "forceDelete": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)
`forceDelete`	布林值	如果憑證處於非作用狀態且未連接到 IoT 物件，則該憑證會遭強制刪除。

輸出

無

錯誤

`CertificateStateException`

不允許此憑證操作。

`DeleteConflictException`

您無法刪除此資源，因其連接至一個或多個資源。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DeleteDynamicThingGroup

刪除動態物件群組。

概要

```
aws iot  delete-dynamic-thing-group \
    --thing-group-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲刪除的動態物件群組名稱。
`expectedVersion`	長整數	欲刪除的動態物件群組預期版本。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteJob

刪除任務及其相關的任務執行。

刪除任務可能會需要時間，根據為任務與各種其他因素所建立的任務執行數量而定。雖然任務被刪除，任務狀態將顯示為「DELETION_IN_PROGRESS」。嘗試刪除或取消狀態為已「DELETION_IN_PROGRESS」的任務將會導致錯誤。

只有 10 個任務可能同時有「DELETION_IN_PROGRESS」狀態，或將出現 LimitExceededException。

概要

```
aws iot  delete-job \
    --job-id <value> \
    [--force | --no-force]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
  "force": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	欲刪除的任務 ID。 任務刪除完成後，當您建立新任務時可以重複使用此 jobId。不過，不建議使用此方法，且您必須確認裝置並未使用 jobId 來引用刪除的任務。

名稱	類型	描述
force	布林值	(選用) 為 true 時，您可以刪除為「IN_PROGRESS」的任務。否則，您只能刪除結束狀態 (「COMPLETED」或「CANCELED」) 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務將導致正在執行任務的裝置無法存取任務資訊或更新任務執行狀態。請謹慎執行並確認每個裝置所執行的已刪除任務是可復原至有效狀態的。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InvalidStateTransitionException`

由於工作執行的當前狀態而使嘗試變更工作執行狀態的更新無效 (例如，嘗試變更將 SUCCESS 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 executionState 欄位。

`ResourceNotFoundException`

指定的資源不存在。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

DeleteJobExecution

刪除任務執行。

概要

```
aws iot  delete-job-execution \
```

```
    --job-id <value> \
    --thing-name <value> \
    --execution-number <value> \
    [--force | --no-force]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
  "thingName": "string",
  "executionNumber": "long",
  "force": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	任務 ID 在特定裝置上的執行將被刪除。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	任務執行將被刪除的物件名稱。
`executionNumber`	長整數	欲刪除的任務執行 ID。`executionNumber` 表示特定任務在特定裝置上執行。 請注意，一旦任務執行遭刪除，`executionNumber` 可能由 IoT 重複使用，所以請確認您在此取得並使用的是正確的值。
`force`	布林值	(選用) 為 true 時，您可以刪除為「IN_PROGRESS」的任務執行。否則，您只能刪除結束狀態 (「SUCCEEDED」、「FAILED」、「REJECTED」、「REMOVED」或「CANCELED」) 的任務，否則會發生例外狀況。預設值為 false。 Note 刪除狀態為「IN_PROGRESS」的任務執行將導致裝置無法存取任務資訊或更新任務執行狀態。請謹慎使用並確認裝置可還原為有效狀態。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InvalidStateTransitionException`

由於工作執行的當前狀態而使嘗試變更工作執行狀態的更新無效 (例如，嘗試變更將 SUCCESS 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 executionState 欄位。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

DeleteMitigationAction

從您的 AWS 帳戶刪除已定義的緩解動作。

概要

```
aws iot  delete-mitigation-action \
    --action-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "actionName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`actionName`	string 長度上限：128 模式：[a-zA-Z0-9_-]+	您要刪除的緩解動作的名稱。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteOTAUpdate

刪除 OTA 更新。

概要

```
aws iot  delete-ota-update \
    --ota-update-id <value> \
    [--delete-stream | --no-delete-stream] \
    [--force-delete-aws-job | --no-force-delete-aws-job]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "otaUpdateId": "string",
  "deleteStream": "boolean",
  "forceDeleteAWSJob": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`otaUpdateId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	欲刪除的 OTA 更新 ID。
`deleteStream`	布林值	指定在刪除 OTA 更新時是否要刪除與 OTA 更新相關聯的串流。
`forceDeleteAWSJob`	布林值	指定在刪除 OTA 更新時是否要一併刪除與 OTA 更新相關聯的 AWS 任務。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ServiceUnavailableException`

此服務暫時無法使用。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

DeletePolicy

刪除指定的政策。

若某政策具有非預設版本，或是已連接至任何憑證，將無法刪除此政策。

若要刪除政策，請透過 DeletePolicyVersion API 刪除政策的所有非預設版本，再透過 DetachPrincipalPolicy API 將政策自憑證分離，接著透過 DeletePolicy API 刪除政策。

若使用 DeletePolicy 刪除政策，將刪除其預設版本。

概要

```
aws iot  delete-policy \
    --policy-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	欲刪除的政策名稱。

輸出

無

錯誤

`DeleteConflictException`

您無法刪除此資源，因其連接至一個或多個資源。

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DeletePolicyVersion

刪除指定政策的特定版本。您無法透過此 API 刪除政策的預設版本。若要刪除政策的預設版本，請使用 DeletePolicy。欲了解哪個版本的政策已標記為預設版本，請使用 ListPolicyVersions。

概要

```
aws iot  delete-policy-version \
    --policy-name <value> \
    --policy-version-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "policyVersionId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。

名稱	類型	描述
policyVersionId	string 模式：[0-9]+	政策版本 ID。

輸出

無

錯誤

DeleteConflictException

您無法刪除此資源，因其連接至一個或多個資源。

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

DeleteRegistrationCode

刪除 CA 憑證註冊碼。

概要

```
aws iot  delete-registration-code  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
}
```

輸出

無

錯誤

`ThrottlingException`

請求頻率超過限制。

`ResourceNotFoundException`

指定的資源不存在。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DeleteRoleAlias

刪除角色別名。

概要

```
aws iot  delete-role-alias \
    --role-alias <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleAlias": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`roleAlias`	string 長度上限：128；下限：1 模式：[w=,@-]+	欲刪除的角色別名。

輸出

無

錯誤

`DeleteConflictException`

您無法刪除此資源，因其連接至一個或多個資源。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DeleteScheduledAudit

刪除已排程的稽核。

概要

```
aws iot  delete-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`scheduledAuditName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	要刪除的已排程稽核的名稱。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteSecurityProfile

刪除 Device Defender 安全性描述檔。

概要

```
aws iot  delete-security-profile \
    --security-profile-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	要刪除的安全性描述檔的名稱。
`expectedVersion`	長整數	安全性描述檔的預期版本。只要更新安全性描述檔便產生新版本。如果指定的值與實際版本不同，便擲出 `VersionConflictException`。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

DeleteStream

刪除串流。

概要

```
aws iot  delete-stream \
    --stream-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "streamId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`streamId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`DeleteConflictException`

您無法刪除此資源，因其連接至一個或多個資源。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DeleteThing

刪除指定的物件。如果刪除成功或您指定的物件不存在，則將成功傳回且不含錯誤。

概要

```
aws iot  delete-thing \
    --thing-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲刪除的物件名稱。
`expectedVersion`	長整數	登錄檔內物件記錄的預期版本。若登錄檔內的記錄版本與請求指定的預期版本不符，則會使用 `VersionConflictException` 來拒絕 `DeleteThing` 請求。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DeleteThingGroup

刪除物件群組。

概要

```
aws iot  delete-thing-group \
    --thing-group-name <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲刪除的物件群組名稱。
`expectedVersion`	長整數	欲刪除的物件群組預期版本。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DeleteThingShadow

刪除特定物件的影子。

如需詳細資訊，請參閱《AWS IoT 開發人員指南》中的 DeleteThingShadow。

概要

```
aws iot-data  delete-thing-shadow \
    --thing-name <value> \
    [--shadow-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "shadowName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`shadowName`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9:_-]+	影子的名稱。

輸出

```
{
```

```
  "payload": "blob"
}
```

CLI 輸出欄位

名稱	類型	描述
payload	blob	JSON 格式的狀態資訊。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`MethodNotAllowedException`

不支援所指定的 HTTP 動詞和 URI 組合。

`UnsupportedDocumentEncodingException`

不支援該編碼。

DeleteThingType

刪除指定的物件類型。若該物件類型已與物件建立關聯，您無法將其刪除。若要刪除物件類型，首先請呼叫 DeprecateThingType 將其標記為作廢，接著呼叫 UpdateThing 變更已建立關聯物件的物件類型，藉此移除關聯物件，最後再透過 DeleteThingType 刪除該物件類型。

概要

```
aws iot  delete-thing-type \
    --thing-type-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
  "thingTypeName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

DeleteTopicRule

刪除規則。

概要

```
aws iot  delete-topic-rule \
    [--rule-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
```

```
  "ruleName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
ruleName	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。

輸出

無

錯誤

InternalException

出現未預期的錯誤。

InvalidRequestException

請求的內容無效。

ServiceUnavailableException

此服務暫時無法使用。

UnauthorizedException

未授權您執行此操作。

ConflictingResourceUpdateException

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

DeleteV2LoggingLevel

刪除記錄層級。

概要

```
aws iot  delete-v2-logging-level \
    --target-type <value> \
    --target-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "targetType": "string",
  "targetName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
targetType	string	您設定欲記錄的資源類型，必須為 THING_Group。 列舉：DEFAULT | THING_GROUP
targetName	string	您設定欲記錄的資源名稱。

輸出

無

錯誤

InternalException

出現未預期的錯誤。

InvalidRequestException

請求的內容無效。

ServiceUnavailableException

此服務暫時無法使用。

DeprecateThingType

棄用物件類型。您無法將新物件與已棄用物件類型建立關聯。

概要

```
aws iot  deprecate-thing-type \
    --thing-type-name <value> \
    [--undo-deprecate | --no-undo-deprecate]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "thingTypeName": "string",
  "undoDeprecate": "boolean"
}
```

cli-input-json 欄位

名稱	類型	描述
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲棄用的物件類型名稱。

名稱	類型	描述
undoDeprecate	布林值	指定是否取消棄用已棄用的物件類型。若此值為 true，將不會棄用該物件類型，而且物件可與其建立關聯。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

DescribeAccountAuditConfiguration

針對此帳戶取得有關 Device Defender 稽核設定的資訊。設定值包括如何傳送稽核通知，以及啟用或停用哪些稽核檢查。

概要

```
aws iot  describe-account-audit-configuration  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
}
```

輸出

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
```

```
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
    "string": {
      "enabled": "boolean"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	描述
roleArn	字串 長度上限：2048；下限：20	角色的 ARN 授與 AWS IoT 許可，可在執行稽核時存取裝置、政策、憑證和其他必要項目的相關資訊。 在對 UpdateAccountAuditConfiguration 的第一個呼叫，需要此參數。
auditNotificationTargetConfigurations	對應	有關對此帳戶傳送稽核通知的目標的資訊。
targetArn	string	傳送稽核通知的目標 ARN (SNS 主題)。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，以傳送通知至目標。
enabled	布林值	若啟用通知至目標則為 True。
auditCheckConfigurations	對應	為此帳戶啟用和停用哪些稽核檢查。
enabled	布林值	若為此帳戶啟用此稽核檢查，則為 True。

錯誤

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

DescribeAuditFinding

取得單一稽核發現結果的相關資訊。屬性包括不合規的原因、問題的嚴重性，以及傳回發現結果的稽核開始時間。

概要

```
aws iot  describe-audit-finding \
```

```
    --finding-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "findingId": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
findingId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	單一稽核發現結果的唯一識別符。您可以使用此識別符，將緩解動作套用至發現結果。

輸出

```
{
  "finding": {
    "findingId": "string",
    "taskId": "string",
    "checkName": "string",
    "taskStartTime": "timestamp",
    "findingTime": "timestamp",
    "severity": "string",
    "nonCompliantResource": {
      "resourceType": "string",
      "resourceIdentifier": {
        "deviceCertificateId": "string",
        "caCertificateId": "string",
        "cognitoIdentityPoolId": "string",
        "clientId": "string",
        "policyVersionIdentifier": {
          "policyName": "string",
          "policyVersionId": "string"
        },
        "account": "string"
      },
      "additionalInfo": {
        "string": "string"
      }
    },
    "relatedResources": [
      {
        "resourceType": "string",
        "resourceIdentifier": {
          "deviceCertificateId": "string",
          "caCertificateId": "string",
          "cognitoIdentityPoolId": "string",
          "clientId": "string",
          "policyVersionIdentifier": {
            "policyName": "string",
            "policyVersionId": "string"
          },
          "account": "string"
        },
        "additionalInfo": {
```

```
          "string": "string"
        }
      }
    ],
    "reasonForNonCompliance": "string",
    "reasonForNonComplianceCode": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`finding`	AuditFinding	
`findingId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	這組稽核發現結果的唯一識別符。此識別符用於將緩解任務套用至一或多組發現結果。
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	產生此結果 (發現) 的稽核 ID。
`checkName`	string	產生此結果的稽核檢查。
`taskStartTime`	timestamp	稽核開始的時間。
`findingTime`	timestamp	發現結果 (尋找) 的時間。
`severity`	string	結果 (發現) 的嚴重性。 列舉：CRITICAL | HIGH | MEDIUM | LOW
`nonCompliantResource`	NonCompliantResource	發現不符合稽核檢查的資源。
`resourceType`	string	未合規的資源類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
`resourceIdentifier`	ResourceIdentifier	可識別不合規資源的資訊。
`deviceCertificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
`caCertificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。

名稱	類型	描述
`cognitoIdentityPoolId`	string	Amazon Cognito 身分集區的 ID。
`clientId`	string	用戶端 ID。
`policyVersionIdentifier`	PolicyVersionIdentifier	與資源相關聯的政策版本。
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
`policyVersionId`	string 模式：[0-9]+	與資源相關聯的政策版本 ID。
`account`	string 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
`additionalInfo`	對應	有關不合規資源的其他資訊。
`relatedResources`	列出 成員：RelatedResource	相關資源的清單。
`resourceType`	string	資源的類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
`resourceIdentifier`	ResourceIdentifier	可識別資源的資訊。
`deviceCertificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
`caCertificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
`cognitoIdentityPoolId`	string	Amazon Cognito 身分集區的 ID。
`clientId`	string	用戶端 ID。
`policyVersionIdentifier`	PolicyVersionIdentifier	與資源相關聯的政策版本。

名稱	類型	描述
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	string 模式：[0-9]+	與資源相關聯的政策版本 ID。
account	string 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
additionalInfo	對應	關於資源的其他資訊。
reasonForNonCompliance	string	資源未合規的原因。
reasonForNonComplianceCode	string	可指出資源不合規原因的代碼。

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

DescribeAuditMitigationActionsTask

取得有關用來將緩解動作套用至一組稽核發現結果的稽核緩解任務的資訊。屬性包括要套用的動作、要套用項目的稽核檢查、任務狀態，以及彙總的任務統計資料。

概要

```
aws iot  describe-audit-mitigation-actions-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "taskId": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
taskId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	稽核緩解任務的唯一識別符。

輸出

```
{
  "taskStatus": "string",
  "startTime": "timestamp",
  "endTime": "timestamp",
  "taskStatistics": {
    "string": {
      "totalFindingsCount": "long",
      "failedFindingsCount": "long",
      "succeededFindingsCount": "long",
      "skippedFindingsCount": "long",
      "canceledFindingsCount": "long"
    }
  },
  "target": {
    "auditTaskId": "string",
    "findingIds": [
      "string"
    ],
    "auditCheckToReasonCodeFilter": {
      "string": [
        "string"
      ]
    }
  },
  "auditCheckToActionsMapping": {
    "string": [
      "string"
    ]
  },
  "actionsDefinition": [
    {
      "name": "string",
      "id": "string",
      "roleArn": "string",
      "actionParams": {
        "updateDeviceCertificateParams": {
          "action": "string"
        },
        "updateCACertificateParams": {
          "action": "string"
        },
        "addThingsToThingGroupParams": {
          "thingGroupNames": [
            "string"
          ],
          "overrideDynamicGroups": "boolean"
        },
        "replaceDefaultPolicyVersionParams": {
          "templateName": "string"
        },
        "enableIoTLoggingParams": {
```

```
          "roleArnForLogging": "string",
          "logLevel": "string"
        },
        "publishFindingToSnsParams": {
          "topicArn": "string"
        }
      }
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
taskStatus	string	任務目前的狀態。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
startTime	timestamp	任務開始的日期和時間。
endTime	timestamp	任務完成或取消的日期和時間。
taskStatistics	對應	將緩解任務套用至此稽核緩解動作任務的發現結果時，結果的彙總計數。
totalFindingsCount	長整數	正在套用任務的發現結果總數。
failedFindingsCount	長整數	套用時至少一個動作失敗的發現結果數目。
succeededFindingsCount	長整數	套用時所有緩解動作時成功的發現結果數目。
skippedFindingsCount	長整數	因為在參數中提供給命令的篩選條件，略過的發現結果數量。
canceledFindingsCount	長整數	套用時取消緩解動作任務的發現結果數目。
target	AuditMitigationActionsTaskTarget	識別套用緩解動作的結果。這可以是透過稽核檢查、稽核任務或一組發現結果。
auditTaskId	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	如果任務將從特定稽核對發現結果套用緩解動作，則此值可唯一識別稽核。
findingIds	列出 成員：FindingId	如果任務將套用緩解動作到一或多個列出的發現結果，這個值可唯一識別其發現結果。
auditCheckToReasonCodeFilter	對應	以稽核檢查與一組原因代碼的形式指定篩選條件，該原因代碼可識別來自稽核緩解動作任務適用之稽核的發現結果。

名稱	類型	描述
auditCheckToActionsMapping	對應	指定應套用至特定稽核檢查的緩解動作。
actionsDefinition	列出 成員：MitigationAction	指定隨著此任務套用的緩解動作及其參數。
name	string 長度上限：128 模式：[a-zA-Z0-9_-]+	緩解動作的使用者易記名稱。
id	字串	緩解行動的唯一識別符。
roleArn	字串 長度上限：2048；下限：20	用於套用此緩解動作的 IAM 角色 ARN。
actionParams	MitigationActionParams	此緩解動作的一組參數。參數會因您套用的動作類型而有所不同。
updateDeviceCertificateParams	UpdateDeviceCertificateParams	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
action	string	您要套用到裝置憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
updateCACertificateParams	UpdateCACertificateParams	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。
action	string	您要套用到 CA 憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
addThingsToThingGroupParams	AddThingsToThingGroupParams	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
thingGroupNames	列出 成員：ThingGroupName	觸發緩解行動新增物件的群組清單。您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。
overrideDynamicGroups	布林值	指定此緩解動作是否可以移動觸發緩解動作的實物，即使該實物為一或多個動態物件群組的一部分。
replaceDefaultPolicyVersionParams	ReplaceDefaultPolicyVersionParams	參數定義緩解行動，新增空白政策來限制權限。

名稱	類型	描述
templateName	字串	要套用範本的名稱。唯一支援的值為 BLANK_POLICY。 列舉：BLANK_POLICY
enableIoTLoggingParams	EnableIoTLoggingParams	定義可以指定的細節層級啟用 AWS IoT 記錄的緩解動作的參數。
roleArnForLogging	字串 長度上限：2048；下限：20	用於記錄 IAM 角色的 ARN。
logLevel	string	指定要記錄的資訊類型。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
publishFindingToSnsParams	PublishFindingToSnsParams	用來定義發佈結果到 Amazon SNS 緩解動作的參數。您可以實作自訂的動作來回應 Amazon SNS 訊息。
topicArn	string 長度上限：350	您要發佈結果主題的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeAuditTask

取得有關 Device Defender 稽核的資訊。

概要

```
aws iot  describe-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
taskId	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您需要取得其資訊的稽核的 ID。

輸出

```
{
  "taskStatus": "string",
  "taskType": "string",
  "taskStartTime": "timestamp",
  "taskStatistics": {
    "totalChecks": "integer",
    "inProgressChecks": "integer",
    "waitingForDataCollectionChecks": "integer",
    "compliantChecks": "integer",
    "nonCompliantChecks": "integer",
    "failedChecks": "integer",
    "canceledChecks": "integer"
  },
  "scheduledAuditName": "string",
  "auditDetails": {
    "string": {
      "checkRunStatus": "string",
      "checkCompliant": "boolean",
      "totalResourcesCount": "long",
      "nonCompliantResourcesCount": "long",
      "errorCode": "string",
      "message": "string"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	描述
taskStatus	string	稽核的狀態：「IN_PROGRESS」、「COMPLETED」、「FAILED」或「CANCELED」的其中之一。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
taskType	string	稽核的類型：「ON_DEMAND_AUDIT_TASK」或「SCHEDULED_AUDIT_TASK」。

名稱	類型	描述
		列舉：ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStartTime	timestamp	稽核開始的時間。
taskStatistics	TaskStatistics	有關稽核的統計資訊。
totalChecks	integer	此稽核中的檢查數量。
inProgressChecks	integer	進行中的檢查數量。
waitingForDataCollectionChecks	integer	正在等待資料收集的檢查數量。
compliantChecks	integer	找到合規資源的檢查數量。
nonCompliantChecks	integer	找到未合規之資源的檢查數量。
failedChecks	integer	檢查數量。
canceledChecks	integer	因稽核遭取消無法執行的檢查數量。
scheduledAuditName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱 (僅當稽核為已排程的稽核時)。
auditDetails	對應	有關在這個稽核期間所執行的每個檢查的詳細資訊。
checkRunStatus	string	此檢查的完成狀態。「IN_PROGRESS」、「WAITING_FOR_DATA_COLLECTION」、「CANCELED」、「COMPLETED_COMPLIANT」、「COMPLETED_NON_COMPLIANT」或「FAILED」的其中之一。 列舉：IN_PROGRESS | WAITING_FOR_DATA_COLLECTION | CANCELED | COMPLETED_COMPLIANT | COMPLETED_NON_COMPLIANT | FAILED
checkCompliant	布林值	如果檢查完成且發現所有資源合規，則為 True。
totalResourcesCount	長整數	執行檢查的資源數量。
nonCompliantResourcesCount	長整數	檢查期間發現不合規的資源數量。

名稱	類型	描述
errorCode	string	當在此稽核期間執行此檢查時，遇到的任何錯誤代碼。「INSUFFICIENT_PERMISSIONS」或「AUDIT_CHECK_DISABLED」的其中之一。
message	字串 長度上限：2048	當在此稽核期間執行此檢查時，遇到的任何錯誤相關訊息。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeAuthorizer

描述授權方。

概要

```
aws iot  describe-authorizer \
    --authorizer-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
authorizerName	string 長度上限：128；下限：1	欲描述的授權方名稱。

名稱	類型	描述
	模式：[w=,@-]+	

輸出

```
{
  "authorizerDescription": {
    "authorizerName": "string",
    "authorizerArn": "string",
    "authorizerFunctionArn": "string",
    "tokenKeyName": "string",
    "tokenSigningPublicKeys": {
      "string": "string"
    },
    "status": "string",
    "creationDate": "timestamp",
    "lastModifiedDate": "timestamp"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`authorizerDescription`	AuthorizerDescription	授權方的描述。
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerArn`	string	授權方 ARN。
`authorizerFunctionArn`	string	授權方 Lambda 函數的 ARN。
`tokenKeyName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	用於自 HTTP 標頭擷取符記的金鑰。
`tokenSigningPublicKeys`	對應	針對從自訂身分驗證服務回傳的符記簽章進行驗證之公開金鑰。
`status`	string	授權方的狀態。 列舉：ACTIVE | INACTIVE
`creationDate`	timestamp	建立授權方時的 UNIX 時間戳記。
`lastModifiedDate`	timestamp	授權方上次更新時的 UNIX 時間戳記。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DescribeBillingGroup

傳回有關帳單群組的資訊。

概要

```
aws iot  describe-billing-group \
    --billing-group-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "billingGroupName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。

輸出

```
{
  "billingGroupName": "string",
  "billingGroupId": "string",
  "billingGroupArn": "string",
  "version": "long",
  "billingGroupProperties": {
    "billingGroupDescription": "string"
  },
  "billingGroupMetadata": {
```

```
    "creationDate": "timestamp"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
billingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
billingGroupId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	帳單群組的 ID。
billingGroupArn	string	帳單群組的 ARN。
version	長整數	帳單群組的版本。
billingGroupProperties	BillingGroupProperties	帳單群組的屬性。
billingGroupDescription	string 長度上限：2028 模式：[\\p{Graph}]*	帳單群組的描述。
billingGroupMetadata	BillingGroupMetadata	關於帳單群組的其他資訊。
creationDate	timestamp	帳單群組的建立日期。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeCACertificate

描述已註冊的 CA 憑證。

概要

```
aws iot  describe-ca-certificate \
    --certificate-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證識別符。

輸出

```
{
  "certificateDescription": {
    "certificateArn": "string",
    "certificateId": "string",
    "status": "string",
    "certificatePem": "string",
    "ownedBy": "string",
    "creationDate": "timestamp",
    "autoRegistrationStatus": "string",
    "lastModifiedDate": "timestamp",
    "customerVersion": "integer",
    "generationId": "string",
    "validity": {
      "notBefore": "timestamp",
      "notAfter": "timestamp"
    }
  },
  "registrationConfig": {
    "templateBody": "string",
    "roleArn": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`certificateDescription`	CACertificateDescription	CA 憑證描述。
`certificateArn`	string	CA 憑證 ARN。
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證 ID。

名稱	類型	描述
status	string	CA 憑證的狀態。 列舉：ACTIVE | INACTIVE
certificatePem	string 長度上限：65536；下限：1	CA 憑證資料 (PEM 格式)。
ownedBy	string 長度上限：12；下限：12 模式：[0-9]+	CA 憑證的擁有者。
creationDate	timestamp	建立 CA 憑證的日期。
autoRegistrationStatus	string	CA 憑證是否設定自動註冊裝置憑證。有效值為「ENABLE」和「DISABLE」。 列舉：ENABLE | DISABLE
lastModifiedDate	timestamp	上次修改 CA 憑證的日期和時間。
customerVersion	integer 範圍 - 分鐘：1	CA 憑證的客戶版本。
generationId	string	CA 憑證的產生 ID。
validity	CertificateValidity	當 CA 憑證有效時。
notBefore	timestamp	在此日期之前憑證無效。
notAfter	timestamp	在此日期之後憑證無效。
registrationConfig	RegistrationConfig	註冊組態的資訊。
templateBody	string	範本本身。
roleArn	字串 長度上限：2048；下限：20	此角色的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeCertificate

取得有關指定憑證的資訊。

概要

```
aws iot  describe-certificate \
    --certificate-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)

輸出

```
{
  "certificateDescription": {
    "certificateArn": "string",
    "certificateId": "string",
    "caCertificateId": "string",
    "status": "string",
    "certificatePem": "string",
    "ownedBy": "string",
    "previousOwnedBy": "string",
    "creationDate": "timestamp",
    "lastModifiedDate": "timestamp",
    "customerVersion": "integer",
    "transferData": {
      "transferMessage": "string",
      "rejectReason": "string",
      "transferDate": "timestamp",
      "acceptDate": "timestamp",
      "rejectDate": "timestamp"
    },
    "generationId": "string",
    "validity": {
      "notBefore": "timestamp",
```

```
      "notAfter": "timestamp"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	描述
certificateDescription	CertificateDescription	憑證的描述。
certificateArn	string	憑證的 ARN。
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。
caCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於簽署憑證之 CA 憑證的憑證 ID。
status	string	憑證的狀態。 列舉：ACTIVE | INACTIVE | REVOKED | PENDING_TRANSFER | REGISTER_INACTIVE | PENDING_ACTIVATION
certificatePem	string 長度上限：65536；下限：1	憑證資料 (PEM 格式)。
ownedBy	string 長度上限：12；下限：12 模式：[0-9]+	擁有憑證的 AWS 帳戶 ID。
previousOwnedBy	string 長度上限：12；下限：12 模式：[0-9]+	前一個擁有憑證的 AWS 帳戶 ID。
creationDate	timestamp	建立憑證的日期和時間。
lastModifiedDate	timestamp	上次修改憑證的日期和時間。
customerVersion	integer 範圍 - 分鐘：1	憑證的客戶版本。
transferData	TransferData	傳輸的資料。
transferMessage	string 長度上限：128	傳輸的訊息。

名稱	類型	描述
rejectReason	string 長度上限：128	傳輸遭拒的原因。
transferDate	timestamp	傳輸的日期。
acceptDate	timestamp	接受傳輸的日期。
rejectDate	timestamp	傳輸遭拒的日期。
generationId	string	憑證的產生 ID。
validity	CertificateValidity	當憑證有效時。
notBefore	timestamp	在此日期之前憑證無效。
notAfter	timestamp	在此日期之後憑證無效。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeDefaultAuthorizer

描述預設授權方。

概要

```
aws iot  describe-default-authorizer  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
}
```

輸出

```
{
  "authorizerDescription": {
    "authorizerName": "string",
    "authorizerArn": "string",
    "authorizerFunctionArn": "string",
    "tokenKeyName": "string",
    "tokenSigningPublicKeys": {
      "string": "string"
    },
    "status": "string",
    "creationDate": "timestamp",
    "lastModifiedDate": "timestamp"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`authorizerDescription`	AuthorizerDescription	預設授權方的描述。
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerArn`	string	授權方 ARN。
`authorizerFunctionArn`	string	授權方 Lambda 函數的 ARN。
`tokenKeyName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	用於自 HTTP 標頭擷取符記的金鑰。
`tokenSigningPublicKeys`	對應	針對從自訂身分驗證服務回傳的符記簽章進行驗證之公開金鑰。
`status`	string	授權方的狀態。 列舉：ACTIVE | INACTIVE
`creationDate`	timestamp	建立授權方時的 UNIX 時間戳記。
`lastModifiedDate`	timestamp	授權方上次更新時的 UNIX 時間戳記。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DescribeEndpoint

回傳此特定 AWS 帳戶進行呼叫的唯一端點。

概要

```
aws iot  describe-endpoint \
    [--endpoint-type <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "endpointType": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`endpointType`	string	此端點類型。有效端點類型包括： • `iot:Data` – 傳回 VeriSign 簽署的資料端點。 • `iot:Data-ATS` – 傳回 ATS 簽署的資料端點。 • `iot:CredentialProvider` – 傳回 AWS IoT 登入資料提供者 API 端點。 • `iot:Jobs` – 傳回 AWS IoT 裝置管理 Jobs API 端點。

輸出

```
{
  "endpointAddress": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
endpointAddress	string	此端點。端點格式如下：identifier.iot.region.amazonaws.com。

錯誤

`InternalFailureException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`UnauthorizedException`

未授權您執行此操作。

`ThrottlingException`

請求頻率超過限制。

DescribeEventConfigurations

描述事件組態。

概要

```
aws iot  describe-event-configurations  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "eventConfigurations": {
    "string": {
      "Enabled": "boolean"
    }
  },
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位

名稱	類型	描述
`eventConfigurations`	對應	事件組態。
`Enabled`	布林值	True 將會啟用此設定。
`creationDate`	timestamp	事件組態的建立日期。
`lastModifiedDate`	timestamp	上次修改事件組態的日期。

錯誤

`InternalFailureException`

出現未預期的錯誤。

`ThrottlingException`

請求頻率超過限制。

DescribeIndex

描述搜尋索引。

概要

```
aws iot  describe-index \
    --index-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "indexName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`indexName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	索引名稱。

輸出

```
{
  "indexName": "string",
  "indexStatus": "string",
  "schema": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`indexName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	索引名稱。
`indexStatus`	string	索引狀態。 列舉：ACTIVE | BUILDING | REBUILDING
`schema`	string	內含指定執行索引之類型的值。有效值為： • REGISTRY - 您的事物索引僅包含了登錄檔資料。 • REGISTRY_AND_SHADOW - 您的事物索引包含了登錄檔資料與陰影資料。 • REGISTRY_AND_CONNECTIVITY_STATUS - 您的事物索引包含了登錄檔資料和事物連線能力狀態資料。 • REGISTRY_AND_SHADOW_AND_CONNECT - 您的事物索引包含了登錄檔資料、陰影資料，和事物連線能力狀態資料。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeJob

描述工作。

概要

```
aws iot  describe-job \
    --job-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

輸出

```
{
  "documentSource": "string",
  "job": {
    "jobArn": "string",
    "jobId": "string",
    "targetSelection": "string",
    "status": "string",
    "forceCanceled": "boolean",
    "reasonCode": "string",
    "comment": "string",
    "targets": [
      "string"
    ],
    "description": "string",
    "presignedUrlConfig": {
      "roleArn": "string",
      "expiresInSec": "long"
    },
    "jobExecutionsRolloutConfig": {
      "maximumPerMinute": "integer",
      "exponentialRate": {
        "baseRatePerMinute": "integer",
        "incrementFactor": "double",
        "rateIncreaseCriteria": {
          "numberOfNotifiedThings": "integer",
          "numberOfSucceededThings": "integer"
        }
      }
    },
    "abortConfig": {
      "criteriaList": [
        {
          "failureType": "string",
          "action": "string",
          "thresholdPercentage": "double",
          "minNumberOfExecutedThings": "integer"
```

```
        }
      ]
    },
    "createdAt": "timestamp",
    "lastUpdatedAt": "timestamp",
    "completedAt": "timestamp",
    "jobProcessDetails": {
      "processingTargets": [
        "string"
      ],
      "numberOfCanceledThings": "integer",
      "numberOfSucceededThings": "integer",
      "numberOfFailedThings": "integer",
      "numberOfRejectedThings": "integer",
      "numberOfQueuedThings": "integer",
      "numberOfInProgressThings": "integer",
      "numberOfRemovedThings": "integer",
      "numberOfTimedOutThings": "integer"
    },
    "timeoutConfig": {
      "inProgressTimeoutInMinutes": "long"
    }
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`documentSource`	string 長度上限：1350；下限：1	工作文件的 S3 連結。
`job`	任務 (Job)	有關工作的資訊。
`jobArn`	string	以 "arn:aws:iot:region:account:job/jobId" 格式識別工作的 ARN。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`targetSelection`	string	指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當代表裝置的物件加入目標群組，工作就會在裝置上運作，即使該工作已被原本就在群組中的所有物件完成。 列舉：CONTINUOUS | SNAPSHOT
`status`	string	任務的狀態，為 `IN_PROGRESS`、`CANCELED`、`DELETION_IN_P` 或 `COMPLETED` 其中一項。

名稱	類型	描述
		enum：IN_PROGRESS | CANCELED | COMPLETED | DELETION_IN_PROGRESS
`forceCanceled`	布林值	若任務取消且選用的 `force` 參數設為 `true`，將為 `true`。
`reasonCode`	string 長度上限：128 模式：[\\p{Upper}p *`Digit`*_]+	如果任務已更新，請提供該更新的原因代碼。
`comment`	string 長度上限：2028 模式：[^\\p{C}]+	如果工作已更新，則會說明更新原因。
`targets`	列出 成員：TargetArn	應收到工作的 IoT 物件及物件群組清單。
`description`	string 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
`presignedUrlConfig`	PresignedUrlConfig	預先簽章的 S3 URL 之組態。
`roleArn`	字串 長度上限：2048；下限：20	授予自工作資料/更新儲存位置 S3 儲存貯體下載檔案的許可之 IAM 角色 ARN。此角色也必須授權讓 IoT 下載檔案。
`expiresInSec`	長整數 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 - 3600，預設值為 3600 秒。當 Jobs 收到對工作文件的 MQTT 請求，就會產生預先簽章的 URL。
`jobExecutionsRolloutConfig`	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。
`maximumPerMinute`	integer 範圍 - 分鐘：1	每分鐘接收待定工作的物件之數量上限。此參數可讓您建立分階段推展的任務。
`exponentialRate`	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
`baseRatePerMinute`	integer 範圍 - 上限：1000；下限：1	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。

名稱	類型	描述
rateIncreaseCriteria	RateIncreaseCriteria	任務啟動增加推展速率的條件。 AWS IoT 最多支援小數點後一位數 (例如，1.5，但不支援 1.55)。
numberOfNotifiedThings	integer 範圍 - 分鐘：1	啟動增加推展速率通知物件數量的閾值。
numberOfSucceededThings	integer 範圍 - 分鐘：1	啟動增加推展速率成功物件數量的閾值。
abortConfig	AbortConfig	用於中止任務的條件組態。
criteriaList	列出 成員：AbortCriteria Java 類別：java.util.List	用於定義中止任務規則的中止條件清單。
failureType	string	用於定義初始化任務中止的任務執行失敗類型。 enum: FAILED | REJECTED | TIMED_OUT | ALL
action	string	用於初始化任務中止的中止動作類型。 enum: CANCEL
minNumberOfExecutedThings	integer 範圍 - 分鐘：1	評估中止規則之前的已執行物件最低數量。
createdAt	timestamp	任務建立的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務最近一次更新的 epoch 時間，以秒為單位。
completedAt	timestamp	任務完成的 epoch 時間，以秒為單位。
jobProcessDetails	JobProcessDetails	關於工作流程的詳細資訊。
processingTargets	列出 成員：ProcessingTargetName Java 類別：java.util.List	任務執行展開時的目標裝置。在任務執行完成展開到所有目標裝置時，此值將為 null。
numberOfCanceledThings	integer	取消工作的物件數量。
numberOfSucceededThings	integer	成功完成工作的物件數量。
numberOfFailedThings	integer	執行工作失敗的物件數量。
numberOfRejectedThings	integer	拒絕工作的物件數量。

名稱	類型	描述
numberOfQueuedThings	integer	等待執行工作的物件數量。
numberOfInProgressThings	integer	現正執行工作的物件數量。
numberOfRemovedThings	integer	因為從當成工作目標的群組中被刪除或移除而不再排定執行工作的物件數量。
numberOfTimedOutThings	integer	任務執行狀態為 TIMED_OUT 的實物數目。
timeoutConfig	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 IN_PROGRESS 時，計時器即會開始。在計時器到期之前，如果任務執行狀態未設定為其他結束狀態，就會自動設為 TIMED_OUT。
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 IN_PROGRESS 狀態超過此間隔時，任務執行就會失敗，並切換到結束 TIMED_OUT 狀態。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

DescribeJobExecution

取得工作執行的詳細資訊。

概要

```
aws iot-jobs-data  describe-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--include-job-document | --no-include-job-document] \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
```

```
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "jobId": "string",
  "thingName": "string",
  "includeJobDocument": "boolean",
  "executionNumber": "long"
}
```

cli-input-json 欄位

名稱	類型	描述
jobId	string 模式：[a-zA-Z0-9_-]+|^$next	此工作在建立時被指派的獨特識別符。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	跟運作工作執行之裝置關聯的物件名稱。
includeJobDocument	布林值	選用。除非設為 false，否則回應會包含任務文件。預設為 true。
executionNumber	長整數	選用。可識別在特定裝置上之特定工作執行的編號。如果未指定，則會傳回最新的工作執行。

輸出

```
{
  "execution": {
    "jobId": "string",
    "thingName": "string",
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "queuedAt": "long",
    "startedAt": "long",
    "lastUpdatedAt": "long",
    "approximateSecondsBeforeTimedOut": "long",
    "versionNumber": "long",
    "executionNumber": "long",
    "jobDocument": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
execution	JobExecution	包含關於工作執行的資料。
jobId	string	您在建立此工作時所指派的獨特識別符。

名稱	類型	描述
	長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。
`status`	string	工作執行的狀態。可以是這些狀態其中之一：「QUEUED」、「IN_PROGRESS」、「FAILED」、「SUCCESS」、「CANCELED」、「TIMED_OUT」、「REJECTED」或「REMOVED」。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`statusDetails`	對應	名稱/值對的集合，能夠描述工作執行的狀態。
`queuedAt`	長整數	任務排入佇列的 epoch 時間，以秒為單位。
`startedAt`	長整數	任務執行開始的 epoch 時間，以秒為單位。
`lastUpdatedAt`	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
`approximateSecondsBeforeTimedOut`	長整數	將任務執行狀態變更為 `TIMED_OUT` 之前要保留狀態的預估秒數。實際任務執行逾時可能會比預估持續時間最多晚 60 秒發生。
`versionNumber`	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
`executionNumber`	長整數	可識別在特定裝置上之特定工作執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
`jobDocument`	string 長度上限：32768	工作文件的內容。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

`CertificateValidationException`

憑證無效。

`TerminalStateException`

工作已經處於結束狀態。

DescribeJobExecution

描述工作執行。

概要

```
aws iot  describe-job-execution \
    --job-id <value> \
    --thing-name <value> \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
  "thingName": "string",
  "executionNumber": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	運作工作執行的物件名稱。

名稱	類型	描述
executionNumber	長整數	用於指定特定裝置執行之特定工作的字串 (由數字「0」至「9」組成)。

輸出

```
{
  "execution": {
    "jobId": "string",
    "status": "string",
    "forceCanceled": "boolean",
    "statusDetails": {
      "detailsMap": {
        "string": "string"
      }
    },
    "thingArn": "string",
    "queuedAt": "timestamp",
    "startedAt": "timestamp",
    "lastUpdatedAt": "timestamp",
    "executionNumber": "long",
    "versionNumber": "long",
    "approximateSecondsBeforeTimedOut": "long"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
execution	JobExecution	有關工作執行的資訊。
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
status	string	任務執行的狀態 (IN_PROGRESS、QUEUED、FAILED、SUCCE 或 REJECTED)。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
forceCanceled	布林值	若任務執行取消且選用的 force 參數設為 true，將為 true。
statusDetails	JobExecutionStatusDetails	名稱/值對的集合，能夠描述工作執行的狀態。
detailsMap	對應	工作執行狀態。
thingArn	string	運作工作執行的物件 ARN。

名稱	類型	描述
queuedAt	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	timestamp	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
executionNumber	長整數	用於辨識此特定裝置執行之特定工作的字串 (由數字「0」至「9」組成)。這可於命令中使用，以傳回或更新工作執行的資訊。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
approximateSecondsBeforeTimedOut	長整數	將任務執行狀態變更為 TIMED_OUT 之前要保留狀態的預估秒數。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。實際任務執行逾時可能會比預估持續時間最多晚 60 秒發生。如果任務執行已經達到結束狀態，就不會包含這個值。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

DescribeMitigationAction

取得有關緩解動作的資訊。

概要

```
aws iot  describe-mitigation-action \
    --action-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "actionName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
actionName	string 長度上限：128 模式：[a-zA-Z0-9_-]+	唯一識別緩解行動易記的名稱。

輸出

```
{
  "actionName": "string",
  "actionType": "string",
  "actionArn": "string",
  "actionId": "string",
  "roleArn": "string",
  "actionParams": {
    "updateDeviceCertificateParams": {
      "action": "string"
    },
    "updateCACertificateParams": {
      "action": "string"
    },
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "string"
      ],
      "overrideDynamicGroups": "boolean"
    },
    "replaceDefaultPolicyVersionParams": {
      "templateName": "string"
    },
    "enableIoTLoggingParams": {
      "roleArnForLogging": "string",
      "logLevel": "string"
    },
    "publishFindingToSnsParams": {
      "topicArn": "string"
    }
  },
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位

名稱	類型	描述
actionName	string 長度上限：128 模式：[a-zA-Z0-9_-]+	唯一識別緩解行動易記的名稱。
actionType	string	緩解動作的類型。

名稱	類型	描述
		列舉：UPDATE_DEVICE_CERTIFICATE | UPDATE_CA_CERTIFICATE | ADD_THINGS_TO_THING_GROUP | REPLACE_DEFAULT_POLICY_VERSION | ENABLE_IOT_LOGGING | PUBLISH_FINDING_TO_SNS
`actionArn`	string	識別此遷移動作的 ARN。
`actionId`	string	此規則的唯一識別符。
`roleArn`	字串 長度上限：2048；下限：20	用於套用此動作的 IAM 角色的 ARN。
`actionParams`	MitigationActionParams	控制如何套用緩解動作的參數，緩解動作類型特定。
`updateDeviceCertificateParams`	UpdateDeviceCertificateParams	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
`action`	string	您要套用到裝置憑證的動作。唯一支援的值為 `DEACTIVATE`。 列舉：DEACTIVATE
`updateCACertificateParams`	UpdateCACertificateParams	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。
`action`	string	您要套用到 CA 憑證的動作。唯一支援的值為 `DEACTIVATE`。 列舉：DEACTIVATE
`addThingsToThingGroupParams`	AddThingsToThingGroupParams	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
`thingGroupNames`	列出 成員：ThingGroupName	觸發緩解行動新增物件的群組清單。您可以新增物件最多到 10 個群組，但是在相同的層級內，您無法讓一個物件加入超過一個群組。
`overrideDynamicGroups`	布林值	指定此緩解動作是否可以移動觸發緩解動作的實物，即使該實物為一或多個動態物件群組的一部分。
`replaceDefaultPolicyVersionParams`	ReplaceDefaultPolicyVersionParams	參數定義緩解行動，新增空白政策來限制權限。
`templateName`	字串	要套用範本的名稱。唯一支援的值為 `BLANK_POLICY`。 列舉：BLANK_POLICY

名稱	類型	描述
`enableIoTLoggingParams`	EnableIoTLoggingParams	定義可以指定的細節層級啟用 AWS IoT 記錄的緩解動作的參數。
`roleArnForLogging`	字串 長度上限：2048；下限：20	用於記錄 IAM 角色的 ARN。
`logLevel`	string	指定要記錄的資訊類型。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
`publishFindingToSnsParams`	PublishFindingToSnsParams	用來定義發佈結果到 Amazon SNS 緩解動作的參數。您可以實作自訂的動作來回應 Amazon SNS 訊息。
`topicArn`	string 長度上限：350	您要發佈結果主題的 ARN。
`creationDate`	timestamp	將緩解動作新增到您的 AWS 帳戶的日期和時間。
`lastModifiedDate`	timestamp	上次變更緩解動作的日期和時間。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeRoleAlias

描述角色別名。

概要

```
aws iot  describe-role-alias \
    --role-alias <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleAlias": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`roleAlias`	string 長度上限：128；下限：1 模式：[w=,@-]+	欲描述的角色別名。

輸出

```
{
  "roleAliasDescription": {
    "roleAlias": "string",
    "roleAliasArn": "string",
    "roleArn": "string",
    "owner": "string",
    "credentialDurationSeconds": "integer",
    "creationDate": "timestamp",
    "lastModifiedDate": "timestamp"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`roleAliasDescription`	RoleAliasDescription	角色別名描述。
`roleAlias`	string 長度上限：128；下限：1 模式：[w=,@-]+	角色別名。
`roleAliasArn`	string	此角色別名的 ARN。
`roleArn`	字串 長度上限：2048；下限：20	角色 ARN。
`owner`	string 長度上限：12；下限：12 模式：[0-9]+	角色別名的擁有者。
`credentialDurationSeconds`	integer 範圍 - 上限：3600；下限：900	登入資料有效秒數。
`creationDate`	timestamp	建立角色別名時的 UNIX 時間戳記。

名稱	類型	描述
lastModifiedDate	timestamp	上次修改角色別名的 UNIX 時間戳記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeScheduledAudit

取得有關已排程的稽核的資訊。

概要

```
aws iot  describe-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
scheduledAuditName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	您需要取得其資訊的已排程稽核的名稱。

輸出

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string",
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`frequency`	string	已排程的稽核執行的頻率。「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
`dayOfMonth`	string 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程稽核的日期。將會是「1」到「31」或「最後一天」。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
`dayOfWeek`	string	當週執行已排程稽核的日期。「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
`targetCheckNames`	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括已啟用的檢查，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。)
`scheduledAuditName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱。
`scheduledAuditArn`	string	已排程的稽核的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DescribeSecurityProfile

取得有關 Device Defender 的安全性描述檔的資訊。

概要

```
aws iot  describe-security-profile \
    --security-profile-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您需要取得其資訊的安全性描述檔的名稱。

輸出

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
```

```
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "version": "long",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	安全性描述檔的名稱。
`securityProfileArn`	字串	安全性描述檔的 ARN。
`securityProfileDescription`	string 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的描述 (與安全性描述檔建立或更新時間相關聯)。
`behaviors`	列出 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
`name`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
`metric`	string	行為測量的內容。
`criteria`	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
`comparisonOperator`	string	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。

名稱	類型	描述
		列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
`value`	MetricValue	相較於 `metric` 的值。
`count`	長整數 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
`cidrs`	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
`ports`	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
`durationSeconds`	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
`consecutiveDatapointsToAlarm`	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
`consecutiveDatapointsToClear`	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
`statisticalThreshold`	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
`statistic`	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。

名稱	類型	描述
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。
alertTargetArn	string	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	列出 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 behaviors 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
version	長整數	安全性描述檔的版本。只要更新安全性描述檔便產生新版本。
creationDate	timestamp	安全性描述檔的建立時間。
lastModifiedDate	timestamp	安全性描述檔的最後建立時間。

錯誤

InvalidRequestException

請求的內容無效。

ResourceNotFoundException

指定的資源不存在。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

DescribeStream

取得有關串流的資訊。

概要

```
aws iot  describe-stream \
    --stream-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
```

```
  "streamId": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。

輸出

```
{
  "streamInfo": {
    "streamId": "string",
    "streamArn": "string",
    "streamVersion": "integer",
    "description": "string",
    "files": [
      {
        "fileId": "integer",
        "s3Location": {
          "bucket": "string",
          "key": "string",
          "version": "string"
        }
      }
    ],
    "createdAt": "timestamp",
    "lastUpdatedAt": "timestamp",
    "roleArn": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
streamInfo	StreamInfo	有關串流的資訊。
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
streamArn	string	串流 ARN。
streamVersion	integer 範圍 - 上限：65535；下限：0	此串流的版本。
description	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。

名稱	類型	描述
`files`	列出 成員：StreamFile	欲串流的檔案。
`fileId`	integer 範圍 - 上限：255；下限：0	檔案 ID。
`s3Location`	S3Location	檔案於 S3 中的位置。
`bucket`	string 長度下限：1	S3 儲存貯體。
`key`	string 長度下限：1	S3 金鑰。
`version`	string	S3 儲存貯體版本。
`createdAt`	timestamp	建立串流的日期。
`lastUpdatedAt`	timestamp	上次更新串流的日期。
`roleArn`	字串 長度上限：2048；下限：20	AWS IoT 為了存取您的 S3 檔案而擔任的 IAM 角色。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DescribeThing

取得有關指定物件的資訊。

概要

```
aws iot  describe-thing \
    --thing-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。

輸出

```
{
  "defaultClientId": "string",
  "thingName": "string",
  "thingId": "string",
  "thingArn": "string",
  "thingTypeName": "string",
  "attributes": {
    "string": "string"
  },
  "version": "long",
  "billingGroupName": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`defaultClientId`	string	預設用戶端 ID。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`thingId`	string	欲描述的物件 ID。
`thingArn`	string	欲描述的物件 ARN。
`thingTypeName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
`attributes`	對應	物件屬性。

名稱	類型	描述
`version`	長整數	登錄檔內物件記錄的目前版本。 Note 為了避免無意變更登錄檔中的資訊，您可於 `expectedVersion`和 `UpdateThing` 呼叫的 `DeleteThing` 參數中，傳入版本資訊。
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	該物件所屬的帳單群組名稱。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DescribeThingGroup

描述物件群組。

概要

```
aws iot  describe-thing-group \
    --thing-group-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件群組名稱。

輸出

```
{
  "thingGroupName": "string",
  "thingGroupId": "string",
  "thingGroupArn": "string",
  "version": "long",
  "thingGroupProperties": {
    "thingGroupDescription": "string",
    "attributePayload": {
      "attributes": {
        "string": "string"
      },
      "merge": "boolean"
    }
  },
  "thingGroupMetadata": {
    "parentGroupName": "string",
    "rootToParentThingGroups": [
      {
        "groupName": "string",
        "groupArn": "string"
      }
    ],
    "creationDate": "timestamp"
  },
  "indexName": "string",
  "queryString": "string",
  "queryVersion": "string",
  "status": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件群組名稱。
thingGroupId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	物件群組 ID。
thingGroupArn	string	物件群組 ARN。
version	長整數	物件群組版本。

名稱	類型	描述
`thingGroupProperties`	ThingGroupProperties	物件群組屬性。
`thingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
`attributePayload`	AttributePayload	JSON 格式的物件群組屬性。
`attributes`	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： `\"attributes\": {\"string1\": \"string2\"}`
`merge`	布林值	指定 `AttributePayload`提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 `UpdateThing` 加上一個空白屬性值。 Note `merge` 屬性僅於呼叫 `UpdateThing` 或 `UpdateThingGroup` 時有效。
`thingGroupMetadata`	ThingGroupMetadata	物件群組中繼資料。
`parentGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	父物件群組名稱。
`rootToParentThingGroups`	列出 成員：GroupNameAndArn Java 類別：java.util.List	根父物件群組。
`groupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組名稱。
`groupArn`	string	群組 ARN。
`creationDate`	timestamp	建立物件群組時的 UNIX 時間戳記。

名稱	類型	描述
indexName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	動態物件群組索引名稱。
queryString	string 長度下限：1	動態物件群組搜尋查詢字串。
queryVersion	string	動態物件群組查詢版本。
status	string	動態物件群組狀態。 列舉：ACTIVE | BUILDING | REBUILDING

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeThingRegistrationTask

描述大量物件佈建任務。

概要

```
aws iot  describe-thing-registration-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
taskId	string	任務 ID。

名稱	類型	描述
	長度上限：40	

輸出

```
{
  "taskId": "string",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp",
  "templateBody": "string",
  "inputFileBucket": "string",
  "inputFileKey": "string",
  "roleArn": "string",
  "status": "string",
  "message": "string",
  "successCount": "integer",
  "failureCount": "integer",
  "percentageProgress": "integer"
}
```

CLI 輸出欄位

名稱	類型	描述
`taskId`	string 長度上限：40	任務 ID。
`creationDate`	timestamp	任務建立日期。
`lastModifiedDate`	timestamp	上次修改任務的日期。
`templateBody`	string	任務的範本。
`inputFileBucket`	string 長度上限：256；下限：3 模式：[a-zA-Z0-9._-]+	內含輸入檔案的 S3 儲存貯體。
`inputFileKey`	string 長度上限：1024；下限：1 模式：[a-zA-Z0-9!_.*'()-/]+	輸入檔案金鑰。
`roleArn`	字串 長度上限：2048；下限：20	授予輸入檔案儲存貯體存取權限的角色 ARN。
`status`	string	大量物件佈建任務的狀態。 列舉：InProgress | Completed | Failed | Cancelled | Cancelling
`message`	字串 長度上限：2048	訊息。
`successCount`	integer	佈建成功的物件數量。

名稱	類型	描述
failureCount	integer	佈建失敗的物件數量。
percentageProgress	integer 範圍 - 上限：100；下限：0	大量佈建任務的進度 (以百分比表示)。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

DescribeThingType

取得有關指定物件類型的資訊。

概要

```
aws iot  describe-thing-type \
    --thing-type-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingTypeName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。

輸出

```
{
  "thingTypeName": "string",
  "thingTypeId": "string",
  "thingTypeArn": "string",
  "thingTypeProperties": {
    "thingTypeDescription": "string",
    "searchableAttributes": [
      "string"
    ]
  },
  "thingTypeMetadata": {
    "deprecated": "boolean",
    "deprecationDate": "timestamp",
    "creationDate": "timestamp"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
thingTypeId	string	物件類型 ID。
thingTypeArn	string	物件類型 ARN。
thingTypeProperties	ThingTypeProperties	ThingTypeProperties 內含物件類型的資訊，包括描述和可搜尋物件屬性名稱清單。
thingTypeDescription	string 長度上限：2028 模式：[\\p{Graph}]*	物件類型描述。
searchableAttributes	列出 成員：AttributeName Java 類別：java.util.List	可搜尋物件屬性名稱清單。
thingTypeMetadata	ThingTypeMetadata	ThingTypeMetadata 內含物件類型的其他資訊，包括建立日期和時間、表示物件類型是否已棄用的值，以及棄用的日期和時間。
deprecated	布林值	指定是否棄用物件類型。若此值為 true，新物件將無法與此類型建立關聯。
deprecationDate	timestamp	棄用物件類型的日期和時間。
creationDate	timestamp	建立物件類型的日期和時間。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DetachPolicy

將政策從指定目標分離。

概要

```
aws iot  detach-policy \
    --policy-name <value> \
    --target <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "target": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	欲分離的政策。
`target`	string	欲將政策分離的目標。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

DetachPrincipalPolicy

自指定憑證移除特定政策。

備註：此 API 已作廢。請改用 DetachPolicy。

概要

```
aws iot  detach-principal-policy \
    --policy-name <value> \
    --principal <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "principal": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	欲分離的政策名稱。
`principal`	string	委託人。 若委託人是憑證，請指定憑證 ARN。若委託人是 Amazon Cognito Identity，請指定身分 ID。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DetachSecurityProfile

將 Device Defender 安全性描述檔與實物群組或此帳戶的關聯取消。

概要

```
aws iot  detach-security-profile \
    --security-profile-name <value> \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "securityProfileTargetArn": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	分離的安全性描述檔。
`securityProfileTargetArn`	string	安全性描述檔與之分離的實物群組的 ARN。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

DetachThingPrincipal

將指定委託人自指定物件分離。委託人可以是 X.509 憑證、IAM 使用者、群組和角色、Amazon Cognito 身分或聯合身分。

> **Note**
>
> 此為非同步呼叫。這可能需要花費幾秒鐘時間讓分離進行傳播。

概要

```
aws iot  detach-thing-principal \
    --thing-name <value> \
    --principal <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "principal": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`principal`	string	若委託人是憑證，此值必須為憑證 ARN。若委託人是 Amazon Cognito Identity，此值必須為 Amazon Cognito Identity 的 ID。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

DisableTopicRule

停用規則。

概要

```
aws iot  disable-topic-rule \
    --rule-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "ruleName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`ruleName`	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	欲停用的規則名稱。

輸出

無

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

`UnauthorizedException`

未授權您執行此操作。

`ConflictingResourceUpdateException`

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

EnableTopicRule

啟用規則。

概要

```
aws iot  enable-topic-rule \
    --rule-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "ruleName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`ruleName`	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	欲啟用的主題規則名稱。

輸出

無

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

`UnauthorizedException`

未授權您執行此操作。

`ConflictingResourceUpdateException`

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

GetEffectivePolicies

取得指定裝置行為在連接到 AWS IoT 裝置閘道時對其授權行為發生效用的政策清單。

概要

```
aws iot  get-effective-policies \
    [--principal <value>] \
    [--cognito-identity-pool-id <value>] \
    [--thing-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "principal": "string",
  "cognitoIdentityPoolId": "string",
  "thingName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`principal`	string	委託人。
`cognitoIdentityPoolId`	string	Cognito Identity Pool 的 ID。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。

輸出

```
{
```

```
  "effectivePolicies": [
    {
      "policyName": "string",
      "policyArn": "string",
      "policyDocument": "string"
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
effectivePolicies	列出 成員：EffectivePolicy Java 類別：java.util.List	有效政策。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
policyDocument	string	IAM 政策文件。

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

LimitExceededException

已超出限制。

GetIndexingConfiguration

取得搜尋組態。

概要

```
aws iot  get-indexing-configuration  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "thingIndexingConfiguration": {
    "thingIndexingMode": "string",
    "thingConnectivityIndexingMode": "string"
  },
  "thingGroupIndexingConfiguration": {
    "thingGroupIndexingMode": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`thingIndexingConfiguration`	ThingIndexingConfiguration	物件索引組態。
`thingIndexingMode`	string	物件索引模式。有效值為： • REGISTRY - 您的事物索引僅包含登錄檔資料。 • REGISTRY_AND_SHADOW - 您的事物索引包含登錄檔和陰影資料。 • OFF - 物件索引已停用。 列舉：OFF | REGISTRY | REGISTRY_AND_SHADOW
`thingConnectivityIndexingMode`	string	物件連線索引模式。有效值為： • STATUS - 您的事物索引包含連線能力狀態。若要啟用事物連線能力索引建立，則 thingIndexMode 必不得設為 OFF。 • OFF - 物件連線狀態索引已停用。 enum: OFF | STATUS
`thingGroupIndexingConfiguration`	ThingGroupIndexingConfiguration	索引組態。
`thingGroupIndexingMode`	string	實物群組索引模式。

名稱	類型	描述
		列舉：OFF | ON

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

GetJobDocument

取得工作文件。

概要

```
aws iot  get-job-document \
    --job-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。

輸出

```
{
  "document": "string"
```

```
}
```

CLI 輸出欄位

名稱	類型	描述
`document`	string 長度上限：32768	工作文件的內容。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

GetLoggingOptions

取得記錄選項。

注意：不建議使用此命令。請改用 `GetV2LoggingOptions`。

概要

```
aws iot  get-logging-options  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "roleArn": "string",
  "logLevel": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`roleArn`	string	授予存取權限的 IAM 角色 ARN。

名稱	類型	描述
logLevel	string	記錄層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

GetOTAUpdate

取得 OTA 更新。

概要

```
aws iot  get-ota-update \
    --ota-update-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "otaUpdateId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
otaUpdateId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	OTA 更新 ID。

輸出

```
{
  "otaUpdateInfo": {
    "otaUpdateId": "string",
    "otaUpdateArn": "string",
    "creationDate": "timestamp",
    "lastModifiedDate": "timestamp",
```

```
    "description": "string",
    "targets": [
      "string"
    ],
    "awsJobExecutionsRolloutConfig": {
      "maximumPerMinute": "integer"
    },
    "targetSelection": "string",
    "otaUpdateFiles": [
      {
        "fileName": "string",
        "fileVersion": "string",
        "fileLocation": {
          "stream": {
            "streamId": "string",
            "fileId": "integer"
          },
          "s3Location": {
            "bucket": "string",
            "key": "string",
            "version": "string"
          }
        },
        "codeSigning": {
          "awsSignerJobId": "string",
          "startSigningJobParameter": {
            "signingProfileParameter": {
              "certificateArn": "string",
              "platform": "string",
              "certificatePathOnDevice": "string"
            },
            "signingProfileName": "string",
            "destination": {
              "s3Destination": {
                "bucket": "string",
                "prefix": "string"
              }
            }
          },
          "customCodeSigning": {
            "signature": {
              "inlineDocument": "blob"
            },
            "certificateChain": {
              "certificateName": "string",
              "inlineDocument": "string"
            },
            "hashAlgorithm": "string",
            "signatureAlgorithm": "string"
          }
        },
        "attributes": {
          "string": "string"
        }
      }
    ],
    "otaUpdateStatus": "string",
    "awsIotJobId": "string",
    "awsIotJobArn": "string",
    "errorInfo": {
      "code": "string",
      "message": "string"
    },
    "additionalParameters": {
      "string": "string"
    }
```

```
  }
}
```

CLI 輸出欄位

名稱	類型	描述
otaUpdateInfo	OTAUpdateInfo	OTA 更新資訊。
otaUpdateId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	OTA 更新 ID。
otaUpdateArn	string	OTA 更新 ARN。
creationDate	timestamp	建立 OTA 更新的日期。
lastModifiedDate	timestamp	上次更新 OTA 更新的日期。
description	string 長度上限：2028 模式：[^\\p{C}]+	OTA 更新的描述。
targets	列出 成員：Target	OTA 更新的目標。
awsJobExecutionsRolloutConfig	AwsJobExecutionsRolloutConfig	OTA 更新的發行組態。
maximumPerMinute	integer 範圍 - 上限：1000；下限：1	每分鐘可啟動的 OTA 更新任務執行數上限。
targetSelection	string	指定 OTA 更新是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成 OTA 更新後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，OTA 更新也可能會運作於物件上。例如，當物件新增至目標群組，OTA 更新就會在物件上運作，即使該 OTA 更新已被原本就在群組中的所有物件完成。 列舉：CONTINUOUS | SNAPSHOT
otaUpdateFiles	列出 成員：OTAUpdateFile	與 OTA 更新建立關聯的檔案清單。
fileName	string	檔案名稱。
fileVersion	string	檔案版本。
fileLocation	FileLocation	已更新韌體的位置。

名稱	類型	描述
stream	串流	包含 OTA 更新的串流。
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
fileId	integer 範圍 - 上限：255；下限：0	與串流建立關聯的檔案 ID。
s3Location	S3Location	S3 中已更新韌體的位置。
bucket	string 長度下限：1	S3 儲存貯體。
key	string 長度下限：1	S3 金鑰。
version	string	S3 儲存貯體版本。
codeSigning	CodeSigning	檔案的代碼簽署方式。
awsSignerJobId	string	用於簽署檔案所建立的 AWSSignerJob ID。
startSigningJobParameter	StartSigningJobParameter	描述程式碼簽署任務。
signingProfileParameter	SigningProfileParameter	描述程式碼簽署描述檔。
certificateArn	string	憑證 ARN。
platform	string	您裝置的硬體平台。
certificatePathOnDevice	string	程式碼簽署憑證在您裝置上的位置。
signingProfileName	string	程式碼簽署描述檔名稱。
destination	目的地	撰寫程式碼簽章檔案的位置。
s3Destination	S3Destination	描述 S3 中已更新韌體的位置。
bucket	string 長度下限：1	內含已更新韌體的 S3 儲存貯體。
prefix	string	S3 字首。
customCodeSigning	CustomCodeSigning	代碼簽署檔案的自訂方式。
signature	CodeSigningSignature	檔案的簽章。
inlineDocument	blob	以 base64 編碼呈現的二進位代碼簽署簽章。
certificateChain	CodeSigningCertificateChain	憑證鏈。

名稱	類型	描述
certificateName	string	憑證名稱。
inlineDocument	string	以 base64 編碼呈現的二進位代碼簽署憑證鏈。
hashAlgorithm	string	用於以代碼簽署檔案的雜湊演算法。
signatureAlgorithm	string	用於以代碼簽署檔案的簽章演算法。
attributes	對應	名稱/屬性對清單。
otaUpdateStatus	string	OTA 更新的狀態。 列舉：CREATE_PENDING | CREATE_IN_PROGRESS | CREATE_COMPLETE | CREATE_FAILED
awsIotJobId	string	與 OTA 更新建立關聯的 AWS IoT 工作 ID。
awsIotJobArn	string	與 OTA 更新建立關聯的 AWS IoT 工作 ARN。
errorInfo	ErrorInfo	與 OTA 更新建立關聯的錯誤資訊。
code	string	錯誤代碼。
message	string	錯誤訊息。
additionalParameters	對應	名稱/值對的集合。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ServiceUnavailableException`

此服務暫時無法使用。

`ResourceNotFoundException`

指定的資源不存在。

GetPendingJobExecutions

為物件取得所有未處於結束狀態之工作的清單。

概要

```
aws iot-jobs-data  get-pending-job-executions \
    --thing-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。

輸出

```
{
  "inProgressJobs": [
    {
      "jobId": "string",
      "queuedAt": "long",
      "startedAt": "long",
      "lastUpdatedAt": "long",
      "versionNumber": "long",
      "executionNumber": "long"
    }
  ],
  "queuedJobs": [
    {
      "jobId": "string",
      "queuedAt": "long",
      "startedAt": "long",
      "lastUpdatedAt": "long",
      "versionNumber": "long",
      "executionNumber": "long"
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
`inProgressJobs`	列出 成員：JobExecutionSummary	狀態為 IN_PROGRESS 的 JobExecutionSummary 物件清單。

名稱	類型	描述
	Java 類別：java.util.List	
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`queuedAt`	長整數	任務排入佇列的 epoch 時間，以秒為單位。
`startedAt`	長整數	任務執行開始的 epoch 時間，以秒為單位。
`lastUpdatedAt`	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
`versionNumber`	長整數	工作執行的版本。每次 AWS IoT Jobs 從裝置收到更新時，工作執行的版本編號就會增加。
`executionNumber`	長整數	可識別在特定裝置上之特定工作執行的編號。
`queuedJobs`	列出 成員：JobExecutionSummary Java 類別：java.util.List	狀態為 QUEUED 的 JobExecutionSummary 物件清單。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`queuedAt`	長整數	任務排入佇列的 epoch 時間，以秒為單位。
`startedAt`	長整數	任務執行開始的 epoch 時間，以秒為單位。
`lastUpdatedAt`	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
`versionNumber`	長整數	工作執行的版本。每次 AWS IoT Jobs 從裝置收到更新時，工作執行的版本編號就會增加。
`executionNumber`	長整數	可識別在特定裝置上之特定工作執行的編號。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

`CertificateValidationException`

憑證無效。

GetPolicy

取得有關指定政策及其預設版本政策文件的資訊。

概要

```
aws iot  get-policy \
    --policy-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。

輸出

```
{
  "policyName": "string",
  "policyArn": "string",
  "policyDocument": "string",
  "defaultVersionId": "string",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp",
  "generationId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`policyName`	string	政策名稱。

名稱	類型	描述
	長度上限：128；下限：1 模式：[w+=,.@-]+	
policyArn	string	政策 ARN。
policyDocument	string	描述政策的 JSON 文件。
defaultVersionId	string 模式：[0-9]+	預設政策版本 ID。
creationDate	timestamp	建立政策的日期。
lastModifiedDate	timestamp	上次修改政策的日期。
generationId	string	政策的產生 ID。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

GetPolicyVersion

取得有關指定政策版本的資訊。

概要

```
aws iot  get-policy-version \
    --policy-name <value> \
    --policy-version-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
  "policyName": "string",
  "policyVersionId": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	string 模式：[0-9]+	政策版本 ID。

輸出

```
{
  "policyArn": "string",
  "policyName": "string",
  "policyDocument": "string",
  "policyVersionId": "string",
  "isDefaultVersion": "boolean",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp",
  "generationId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
policyArn	string	政策 ARN。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyDocument	string	描述政策的 JSON 文件。
policyVersionId	string 模式：[0-9]+	政策版本 ID。
isDefaultVersion	布林值	指定該政策版本是否為預設版本。
creationDate	timestamp	建立政策版本的日期。
lastModifiedDate	timestamp	上次修改政策版本的日期。
generationId	string	政策版本的產生 ID。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

GetRegistrationCode

取得向 AWS IoT 註冊 CA 憑證所使用的註冊碼。

概要

```
aws iot  get-registration-code  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "registrationCode": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`registrationCode`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證註冊碼。

錯誤

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

GetStatistics

取得符合特定查詢的事物的統計資料。

概要

```
aws iot  get-statistics \
    [--index-name <value>] \
    --query-string <value> \
    [--aggregation-field <value>] \
    [--query-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "indexName": "string",
  "queryString": "string",
  "aggregationField": "string",
  "queryVersion": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`indexName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	要搜尋的索引名稱。預設值為 `AWS_Things`。
`queryString`	string 長度下限：1	用於搜尋的查詢。您可指定查詢字串的「*」，以取得在您 AWS 帳戶中所有已建立索引的事物數目。

名稱	類型	描述
aggregationField	string 長度下限：1	彙總欄位名稱。目前尚未支援。
queryVersion	string	用於搜尋的查詢版本。

輸出

```
{
  "statistics": {
    "count": "integer"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
statistics	統計資料	由機群索引建立服務根據查詢和彙總欄位所傳回的統計資料。
count	integer	符合查詢的事物計數。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`InvalidQueryException`

查詢無效。

`InvalidAggregationException`

該彙總無效。

`IndexNotReadyException`

索引尚未就緒。

GetThingShadow

取得特定物件的影子。

如需詳細資訊，請參閱《AWS IoT 開發人員指南》中的 GetThingShadow。

概要

```
aws iot-data  get-thing-shadow \
    --thing-name <value> \
    [--shadow-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "shadowName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`shadowName`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9:_-]+	影子的名稱。

輸出

```
{
  "payload": "blob"
}
```

CLI 輸出欄位

名稱	類型	描述
`payload`	blob	JSON 格式的狀態資訊。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`MethodNotAllowedException`

不支援所指定的 HTTP 動詞和 URI 組合。

`UnsupportedDocumentEncodingException`

不支援該編碼。

GetTopicRule

取得有關規則的資訊。

概要

```
aws iot  get-topic-rule \
    --rule-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "ruleName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`ruleName`	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。

輸出

```
{
  "ruleArn": "string",
```

```
"rule": {
  "ruleName": "string",
  "sql": "string",
  "description": "string",
  "createdAt": "timestamp",
  "actions": [
    {
      "dynamoDB": {
        "tableName": "string",
        "roleArn": "string",
        "operation": "string",
        "hashKeyField": "string",
        "hashKeyValue": "string",
        "hashKeyType": "string",
        "rangeKeyField": "string",
        "rangeKeyValue": "string",
        "rangeKeyType": "string",
        "payloadField": "string"
      },
      "dynamoDBv2": {
        "roleArn": "string",
        "putItem": {
          "tableName": "string"
        }
      },
      "lambda": {
        "functionArn": "string"
      },
      "sns": {
        "targetArn": "string",
        "roleArn": "string",
        "messageFormat": "string"
      },
      "sqs": {
        "roleArn": "string",
        "queueUrl": "string",
        "useBase64": "boolean"
      },
      "kinesis": {
        "roleArn": "string",
        "streamName": "string",
        "partitionKey": "string"
      },
      "republish": {
        "roleArn": "string",
        "topic": "string",
        "qos": "integer"
      },
      "s3": {
        "roleArn": "string",
        "bucketName": "string",
        "key": "string",
        "cannedAcl": "string"
      },
      "firehose": {
        "roleArn": "string",
        "deliveryStreamName": "string",
        "separator": "string"
      },
      "cloudwatchMetric": {
        "roleArn": "string",
        "metricNamespace": "string",
        "metricName": "string",
        "metricValue": "string",
        "metricUnit": "string",
        "metricTimestamp": "string"
```

```
        },
        "cloudwatchAlarm": {
          "roleArn": "string",
          "alarmName": "string",
          "stateReason": "string",
          "stateValue": "string"
        },
        "elasticsearch": {
          "roleArn": "string",
          "endpoint": "string",
          "index": "string",
          "type": "string",
          "id": "string"
        },
        "salesforce": {
          "token": "string",
          "url": "string"
        },
        "iotAnalytics": {
          "channelArn": "string",
          "channelName": "string",
          "roleArn": "string"
        },
        "iotEvents": {
          "inputName": "string",
          "messageId": "string",
          "roleArn": "string"
        },
        "stepFunctions": {
          "executionNamePrefix": "string",
          "stateMachineName": "string",
          "roleArn": "string"
        }
      }
    ],
    "ruleDisabled": "boolean",
    "awsIotSqlVersion": "string",
    "errorAction": {
      "dynamoDB": {
        "tableName": "string",
        "roleArn": "string",
        "operation": "string",
        "hashKeyField": "string",
        "hashKeyValue": "string",
        "hashKeyType": "string",
        "rangeKeyField": "string",
        "rangeKeyValue": "string",
        "rangeKeyType": "string",
        "payloadField": "string"
      },
      "dynamoDBv2": {
        "roleArn": "string",
        "putItem": {
          "tableName": "string"
        }
      },
      "lambda": {
        "functionArn": "string"
      },
      "sns": {
        "targetArn": "string",
        "roleArn": "string",
        "messageFormat": "string"
      },
      "sqs": {
        "roleArn": "string",
```

```
        "queueUrl": "string",
        "useBase64": "boolean"
      },
      "kinesis": {
        "roleArn": "string",
        "streamName": "string",
        "partitionKey": "string"
      },
      "republish": {
        "roleArn": "string",
        "topic": "string",
        "qos": "integer"
      },
      "s3": {
        "roleArn": "string",
        "bucketName": "string",
        "key": "string",
        "cannedAcl": "string"
      },
      "firehose": {
        "roleArn": "string",
        "deliveryStreamName": "string",
        "separator": "string"
      },
      "cloudwatchMetric": {
        "roleArn": "string",
        "metricNamespace": "string",
        "metricName": "string",
        "metricValue": "string",
        "metricUnit": "string",
        "metricTimestamp": "string"
      },
      "cloudwatchAlarm": {
        "roleArn": "string",
        "alarmName": "string",
        "stateReason": "string",
        "stateValue": "string"
      },
      "elasticsearch": {
        "roleArn": "string",
        "endpoint": "string",
        "index": "string",
        "type": "string",
        "id": "string"
      },
      "salesforce": {
        "token": "string",
        "url": "string"
      },
      "iotAnalytics": {
        "channelArn": "string",
        "channelName": "string",
        "roleArn": "string"
      },
      "iotEvents": {
        "inputName": "string",
        "messageId": "string",
        "roleArn": "string"
      },
      "stepFunctions": {
        "executionNamePrefix": "string",
        "stateMachineName": "string",
        "roleArn": "string"
      }
    }
  }
```

```
}
```

CLI 輸出欄位

名稱	類型	描述
`ruleArn`	string	規則 ARN。
`rule`	TopicRule	規則。
`ruleName`	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。
`sql`	string	用於查詢主題的 SQL 陳述式。若使用多行的 SQL 查詢，請務必逸出換行字元。
`description`	string	規則的描述。
`createdAt`	timestamp	建立規則的日期和時間。
`actions`	列出 成員：Action	與規則建立關聯的動作。
`dynamoDB`	DynamoDBAction	寫入至 DynamoDB 資料表。
`tableName`	string	DynamoDB 資料表的名稱。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`operation`	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。
`hashKeyField`	string	雜湊索引鍵名稱。
`hashKeyValue`	string	雜湊索引鍵值。
`hashKeyType`	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`rangeKeyField`	string	範圍索引鍵名稱。
`rangeKeyValue`	string	範圍索引鍵值。
`rangeKeyType`	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`payloadField`	string	動作承載。此名稱可自訂。

名稱	類型	描述
`dynamoDBv2`	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`putItem`	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
`tableName`	string	將寫入訊息資料的資料表。
`lambda`	LambdaAction	呼叫 Lambda 函數。
`functionArn`	string	Lambda 函式的 ARN。
`sns`	SnsAction	發佈至 Amazon SNS 主題。
`targetArn`	string	SNS 主題的 ARN。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`messageFormat`	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
`sqs`	SqsAction	發佈至 Amazon SQS 佇列。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`queueUrl`	string	Amazon SQS 佇列的 URL。
`useBase64`	布林值	指定是否使用 Base64 編碼。
`kinesis`	KinesisAction	將資料寫入至 Amazon Kinesis 串流。

名稱	類型	描述
`roleArn`	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
`streamName`	string	Amazon Kinesis 串流的名稱。
`partitionKey`	string	分割區索引鍵。
`republish`	RepublishAction	發佈至另一個 MQTT 主題。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`topic`	string	MQTT 主題名稱。
`qos`	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
`s3`	S3Action	寫入至 Amazon S3 儲存貯體。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`bucketName`	string	Amazon S3 儲存貯體。
`key`	string	物件金鑰。
`cannedAcl`	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
`firehose`	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
`roleArn`	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
`deliveryStreamName`	string	交付串流名稱。
`separator`	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
`cloudwatchMetric`	CloudwatchMetricAction	擷取 CloudWatch 指標。
`roleArn`	string	允許存取 CloudWatch 指標的 IAM 角色。
`metricNamespace`	string	CloudWatch 指標命名空間名稱。
`metricName`	string	CloudWatch 指標名稱。
`metricValue`	string	CloudWatch 指標值。

名稱	類型	描述
metricUnit	string	CloudWatch 支援的指標單位。
metricTimestamp	string	Unix 時間戳記 (選用)。
cloudwatchAlarm	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
roleArn	string	允許存取 CloudWatch 警示的 IAM 角色。
alarmName	string	CloudWatch 警示名稱。
stateReason	string	警示變更的原因。
stateValue	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
elasticsearch	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
roleArn	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。
endpoint	string 模式：https?://.*	您 Elasticsearch 網域的端點。
index	string	欲用來存放資料的 Elasticsearch 索引。
type	string	欲存放文件的類型。
id	string	欲存放文件的唯一識別符。
salesforce	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
token	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
url	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w *1,20*/w *1,20*/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
iotAnalytics	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
channelArn	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。
channelName	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。

名稱	類型	描述
`roleArn`	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
`iotEvents`	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
`inputName`	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
`messageId`	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
`roleArn`	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。 ("Action":"iotevents:BatchPutMessage")。
`stepFunctions`	StepFunctionsAction	開始執行 Step Functions 狀態機器。
`executionNamePrefix`	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
`stateMachineName`	string	Step Functions 狀態機器的名稱會開始執行。
`roleArn`	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。
`ruleDisabled`	布林值	指定規則是否停用。
`awsIotSqlVersion`	string	評估規則時所用的 SQL 規則引擎版本。
`errorAction`	動作	發生錯誤時欲執行的動作。
`dynamoDB`	DynamoDBAction	寫入至 DynamoDB 資料表。
`tableName`	string	DynamoDB 資料表的名稱。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`operation`	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。

名稱	類型	描述
`hashKeyField`	string	雜湊索引鍵名稱。
`hashKeyValue`	string	雜湊索引鍵值。
`hashKeyType`	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`rangeKeyField`	string	範圍索引鍵名稱。
`rangeKeyValue`	string	範圍索引鍵值。
`rangeKeyType`	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
`payloadField`	string	動作承載。此名稱可自訂。
`dynamoDBv2`	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`putItem`	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
`tableName`	string	將寫入訊息資料的資料表。
`lambda`	LambdaAction	呼叫 Lambda 函數。
`functionArn`	string	Lambda 函式的 ARN。
`sns`	SnsAction	發佈至 Amazon SNS 主題。
`targetArn`	string	SNS 主題的 ARN。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。

名稱	類型	描述
messageFormat	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
sqs	SqsAction	發佈至 Amazon SQS 佇列。
roleArn	string	授予存取權限的 IAM 角色 ARN。
queueUrl	string	Amazon SQS 佇列的 URL。
useBase64	布林值	指定是否使用 Base64 編碼。
kinesis	KinesisAction	將資料寫入至 Amazon Kinesis 串流。
roleArn	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
streamName	string	Amazon Kinesis 串流的名稱。
partitionKey	string	分割區索引鍵。
republish	RepublishAction	發佈至另一個 MQTT 主題。
roleArn	string	授予存取權限的 IAM 角色 ARN。
topic	string	MQTT 主題名稱。
qos	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
s3	S3Action	寫入至 Amazon S3 儲存貯體。
roleArn	string	授予存取權限的 IAM 角色 ARN。
bucketName	string	Amazon S3 儲存貯體。
key	string	物件金鑰。

名稱	類型	描述
cannedAcl	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
firehose	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
roleArn	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
deliveryStreamName	string	交付串流名稱。
separator	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
cloudwatchMetric	CloudwatchMetricAction	擷取 CloudWatch 指標。
roleArn	string	允許存取 CloudWatch 指標的 IAM 角色。
metricNamespace	string	CloudWatch 指標命名空間名稱。
metricName	string	CloudWatch 指標名稱。
metricValue	string	CloudWatch 指標值。
metricUnit	string	CloudWatch 支援的指標單位。
metricTimestamp	string	Unix 時間戳記 (選用)。
cloudwatchAlarm	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
roleArn	string	允許存取 CloudWatch 警示的 IAM 角色。
alarmName	string	CloudWatch 警示名稱。
stateReason	string	警示變更的原因。
stateValue	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
elasticsearch	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
roleArn	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。

名稱	類型	描述
endpoint	string 模式：https?://.*	您 Elasticsearch 網域的端點。
index	string	欲用來存放資料的 Elasticsearch 索引。
type	string	欲存放文件的類型。
id	string	欲存放文件的唯一識別符。
salesforce	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
token	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
url	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w *1,20*/w *1,20*/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
iotAnalytics	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
channelArn	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。
channelName	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。
roleArn	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
iotEvents	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
inputName	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
messageId	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
roleArn	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。("Action":"iotevents:BatchPutMessage")。

名稱	類型	描述
stepFunctions	StepFunctionsAction	開始執行 Step Functions 狀態機器。
executionNamePrefix	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
stateMachineName	string	Step Functions 狀態機器的名稱會開始執行。
roleArn	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

`UnauthorizedException`

未授權您執行此操作。

GetV2LoggingOptions

取得精細定義的記錄選項。

概要

```
aws iot  get-v2-logging-options  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
}
```

輸出

```
{
  "roleArn": "string",
  "defaultLogLevel": "string",
  "disableAllLogs": "boolean"
}
```

CLI 輸出欄位

名稱	類型	描述
`roleArn`	string	AWS IoT 寫入您的 CloudWatch Logs 所使用之 IAM 角色 ARN。
`defaultLogLevel`	string	預設日誌層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
`disableAllLogs`	布林值	停用所有日誌。

錯誤

`InternalException`

出現未預期的錯誤。

`NotConfiguredException`

資源未設定。

`ServiceUnavailableException`

此服務暫時無法使用。

ListActiveViolations

列出指定之 Device Defender 安全性描述檔的作用中違規。

概要

```
aws iot  list-active-violations \
    [--thing-name <value>] \
    [--security-profile-name <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string	已列出作用中違規的實物的名稱。

名稱	類型	描述
	長度上限：128；下限：1	
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	已列出違規之 Device Defender 安全性描述檔的名稱。
`nextToken`	字串	下一組結果的字符。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "activeViolations": [
    {
      "violationId": "string",
      "thingName": "string",
      "securityProfileName": "string",
      "behavior": {
        "name": "string",
        "metric": "string",
        "criteria": {
          "comparisonOperator": "string",
          "value": {
            "count": "long",
            "cidrs": [
              "string"
            ],
            "ports": [
              "integer"
            ]
          },
          "durationSeconds": "integer",
          "consecutiveDatapointsToAlarm": "integer",
          "consecutiveDatapointsToClear": "integer",
          "statisticalThreshold": {
            "statistic": "string"
          }
        }
      },
      "lastViolationValue": {
        "count": "long",
        "cidrs": [
          "string"
        ],
        "ports": [
          "integer"
        ]
      },
      "lastViolationTime": "timestamp",
      "violationStartTime": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
activeViolations	列出 成員：ActiveViolation	作用中違規的清單。
violationId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	作用中違規的 ID。
thingName	string 長度上限：128；下限：1	負責作用中違規的實物的名稱。
securityProfileName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	行為違規的安全性描述檔。
behavior	Behavior (行為)	正在違規的行為。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
metric	string	行為測量的內容。
criteria	BehaviorCriteria	可判斷裝置對於 metric 的操作是否正確的條件。
comparisonOperator	string	將測量的實物 (metric) 關聯到條件 (包含 value 或 statisticalThreshold) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
value	MetricValue	相較於 metric 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。

名稱	類型	描述
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，NUM_MESSAGES_SENT)。對於 statisticalThreshhold 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (durationSeconds) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (comparisonOperator)，則該裝置會被視為符合行為，否則就是違反行為。
lastViolationValue	MetricValue	造成最新違規的指標值 (測量)。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
lastViolationTime	timestamp	最新違規的發生時間。
violationStartTime	timestamp	違規開始的時間。

名稱	類型	描述
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

InvalidRequestException

請求的內容無效。

ResourceNotFoundException

指定的資源不存在。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

ListAttachedPolicies

列出連接至指定物件群組的政策。

概要

```
aws iot  list-attached-policies \
    --target <value> \
    [--recursive | --no-recursive] \
    [--marker <value>] \
    [--page-size <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "target": "string",
  "recursive": "boolean",
  "marker": "string",
  "pageSize": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
target	string	將列出政策的群組或委託人。
recursive	布林值	若此值為 true，將遞迴列出所連接的政策。
marker	string 模式：[A-Za-z0-9+/]+={0,2}	用來擷取下一組結果的符記。

名稱	類型	描述
pageSize	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。

輸出

```
{
  "policies": [
    {
      "policyName": "string",
      "policyArn": "string"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
policies	列出 成員：Policy Java 類別：java.util.List	政策。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於擷取下一組結果的符記，如果沒有其他結果則為 null。

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

ListAuditFindings

列出 Device Defender 稽核或在指定期間內執行之稽核時的發現 (結果)。(發現項目會保留 180 天)。

概要

```
aws iot  list-audit-findings \
    [--task-id <value>] \
    [--check-name <value>] \
    [--resource-identifier <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--start-time <value>] \
    [--end-time <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string",
  "checkName": "string",
  "resourceIdentifier": {
    "deviceCertificateId": "string",
    "caCertificateId": "string",
    "cognitoIdentityPoolId": "string",
    "clientId": "string",
    "policyVersionIdentifier": {
      "policyName": "string",
      "policyVersionId": "string"
    },
    "account": "string"
  },
  "maxResults": "integer",
  "nextToken": "string",
  "startTime": "timestamp",
  "endTime": "timestamp"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	可限制具指定 ID 的稽核結果的篩選條件。您必須指定 taskId 或 startTime 和 endTime，但不能同時指定兩者。
`checkName`	string	可限制指定的稽核檢查的發現結果的篩選條件。
`resourceIdentifier`	ResourceIdentifier	識別不合規資源的資訊。

名稱	類型	描述
deviceCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	string	Amazon Cognito 身分集區的 ID。
clientId	string	用戶端 ID。
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	string 模式：[0-9]+	與資源相關聯的政策版本 ID。
account	string 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。
nextToken	字串	下一組結果的字符。
startTime	timestamp	一個篩選功能，其能限制指定的時間過後所找到的結果。您必須指定 startTime 和 endTime 或 taskId，但不能同時指定兩者。
endTime	timestamp	一個篩選功能，其能限制指定的時間之前所找到的結果。您必須指定 startTime 和 endTime 或 taskId，但不能同時指定兩者。

輸出

```
{
  "findings": [
    {
      "findingId": "string",
      "taskId": "string",
      "checkName": "string",
      "taskStartTime": "timestamp",
```

```
      "findingTime": "timestamp",
      "severity": "string",
      "nonCompliantResource": {
        "resourceType": "string",
        "resourceIdentifier": {
          "deviceCertificateId": "string",
          "caCertificateId": "string",
          "cognitoIdentityPoolId": "string",
          "clientId": "string",
          "policyVersionIdentifier": {
            "policyName": "string",
            "policyVersionId": "string"
          },
          "account": "string"
        },
        "additionalInfo": {
          "string": "string"
        }
      },
      "relatedResources": [
        {
          "resourceType": "string",
          "resourceIdentifier": {
            "deviceCertificateId": "string",
            "caCertificateId": "string",
            "cognitoIdentityPoolId": "string",
            "clientId": "string",
            "policyVersionIdentifier": {
              "policyName": "string",
              "policyVersionId": "string"
            },
            "account": "string"
          },
          "additionalInfo": {
            "string": "string"
          }
        }
      ],
      "reasonForNonCompliance": "string",
      "reasonForNonComplianceCode": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`findings`	列出 成員：AuditFinding	稽核的發現 (結果)。
`findingId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	這組稽核發現結果的唯一識別符。此識別符用於將緩解任務套用至一或多組發現結果。
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	產生此結果 (發現) 的稽核 ID。

名稱	類型	描述
checkName	string	產生此結果的稽核檢查。
taskStartTime	timestamp	稽核開始的時間。
findingTime	timestamp	發現結果 (尋找) 的時間。
severity	string	結果 (發現) 的嚴重性。 列舉：CRITICAL | HIGH | MEDIUM | LOW
nonCompliantResource	NonCompliantResource	發現不符合稽核檢查的資源。
resourceType	string	未合規的資源類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	可識別不合規資源的資訊。
deviceCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	string	Amazon Cognito 身分集區的 ID。
clientId	string	用戶端 ID。
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	string 模式：[0-9]+	與資源相關聯的政策版本 ID。
account	string 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
additionalInfo	對應	有關不合規資源的其他資訊。

名稱	類型	描述
relatedResources	列出 成員：RelatedResource	相關資源的清單。
resourceType	string	資源的類型。 列舉：DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	可識別資源的資訊。
deviceCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	連接到資源的憑證 ID。
caCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	用於授權憑證之 CA 憑證的 ID。
cognitoIdentityPoolId	string	Amazon Cognito 身分集區的 ID。
clientId	string	用戶端 ID。
policyVersionIdentifier	PolicyVersionIdentifier	與資源相關聯的政策版本。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策的名稱。
policyVersionId	string 模式：[0-9]+	與資源相關聯的政策版本 ID。
account	string 長度上限：12；下限：12 模式：[0-9]+	與資源相關聯的帳戶。
additionalInfo	對應	關於資源的其他資訊。
reasonForNonCompliance	string	資源未合規的原因。
reasonForNonComplianceCode	string	可指出資源不合規原因的代碼。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditMitigationActionsExecutions

取得已執行之稽核緩解動作任務的狀態。

概要

```
aws iot  list-audit-mitigation-actions-executions \
    --task-id <value> \
    [--action-status <value>] \
    --finding-id <value> \
    [--max-results <value>] \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string",
  "actionStatus": "string",
  "findingId": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	指定此篩選條件，將結果限制在特定稽核緩解動作任務的動作。
`actionStatus`	string	指定此篩選條件，將結果限制在具有特定狀態的結果。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED | SKIPPED | PENDING
`findingId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	指定此篩選條件來限制結果，套用到特定的稽核結果。

名稱	類型	描述
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。
nextToken	字串	下一組結果的字符。

輸出

```
{
  "actionsExecutions": [
    {
      "taskId": "string",
      "findingId": "string",
      "actionName": "string",
      "actionId": "string",
      "status": "string",
      "startTime": "timestamp",
      "endTime": "timestamp",
      "errorCode": "string",
      "message": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
actionsExecutions	列出 成員：AuditMitigationActionExecutionMetadata	基於輸入參數的一組任務執行結果。詳細資訊包括套用的緩解動作、開始時間和任務狀態。
taskId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	套用到任務緩解動作的唯一識別符。
findingId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	套用任務和相關聯的緩解動作的發現結果的唯一識別碼。
actionName	string 長度上限：128 模式：[a-zA-Z0-9_-]+	正在由任務套用的緩解動作的易記名稱。
actionId	字串	正在由任務套用的緩解動作的唯一識別符。
status	字串	正在執行任務目前的狀態。 列舉：IN_PROGRESS | COMPLETED | FAILED |

名稱	類型	描述
		CANCELED | SKIPPED | PENDING
startTime	timestamp	任務開始的日期和時間。
endTime	timestamp	任務完成或取消的日期和時間。如果任務仍在執行，則為空白。
errorCode	string	如果發生錯誤，表示發生錯誤類型的代碼。
message	字串 長度上限：2048	如果發生錯誤，訊息會描述該錯誤。
nextToken	字串	下一組結果的字符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditMitigationActionsTasks

取得符合指定篩選條件的稽核緩解動作任務清單。

概要

```
aws iot  list-audit-mitigation-actions-tasks \
    [--audit-task-id <value>] \
    [--finding-id <value>] \
    [--task-status <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    --start-time <value> \
    --end-time <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "auditTaskId": "string",
  "findingId": "string",
  "taskStatus": "string",
  "maxResults": "integer",
  "nextToken": "string",
  "startTime": "timestamp",
  "endTime": "timestamp"
```

```
}
```

cli-input-json 欄位

名稱	類型	描述
auditTaskId	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	指定此篩選條件，將結果限制在套用到特定稽核結果的任務。
findingId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	指定此篩選條件，將結果限制在套用到特定稽核發現結果的任務。
taskStatus	string	指定此篩選條件，將結果限制在特定狀態的任務。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。
nextToken	字串	下一組結果的字符。
startTime	timestamp	指定此篩選條件，將結果限制在特定日期和時間當天或之後開始的任務。
endTime	timestamp	指定此篩選條件，將結果限制在特定日期和時間當天或之前完成或取消的任務。

輸出

```
{
  "tasks": [
    {
      "taskId": "string",
      "startTime": "timestamp",
      "taskStatus": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
tasks	列出 成員：AuditMitigationActionsTaskMetadata	符合篩選條件的稽核緩解任務集合。

名稱	類型	描述
	Java 類別：java.util.List	
`taskId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	任務的唯一識別符。
`startTime`	timestamp	稽核緩解動作任務的開始時間。
`taskStatus`	字串	稽核緩解行動任務的目前狀態。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
`nextToken`	字串	下一組結果的字符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuditTasks

列出特定期間執行的 Device Defender 稽核。

概要

```
aws iot  list-audit-tasks \
    --start-time <value> \
    --end-time <value> \
    [--task-type <value>] \
    [--task-status <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "startTime": "timestamp",
  "endTime": "timestamp",
  "taskType": "string",
  "taskStatus": "string",
  "nextToken": "string",
  "maxResults": "integer"
```

```
}
```

cli-input-json 欄位

名稱	類型	描述
startTime	timestamp	時期的開始。稽核資訊會保留一段有限時期 (180 天)。要求開始時間，然後才會在「InvalidRequestException」保留結果。
endTime	timestamp	時期的結束。
taskType	string	限制輸出到指定的稽核類型的篩選條件：可以是「ON_DEMAND_AUDIT_TASK」或「SCHEDULED__AUDIT_TASK」的其中一個。 列舉：ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStatus	string	篩選條件，將輸出限制在具有以下指定的完成狀態：可以是「IN_PROGRESS」、「COMPLETED」、「FAILED」或「CANCELED」的其中一個。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。

輸出

```
{
  "tasks": [
    {
      "taskId": "string",
      "taskStatus": "string",
      "taskType": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
tasks	列出	在指定期間內執行的稽核。

名稱	類型	描述
	成員：AuditTaskMetadata Java 類別：java.util.List	
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	此稽核的 ID。
`taskStatus`	string	此稽核的狀態。「IN_PROGRESS」、「COMPLETED」、「FAILED」或「CANCELED」其中之一。 列舉：IN_PROGRESS | COMPLETED | FAILED | CANCELED
`taskType`	string	此稽核的類型。「ON_DEMAND_AUDIT_TASK」或「SCHEDULED_AUDIT_TASK」的其中之一。 列舉：ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
`nextToken`	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListAuthorizers

列出您帳戶註冊的授權方。

概要

```
aws iot  list-authorizers \
    [--page-size <value>] \
    [--marker <value>] \
    [--ascending-order | --no-ascending-order] \
```

```
    [--status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "pageSize": "integer",
  "marker": "string",
  "ascendingOrder": "boolean",
  "status": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`pageSize`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。
`ascendingOrder`	布林值	回傳依字母順序排列的授權方清單。
`status`	string	所列出授權方的要求狀態。 列舉：ACTIVE | INACTIVE

輸出

```
{
  "authorizers": [
    {
      "authorizerName": "string",
      "authorizerArn": "string"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`authorizers`	列出 成員：AuthorizerSummary Java 類別：java.util.List	授權方。
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。

名稱	類型	描述
authorizerArn	string	授權方 ARN。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListBillingGroups

列出您建立的帳單群組。

概要

```
aws iot  list-billing-groups \
    [--next-token <value>] \
    [--max-results <value>] \
    [--name-prefix-filter <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "namePrefixFilter": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
nextToken	字串	用來擷取下一組結果的符記。

名稱	類型	描述
maxResults	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
namePrefixFilter	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	將結果限制為名稱有指定字首的帳單群組。

輸出

```
{
  "billingGroups": [
    {
      "groupName": "string",
      "groupArn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
billingGroups	列出 成員：GroupNameAndArn Java 類別：java.util.List	帳單群組的清單。
groupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組名稱。
groupArn	string	群組 ARN。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

ListCACertificates

列出於您 AWS 帳戶註冊的 CA 憑證。

結果分頁的預設頁面大小為 25。您可以使用回傳的標記來擷取其他結果。

概要

```
aws iot  list-ca-certificates \
    [--page-size <value>] \
    [--marker <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "pageSize": "integer",
  "marker": "string",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`pageSize`	integer 範圍 - 上限：250；下限：1	結果頁面大小。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
`ascendingOrder`	布林值	決定結果排序。

輸出

```
{
  "certificates": [
    {
      "certificateArn": "string",
      "certificateId": "string",
      "status": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`certificates`	列出	您 AWS 帳戶註冊的 CA 憑證。

名稱	類型	描述
	成員：CACertificate Java 類別：java.util.List	
`certificateArn`	string	CA 憑證的 ARN。
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證的 ID。
`status`	string	CA 憑證的狀態。 狀態值 REGISTER_INACTIVE 已棄用，請不要使用。 列舉：ACTIVE | INACTIVE
`creationDate`	timestamp	建立 CA 憑證的日期。
`nextMarker`	string 模式：[A-Za-z0-9+/]+={0,2}	於 CA 憑證清單內的目前位置。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListCertificates

列出於您 AWS 帳戶註冊的憑證。

結果分頁的預設頁面大小為 25。您可以使用回傳的標記來擷取其他結果。

概要

```
aws iot  list-certificates \
    [--page-size <value>] \
    [--marker <value>] \
```

```
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "pageSize": "integer",
  "marker": "string",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`pageSize`	integer 範圍 - 上限：250；下限：1	結果頁面大小。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
`ascendingOrder`	布林值	指定結果排序。若此值為 True，回傳的結果將依建立日期排序。

輸出

```
{
  "certificates": [
    {
      "certificateArn": "string",
      "certificateId": "string",
      "status": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`certificates`	列出 成員：Certificate Java 類別：java.util.List	憑證的描述。
`certificateArn`	string	憑證的 ARN。
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)

名稱	類型	描述
status	string	憑證的狀態。 狀態值 REGISTER_INACTIVE 已棄用，請不要使用。 列舉：ACTIVE | INACTIVE | REVOKED | PENDING_TRANSFER | REGISTER_INACTIVE | PENDING_ACTIVATION
creationDate	timestamp	建立憑證的日期和時間。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於下一組結果的標記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListCertificatesByCA

列出由指定 CA 憑證簽署的裝置憑證。

概要

```
aws iot  list-certificates-by-ca \
    --ca-certificate-id <value> \
    [--page-size <value>] \
    [--marker <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "caCertificateId": "string",
  "pageSize": "integer",
  "marker": "string",
```

```
  "ascendingOrder": "boolean"
}
```

cli-input-json 欄位

名稱	類型	描述
caCertificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證的 ID。此操作將列出所有此 CA 憑證簽署的已註冊裝置憑證。
pageSize	integer 範圍 - 上限：250；下限：1	結果頁面大小。
marker	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
ascendingOrder	布林值	指定結果排序。若此值為 True，回傳的結果將依建立日期排序。

輸出

```
{
  "certificates": [
    {
      "certificateArn": "string",
      "certificateId": "string",
      "status": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
certificates	列出 成員：Certificate Java 類別：java.util.List	由指定 CA 憑證簽署的裝置憑證。
certificateArn	string	憑證的 ARN。
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)
status	string	憑證的狀態。 狀態值 REGISTER_INACTIVE 已棄用，請不要使用。

名稱	類型	描述
		列舉：ACTIVE | INACTIVE | REVOKED | PENDING_TRANSFER | REGISTER_INACTIVE | PENDING_ACTIVATION
`creationDate`	timestamp	建立憑證的日期和時間。
`nextMarker`	string 模式：[A-Za-z0-9+/]+={0,2}	用於下一組結果的標記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListIndices

列出搜尋索引。

概要

```
aws iot  list-indices \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

名稱	類型	描述
maxResults	integer 範圍 - 上限：500；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "indexNames": [
    "string"
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
indexNames	列出 成員：IndexName Java 類別：java.util.List	索引名稱。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListJobExecutionsForJob

列出工作的工作執行。

概要

```
aws iot  list-job-executions-for-job \
    --job-id <value> \
    [--status <value>] \
    [--max-results <value>] \
```

```
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
  "status": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`status`	string	任務的狀態。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`maxResults`	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
`nextToken`	字串	用來擷取下一組結果的符記。

輸出

```
{
  "executionSummaries": [
    {
      "thingArn": "string",
      "jobExecutionSummary": {
        "status": "string",
        "queuedAt": "timestamp",
        "startedAt": "timestamp",
        "lastUpdatedAt": "timestamp",
        "executionNumber": "long"
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`executionSummaries`	列出	工作執行摘要清單。

名稱	類型	描述
	成員：JobExecutionSummaryForJob Java 類別：java.util.List	
`thingArn`	string	運作工作執行的物件 ARN。
`jobExecutionSummary`	JobExecutionSummary	包含關於工作執行的部分資訊。
`status`	string	工作執行的狀態。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`queuedAt`	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
`startedAt`	timestamp	任務執行開始的 epoch 時間，以秒為單位。
`lastUpdatedAt`	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
`executionNumber`	長整數	用於辨識此特定裝置執行之特定工作的字串 (由數字「0」至「9」組成)。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
`nextToken`	string	用於下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

ListJobExecutionsForThing

列出指定物件的工作執行。

概要

```
aws iot  list-job-executions-for-thing \
    --thing-name <value> \
    [--status <value>] \
    [--max-results <value>] \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "status": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`status`	string	讓您搜尋特定狀態之工作的選用篩選條件。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`maxResults`	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
`nextToken`	字串	用來擷取下一組結果的符記。

輸出

```
{
  "executionSummaries": [
    {
      "jobId": "string",
      "jobExecutionSummary": {
        "status": "string",
        "queuedAt": "timestamp",
        "startedAt": "timestamp",
        "lastUpdatedAt": "timestamp",
        "executionNumber": "long"
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`executionSummaries`	列出 成員：JobExecutionSummaryForThing Java 類別：java.util.List	工作執行摘要清單。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`jobExecutionSummary`	JobExecutionSummary	包含關於工作執行的部分資訊。
`status`	string	工作執行的狀態。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`queuedAt`	timestamp	任務排入佇列的 epoch 時間，以秒為單位。
`startedAt`	timestamp	任務執行開始的 epoch 時間，以秒為單位。
`lastUpdatedAt`	timestamp	任務執行最近一次更新的 epoch 時間，以秒為單位。
`executionNumber`	長整數	用於辨識此特定裝置執行之特定工作的字串 (由數字「0」至「9」組成)。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
`nextToken`	string	用於下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

ListJobs

列出工作。

概要

```
aws iot  list-jobs \
    [--status <value>] \
    [--target-selection <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--thing-group-name <value>] \
    [--thing-group-id <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "status": "string",
  "targetSelection": "string",
  "maxResults": "integer",
  "nextToken": "string",
  "thingGroupName": "string",
  "thingGroupId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`status`	string	讓您搜尋特定狀態之工作的選用篩選條件。 enum：IN_PROGRESS | CANCELED | COMPLETED | DELETION_IN_PROGRESS
`targetSelection`	string	指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是 CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。 列舉：CONTINUOUS | SNAPSHOT
`maxResults`	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。
`nextToken`	字串	用來擷取下一組結果的符記。

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	限制僅回傳指定群組之工作的篩選條件。
thingGroupId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	限制僅回傳指定群組之工作的篩選條件。

輸出

```
{
  "jobs": [
    {
      "jobArn": "string",
      "jobId": "string",
      "thingGroupId": "string",
      "targetSelection": "string",
      "status": "string",
      "createdAt": "timestamp",
      "lastUpdatedAt": "timestamp",
      "completedAt": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
jobs	列出 成員：JobSummary Java 類別：java.util.List	工作清單。
jobArn	string	工作 ARN。
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
thingGroupId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	物件群組的 ID。
targetSelection	string	指定工作是否要持續運作 (CONTINUOUS)，或在指定為目標的所有物件完成工作後視為完成 (SNAPSHOT)。如果是

名稱	類型	描述
		CONTINUOUS，則在目標中偵測到變更時，工作也可能會運作於物件上。例如，當物件新增至目標群組，工作就會在物件上運作，即使該工作已被原本就在群組中的所有物件完成。 列舉：CONTINUOUS | SNAPSHOT
`status`	string	工作摘要狀態。 enum：IN_PROGRESS | CANCELED | COMPLETED | DELETION_IN_PROGRESS
`createdAt`	timestamp	任務建立的 epoch 時間，以秒為單位。
`lastUpdatedAt`	timestamp	任務最近一次更新的 epoch 時間，以秒為單位。
`completedAt`	timestamp	任務完成的 epoch 時間，以秒為單位。
`nextToken`	string	用於下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

ListMitigationActions

取得符合指定篩選條件的所有緩解動作清單。

概要

```
aws iot  list-mitigation-actions \
    [--action-type <value>] \
    [--max-results <value>] \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "actionType": "string",
  "maxResults": "integer",
  "nextToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`actionType`	string	指定一個值來使用特定的動作類型限制緩解動作的結果。 列舉：UPDATE_DEVICE_CERTIFICATE | UPDATE_CA_CERTIFICATE | ADD_THINGS_TO_THING_GROUP | REPLACE_DEFAULT_POLICY_VERSION | ENABLE_IOT_LOGGING | PUBLISH_FINDING_TO_SNS
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。
`nextToken`	字串	下一組結果的字符。

輸出

```
{
  "actionIdentifiers": [
    {
      "actionName": "string",
      "actionArn": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`actionIdentifiers`	列出 成員：MitigationActionIdentifier	符合指定篩選條件的一組動作。
`actionName`	string 長度上限：128 模式：[a-zA-Z0-9_-]+	緩解動作的易記名稱。
`actionArn`	string	用於套用此緩解動作的 IAM 角色 ARN。

名稱	類型	描述
creationDate	timestamp	建立此緩解動作的日期。
nextToken	字串	下一組結果的字符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListOTAUpdates

列出 OTA 更新。

概要

```
aws iot  list-ota-updates \
    [--max-results <value>] \
    [--next-token <value>] \
    [--ota-update-status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "maxResults": "integer",
  "nextToken": "string",
  "otaUpdateStatus": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
nextToken	string	用於擷取下一組結果的權杖。
otaUpdateStatus	string	OTA 更新工作狀態。 列舉：CREATE_PENDING | CREATE_IN_PROGRESS | CREATE_COMPLETE | CREATE_FAILED

輸出

```
{
  "otaUpdates": [
    {
      "otaUpdateId": "string",
      "otaUpdateArn": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`otaUpdates`	列出 成員：OTAUpdateSummary	OTA 更新工作清單。
`otaUpdateId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	OTA 更新 ID。
`otaUpdateArn`	string	OTA 更新 ARN。
`creationDate`	timestamp	建立 OTA 更新的日期。
`nextToken`	string	用於取得下一組結果的符記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ServiceUnavailableException`

此服務暫時無法使用。

ListOutgoingCertificates

列出正在傳送但尚未接受的憑證。

概要

```
aws iot  list-outgoing-certificates \
    [--page-size <value>] \
    [--marker <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "pageSize": "integer",
  "marker": "string",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`pageSize`	integer 範圍 - 上限：250；下限：1	結果頁面大小。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
`ascendingOrder`	布林值	指定結果排序。若此值為 True，回傳的結果將依建立日期排序。

輸出

```
{
  "outgoingCertificates": [
    {
      "certificateArn": "string",
      "certificateId": "string",
      "transferredTo": "string",
      "transferDate": "timestamp",
      "transferMessage": "string",
      "creationDate": "timestamp"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`outgoingCertificates`	列出 成員：OutgoingCertificate Java 類別：java.util.List	正在傳送但尚未接受的憑證。
`certificateArn`	string	憑證 ARN。
`certificateId`	string	憑證 ID。

名稱	類型	描述
	長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	
`transferredTo`	string 長度上限：12；下限：12 模式：[0-9]+	傳送目標的 AWS 帳戶。
`transferDate`	timestamp	傳送啟動的日期。
`transferMessage`	string 長度上限：128	傳輸的訊息。
`creationDate`	timestamp	憑證建立日期。
`nextMarker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListPolicies

列出您的政策。

概要

```
aws iot  list-policies \
    [--marker <value>] \
    [--page-size <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
```

```
  "marker": "string",
  "pageSize": "integer",
  "ascendingOrder": "boolean"
}
```

cli-input-json 欄位

名稱	類型	描述
marker	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
pageSize	integer 範圍 - 上限：250；下限：1	結果頁面大小。
ascendingOrder	布林值	指定結果排序。若此值為 true，回傳的結果將依建立時間排序。

輸出

```
{
  "policies": [
    {
      "policyName": "string",
      "policyArn": "string"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
policies	列出 成員：Policy Java 類別：java.util.List	政策的描述。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於下一組結果的標記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListPolicyPrincipals

列出與指定政策建立關聯的委託人。

備註：此 API 已作廢。請改用 ListTargetsForPolicy。

概要

```
aws iot  list-policy-principals \
    --policy-name <value> \
    [--marker <value>] \
    [--page-size <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "marker": "string",
  "pageSize": "integer",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
`pageSize`	integer 範圍 - 上限：250；下限：1	結果頁面大小。
`ascendingOrder`	布林值	指定結果排序。若此值為 true，回傳的結果將依建立時間排序。

輸出

```
{
  "principals": [
    "string"
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
principals	列出 成員：PrincipalArn Java 類別：java.util.List	委託人的描述。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於下一組結果的標記，如果沒有其他結果則為 null。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListPolicyVersions

列出指定政策的版本並辨識預設版本。

概要

```
aws iot  list-policy-versions \
    --policy-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。

輸出

```
{
  "policyVersions": [
    {
      "versionId": "string",
      "isDefaultVersion": "boolean",
      "createDate": "timestamp"
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
`policyVersions`	列出 成員：PolicyVersion Java 類別：java.util.List	政策版本。
`versionId`	string 模式：[0-9]+	政策版本 ID。
`isDefaultVersion`	布林值	指定該政策版本是否為預設版本。
`createDate`	timestamp	建立政策的日期和時間。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListPrincipalPolicies

列出連接至指定委託人的政策。若您使用 Cognito Identity，此 ID 必須為 AmazonCognito Identity 格式。

備註：此 API 已作廢。請改用 ListAttachedPolicies。

概要

```
aws iot  list-principal-policies \
    --principal <value> \
    [--marker <value>] \
    [--page-size <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "principal": "string",
  "marker": "string",
  "pageSize": "integer",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`principal`	string	委託人。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	下一組結果的標記。
`pageSize`	integer 範圍 - 上限：250；下限：1	結果頁面大小。
`ascendingOrder`	布林值	指定結果排序。若此值為 true，回傳的結果將依建立時間排序。

輸出

```
{
```

```
  "policies": [
    {
      "policyName": "string",
      "policyArn": "string"
    }
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
policies	列出 成員：Policy Java 類別：java.util.List	政策。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於下一組結果的標記，如果沒有其他結果則為 null。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListPrincipalThings

列出與指定委託人建立關聯的物件。委託人可以是 X.509 憑證、IAM 使用者、群組和角色、Amazon Cognito 身分或聯合身分。

概要

```
aws iot  list-principal-things \
    [--next-token <value>] \
    [--max-results <value>] \
    --principal <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "principal": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`nextToken`	字串	用來擷取下一組結果的符記。
`maxResults`	integer 範圍 - 上限：250；下限：1	本次操作傳回結果的最大數量。
`principal`	string	委託人。

輸出

```
{
  "things": [
    "string"
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`things`	列出 成員：ThingName	物件。
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListRoleAliases

列出您帳戶註冊的角色別名。

概要

```
aws iot  list-role-aliases \
    [--page-size <value>] \
    [--marker <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "pageSize": "integer",
  "marker": "string",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`pageSize`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。
`ascendingOrder`	布林值	回傳依字母順序排列的角色別名清單。

輸出

```
{
  "roleAliases": [
    "string"
```

```
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
roleAliases	列出 成員：RoleAlias Java 類別：java.util.List	角色別名。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListScheduledAudits

列出所有已排程的稽核。

概要

```
aws iot  list-scheduled-audits \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。預設為 25。

輸出

```
{
  "scheduledAudits": [
    {
      "scheduledAuditName": "string",
      "scheduledAuditArn": "string",
      "frequency": "string",
      "dayOfMonth": "string",
      "dayOfWeek": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
scheduledAudits	列出 成員：ScheduledAuditMetadata Java 類別：java.util.List	已排程的稽核的清單。
scheduledAuditName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱。
scheduledAuditArn	string	已排程的稽核的 ARN。
frequency	string	排程的稽核發生的頻率。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	string 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程之稽核的日期 (如果 frequency 是「每月」)。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	string	當週執行已排程之稽核的日期 (如果 frequency 是「每週」或「每兩週」)。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六

名稱	類型	描述
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListSecurityProfiles

列出您已建立的 Device Defender 安全性描述檔。您可以使用篩選功能，只列出僅與實物群組，或僅與您的帳戶關聯的這些安全性描述檔。

概要

```
aws iot  list-security-profiles \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "securityProfileIdentifiers": [
    {
      "name": "string",
      "arn": "string"
```

```
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
securityProfileIdentifiers	列出 成員：SecurityProfileIdentifier Java 類別：java.util.List	安全性描述檔識別符的清單 (名稱和 ARN)。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供給安全性描述檔的名稱。
arn	字串	安全性描述檔的 ARN。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListSecurityProfilesForTarget

列出連接目標 (實物群組) 的 Device Defender 安全性描述檔。

概要

```
aws iot  list-security-profiles-for-target \
    [--next-token <value>] \
    [--max-results <value>] \
    [--recursive | --no-recursive] \
    --security-profile-target-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
```

```
  "recursive": "boolean",
  "securityProfileTargetArn": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
recursive	布林值	若此值為 true，則也傳回子群組。
securityProfileTargetArn	string	您要取得其連接之安全性描述檔的目標 (實物群組) 的 ARN。

輸出

```
{
  "securityProfileTargetMappings": [
    {
      "securityProfileIdentifier": {
        "name": "string",
        "arn": "string"
      },
      "target": {
        "arn": "string"
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
securityProfileTargetMappings	列出 成員：SecurityProfileTargetMapping Java 類別：java.util.List	安全性描述檔及其關聯目標的清單。
securityProfileIdentifier	SecurityProfileIdentifier	識別安全性描述檔的資訊。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您提供給安全性描述檔的名稱。
arn	字串	安全性描述檔的 ARN。
target	SecurityProfileTarget	有關與安全設定檔相關聯之目標 (實物群組) 的資訊。

名稱	類型	描述
arn	字串	安全性描述檔的 ARN。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListStreams

列出您 AWS 帳戶中的所有串流。

概要

```
aws iot  list-streams \
    [--max-results <value>] \
    [--next-token <value>] \
    [--ascending-order | --no-ascending-order]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "maxResults": "integer",
  "nextToken": "string",
  "ascendingOrder": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
nextToken	string	用於取得下一組結果的符記。
ascendingOrder	布林值	設定為 true 就會以遞增順序回傳串流清單。

輸出

```
{
  "streams": [
    {
      "streamId": "string",
      "streamArn": "string",
      "streamVersion": "integer",
      "description": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
streams	列出 成員：StreamSummary	串流清單。
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
streamArn	string	串流 ARN。
streamVersion	integer 範圍 - 上限：65535；下限：0	此串流的版本。
description	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。
nextToken	string	用於取得下一組結果的符記。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListTagsForResource

列出您已指派給資源的標籤 (中繼資料)。

概要

```
aws iot  list-tags-for-resource \
    --resource-arn <value> \
    [--next-token <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "resourceArn": "string",
  "nextToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`resourceArn`	string	資源的 ARN。
`nextToken`	字串	用來擷取下一組結果的符記。

輸出

```
{
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`tags`	列出 成員：Tag Java 類別：java.util.List	指派給資源之標籤的清單。
`Key`	string	標籤的金鑰。
`Value`	string	標籤的值。
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

ListTargetsForPolicy

列出指定政策的目標。

概要

```
aws iot  list-targets-for-policy \
    --policy-name <value> \
    [--marker <value>] \
    [--page-size <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "marker": "string",
  "pageSize": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`marker`	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。
`pageSize`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
```

```
  "targets": [
    "string"
  ],
  "nextMarker": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
targets	列出 成員：PolicyTarget Java 類別：java.util.List	政策目標。
nextMarker	string 模式：[A-Za-z0-9+/]+={0,2}	用於取得下一組結果的標記。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

ListTargetsForSecurityProfile

列出與指定 Device Defender 安全性描述檔相關聯的目標 (實物群組)。

概要

```
aws iot  list-targets-for-security-profile \
    --security-profile-name <value> \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
```

```
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	安全性描述檔。
`nextToken`	字串	下一組結果的字符。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "securityProfileTargets": [
    {
      "arn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`securityProfileTargets`	清單 成員：SecurityProfileTarget Java 類別：java.util.List	安全性描述檔連接的實物群組。
`arn`	字串	安全性描述檔的 ARN。
`nextToken`	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 `null`。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

ListThingGroups

列出您帳戶內的物件群組。

概要

```
aws iot  list-thing-groups \
    [--next-token <value>] \
    [--max-results <value>] \
    [--parent-group <value>] \
    [--name-prefix-filter <value>] \
    [--recursive | --no-recursive]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "parentGroup": "string",
  "namePrefixFilter": "string",
  "recursive": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`nextToken`	字串	用來擷取下一組結果的符記。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
`parentGroup`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	將結果限制為具備指定父群組的篩選條件。
`namePrefixFilter`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	將結果限制為具備指定名稱字首的篩選條件。

名稱	類型	描述
recursive	布林值	若此值為 true，亦將回傳子群組。

輸出

```
{
  "thingGroups": [
    {
      "groupName": "string",
      "groupArn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
thingGroups	列出 成員：GroupNameAndArn Java 類別：java.util.List	物件群組。
groupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組名稱。
groupArn	string	群組 ARN。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListThingGroupsForThing

列出指定物件所屬的物件群組。

概要

```
aws iot  list-thing-groups-for-thing \
    --thing-name <value> \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`nextToken`	字串	用來擷取下一組結果的符記。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "thingGroups": [
    {
      "groupName": "string",
      "groupArn": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`thingGroups`	列出 成員：GroupNameAndArn Java 類別：java.util.List	物件群組。
`groupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組名稱。

名稱	類型	描述
groupArn	string	群組 ARN。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListThingPrincipals

列出與指定物件建立關聯的委託人。委託人可以是 X.509 憑證、IAM 使用者、群組和角色、Amazon Cognito 身分或聯合身分。

概要

```
aws iot  list-thing-principals \
    --thing-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。

輸出

```
{
  "principals": [
    "string"
  ]
```

```
}
```

CLI 輸出欄位

名稱	類型	描述
`principals`	列出 成員：PrincipalArn Java 類別：java.util.List	與指定物件建立關聯的委託人。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListThingRegistrationTaskReports

有關物件註冊任務的資訊。

概要

```
aws iot  list-thing-registration-task-reports \
    --task-id <value> \
    --report-type <value> \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string",
  "reportType": "string",
  "nextToken": "string",
```

```
  "maxResults": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
taskId	string 長度上限：40	任務的 ID。
reportType	string	任務報告的類型。 列舉：ERRORS | RESULTS
nextToken	字串	用來擷取下一組結果的符記。
maxResults	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。

輸出

```
{
  "resourceLinks": [
    "string"
  ],
  "reportType": "string",
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
resourceLinks	列出 成員：S3FileUrl	任務資源的連結。
reportType	string	任務報告的類型。 列舉：ERRORS | RESULTS
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

ListThingRegistrationTasks

列出大量物件佈建任務。

概要

```
aws iot  list-thing-registration-tasks \
    [--next-token <value>] \
    [--max-results <value>] \
    [--status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "status": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`nextToken`	字串	用來擷取下一組結果的符記。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。
`status`	string	大量物件佈建任務的狀態。 列舉：InProgress | Completed | Failed | Cancelled | Cancelling

輸出

```
{
  "taskIds": [
    "string"
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`taskIds`	列出	大量物件佈建任務的 ID 清單。

名稱	類型	描述
	成員：TaskId	
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

ListThingTypes

列出現有物件類型。

概要

```
aws iot  list-thing-types \
    [--next-token <value>] \
    [--max-results <value>] \
    [--thing-type-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
  "maxResults": "integer",
  "thingTypeName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
nextToken	字串	用來擷取下一組結果的符記。
maxResults	integer 範圍 - 上限：250；下限：1	本次操作傳回結果的最大數量。
thingTypeName	string	物件類型名稱。

名稱	類型	描述
	長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	

輸出

```
{
  "thingTypes": [
    {
      "thingTypeName": "string",
      "thingTypeArn": "string",
      "thingTypeProperties": {
        "thingTypeDescription": "string",
        "searchableAttributes": [
          "string"
        ]
      },
      "thingTypeMetadata": {
        "deprecated": "boolean",
        "deprecationDate": "timestamp",
        "creationDate": "timestamp"
      }
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`thingTypes`	列出 成員：ThingTypeDefinition Java 類別：java.util.List	物件類型。
`thingTypeName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
`thingTypeArn`	string	物件類型 ARN。
`thingTypeProperties`	ThingTypeProperties	物件類型的 ThingTypeProperties。
`thingTypeDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件類型描述。
`searchableAttributes`	列出 成員：AttributeName Java 類別：java.util.List	可搜尋物件屬性名稱清單。

名稱	類型	描述
thingTypeMetadata	ThingTypeMetadata	ThingTypeMetadata 內含物件類型的其他資訊，包括建立日期和時間、表示物件類型是否已棄用的值，以及棄用的日期和時間。
deprecated	布林值	指定是否棄用物件類型。若此值為 true，新物件將無法與此類型建立關聯。
deprecationDate	timestamp	棄用物件類型的日期和時間。
creationDate	timestamp	建立物件類型的日期和時間。
nextToken	string	用於下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListThings

列出您的物件。使用 attributeName 和 attributeValue 參數來篩選您的物件。例如，呼叫 `ListThings`並將其參數 attributeName 和 attributeValue 分別設定為 Color 和 Red，將擷取登錄檔內具備屬性 Color 且其值為 Red 的所有物件。

概要

```
aws iot  list-things \
    [--next-token <value>] \
    [--max-results <value>] \
    [--attribute-name <value>] \
    [--attribute-value <value>] \
    [--thing-type-name <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "nextToken": "string",
```

```
  "maxResults": "integer",
  "attributeName": "string",
  "attributeValue": "string",
  "thingTypeName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
nextToken	字串	用來擷取下一組結果的符記。
maxResults	integer 範圍 - 上限：250；下限：1	本次操作傳回結果的最大數量。
attributeName	string 長度上限：128 模式：[a-zA-Z0-9_.,@/:#-]+	用於搜尋物件的屬性名稱。
attributeValue	string 長度上限：800 模式：[a-zA-Z0-9_.,@/:#-]*	用於搜尋物件的屬性值。
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	用於搜尋物件的物件類型名稱。

輸出

```
{
  "things": [
    {
      "thingName": "string",
      "thingTypeName": "string",
      "thingArn": "string",
      "attributes": {
        "string": "string"
      },
      "version": "long"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
things	列出 成員：ThingAttribute Java 類別：java.util.List	物件。

名稱	類型	描述
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱 (若此物件已與某類型建立關聯)。
thingArn	string	物件 ARN。
attributes	對應	名稱/值對的物件屬性清單。
version	長整數	登錄檔內物件記錄的版本。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

ListThingsInBillingGroup

列出您已新增至特定帳單群組的物件。

概要

```
aws iot  list-things-in-billing-group \
    --billing-group-name <value> \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "billingGroupName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
billingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
nextToken	字串	用來擷取下一組結果的符記。
maxResults	integer 範圍 - 上限：250；下限：1	每筆請求傳回結果的最大數量。

輸出

```
{
  "things": [
    "string"
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
things	列出 成員：ThingName	帳單群組中的物件清單。
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

ListThingsInThingGroup

列出指定群組的物件。

概要

```
aws iot  list-things-in-thing-group \
    --thing-group-name <value> \
    [--recursive | --no-recursive] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "recursive": "boolean",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件群組名稱。
`recursive`	布林值	若此值為 true，將列出此物件群組和所有子群組內的物件。
`nextToken`	字串	用來擷取下一組結果的符記。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "things": [
    "string"
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`things`	列出 成員：ThingName	指定物件群組的物件。

名稱	類型	描述
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ListTopicRules

列出特定主題的規則。

概要

```
aws iot  list-topic-rules \
    [--topic <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--rule-disabled | --no-rule-disabled]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "topic": "string",
  "maxResults": "integer",
  "nextToken": "string",
  "ruleDisabled": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
topic	string	主題。
maxResults	integer 範圍 - 上限：10000；下限：1	回傳結果的數量上限。
nextToken	string	用於擷取下一個值的符記。
ruleDisabled	布林值	指定規則是否停用。

輸出

```
{
  "rules": [
    {
      "ruleArn": "string",
      "ruleName": "string",
      "topicPattern": "string",
      "createdAt": "timestamp",
      "ruleDisabled": "boolean"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
rules	列出 成員：TopicRuleListItem	規則。
ruleArn	string	規則 ARN。
ruleName	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。
topicPattern	string	符合的主題名稱模式。
createdAt	timestamp	建立規則的日期和時間。
ruleDisabled	布林值	指定規則是否停用。
nextToken	string	用於擷取下一個值的符記。

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

ListV2LoggingLevels

列出記錄層級。

概要

```
aws iot  list-v2-logging-levels \
    [--target-type <value>] \
```

```
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "targetType": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`targetType`	string	您設定欲記錄的資源類型，必須為 `THING_Group`。 列舉：DEFAULT | THING_GROUP
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。
`maxResults`	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "logTargetConfigurations": [
    {
      "logTarget": {
        "targetType": "string",
        "targetName": "string"
      },
      "logLevel": "string"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`logTargetConfigurations`	列出 成員：LogTargetConfiguration	目標的記錄組態。
`logTarget`	LogTarget	日誌目標。
`targetType`	string	目標類型。 列舉：DEFAULT | THING_GROUP

名稱	類型	描述
`targetName`	string	目標名稱。
`logLevel`	string	記錄層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。

錯誤

`InternalException`

出現未預期的錯誤。

`NotConfiguredException`

資源未設定。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

ListViolationEvents

列出特定期間內發現的 Device Defender 安全性描述檔違反。您可以針對特別的安全性描述檔、行為或實物 (裝置)，使用篩選功能來限制發出這些提醒的結果。

概要

```
aws iot  list-violation-events \
    --start-time <value> \
    --end-time <value> \
    [--thing-name <value>] \
    [--security-profile-name <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "startTime": "timestamp",
  "endTime": "timestamp",
  "thingName": "string",
  "securityProfileName": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

cli-input-json 欄位

名稱	類型	描述
startTime	timestamp	列出之提醒的開始時間。
endTime	timestamp	列出之提醒的結束時間。
thingName	string 長度上限：128；下限：1	一個篩選功能，其能限制指定之實物所引起的提醒的結果。
securityProfileName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	一個篩選功能，其能限制指定之安全性描述檔所產生的提醒的結果。
nextToken	字串	下一組結果的字符。
maxResults	integer 範圍 - 上限：250；下限：1	一次可回傳結果的數量上限。

輸出

```
{
  "violationEvents": [
    {
      "violationId": "string",
      "thingName": "string",
      "securityProfileName": "string",
      "behavior": {
        "name": "string",
        "metric": "string",
        "criteria": {
          "comparisonOperator": "string",
          "value": {
            "count": "long",
            "cidrs": [
              "string"
            ],
            "ports": [
              "integer"
            ]
          },
          "durationSeconds": "integer",
          "consecutiveDatapointsToAlarm": "integer",
          "consecutiveDatapointsToClear": "integer",
          "statisticalThreshold": {
            "statistic": "string"
          }
        }
      },
      "metricValue": {
        "count": "long",
        "cidrs": [
          "string"
        ],
        "ports": [
          "integer"
        ]
      },
```

```
      "violationEventType": "string",
      "violationEventTime": "timestamp"
    }
  ],
  "nextToken": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
violationEvents	列出 成員：ViolationEvent	在指定時間間隔內針對此帳戶發出的安全性描述檔違規提醒，可能由安全性描述檔、違規的行為或實物 (裝置) 違規所篩選出來。
violationId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	違規事件的 ID。
thingName	string 長度上限：128；下限：1	負責違規事件的實物的名稱。
securityProfileName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	行為違規之安全性描述檔的名稱。
behavior	Behavior (行為)	違規的行為。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
metric	string	行為測量的內容。
criteria	BehaviorCriteria	可判斷裝置對於 metric 的操作是否正確的條件。
comparisonOperator	string	將測量的實物 (metric) 關聯到條件 (包含 value 或 statisticalThreshold) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
value	MetricValue	相較於 metric 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。

名稱	類型	描述
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，NUM_MESSAGES_SENT)。對於 statisticalThreshhold 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (durationSeconds) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (comparisonOperator)，則該裝置會被視為符合行為，否則就是違反行為。
metricValue	MetricValue	指標 (測量) 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。

名稱	類型	描述
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
violationEventType	string	違規事件的類型。 列舉：in-alarm | alarm-cleared | alarm-invalidated
violationEventTime	timestamp	發生違規事件的時間。
nextToken	字串	可用於擷取下一組結果的字符，或如果沒有其他結果則為 null。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

發佈

發佈狀態資訊。

如需詳細資訊，請參閱《AWS IoT 開發人員指南》中的 HTTP 通訊協定。

概要

```
aws iot-data  publish \
    [--topic <value>] \
    [--qos <value>] \
    [--payload <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "topic": "string",
  "qos": "integer",
  "payload": "blob"
}
```

`cli-input-json` 欄位

名稱	類型	描述
topic	string	MQTT 主題名稱。

名稱	類型	描述
qos	integer 範圍 - 上限：1；下限：0	服務品質 (QoS) 等級。
payload	blob	JSON 格式的狀態資訊。

輸出

無

錯誤

`InternalFailureException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`UnauthorizedException`

未授權您執行此操作。

`MethodNotAllowedException`

不支援所指定的 HTTP 動詞和 URI 組合。

RegisterCACertificate

向 AWS IoT 註冊 CA 憑證。此 CA 憑證可用於簽署裝置憑證，而此裝置憑證就能夠向 AWS IoT 註冊。您的每個 AWS 帳戶，相同主體欄位可用於註冊至多 10 個 CA 憑證。如此一來，最多 10 個憑證授權機構能夠簽署您的裝置憑證。若您已註冊多個 CA 憑證，請務必在您註冊裝置憑證時，一同傳送該 CA 憑證和 RegisterCertificate API。

概要

```
aws iot  register-ca-certificate \
    --ca-certificate <value> \
    --verification-certificate <value> \
    [--set-as-active | --no-set-as-active] \
    [--allow-auto-registration | --no-allow-auto-registration] \
    [--registration-config <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "caCertificate": "string",
  "verificationCertificate": "string",
  "setAsActive": "boolean",
  "allowAutoRegistration": "boolean",
  "registrationConfig": {
    "templateBody": "string",
    "roleArn": "string"
  }
```

```
}
```

cli-input-json 欄位

名稱	類型	描述
caCertificate	string 長度上限：65536；下限：1	CA 憑證。
verificationCertificate	string 長度上限：65536；下限：1	私有金鑰驗證憑證。
setAsActive	布林值	可指定 CA 憑證是否設定為作用中的布林值。
allowAutoRegistration	布林值	讓此 CA 憑證可用於裝置憑證的自動註冊。
registrationConfig	RegistrationConfig	註冊組態的資訊。
templateBody	string	範本本身。
roleArn	字串 長度上限：2048；下限：20	此角色的 ARN。

輸出

```
{
  "certificateArn": "string",
  "certificateId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
certificateArn	string	CA 憑證 ARN。
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證識別符。

錯誤

ResourceAlreadyExistsException

此資源已存在。

RegistrationCodeValidationException

註冊碼無效。

InvalidRequestException

請求的內容無效。

`CertificateValidationException`

憑證無效。

`ThrottlingException`

請求頻率超過限制。

`LimitExceededException`

已超出限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

RegisterCertificate

向 AWS IoT 註冊裝置憑證。如果您的多個 CA 憑證具有相同的主體欄位，您必須指定用於簽署該裝置憑證以進行註冊的 CA 憑證。

概要

```
aws iot  register-certificate \
    --certificate-pem <value> \
    [--ca-certificate-pem <value>] \
    [--set-as-active | --no-set-as-active] \
    [--status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificatePem": "string",
  "caCertificatePem": "string",
  "status": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificatePem`	string 長度上限：65536；下限：1	憑證資料 (PEM 格式)。
`caCertificatePem`	string 長度上限：65536；下限：1	用於簽署裝置憑證以進行註冊的 CA 憑證。

名稱	類型	描述
status	string	所註冊憑證的要求狀態。 列舉：ACTIVE | INACTIVE | REVOKED | PENDING_TRANSFER | REGISTER_INACTIVE | PENDING_ACTIVATION

輸出

```
{
  "certificateArn": "string",
  "certificateId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
certificateArn	string	憑證 ARN。
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證識別符。

錯誤

`ResourceAlreadyExistsException`

此資源已存在。

`InvalidRequestException`

請求的內容無效。

`CertificateValidationException`

憑證無效。

`CertificateStateException`

不允許此憑證操作。

`CertificateConflictException`

無法驗證您欲用於簽署裝置憑證的 CA 憑證以進行註冊。當您註冊的多個 CA 憑證使用相同主體欄位和公開金鑰，即會發生此錯誤。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

RegisterThing

佈建物件。

概要

```
aws iot  register-thing \
    --template-body <value> \
    [--parameters <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "templateBody": "string",
  "parameters": {
    "string": "string"
  }
}
```

`cli-input-json` 欄位

名稱	類型	描述
`templateBody`	string	該佈建範本。如需詳細資訊，請參閱「程式設計佈建」。
`parameters`	對應	用於佈建物件的參數。如需詳細資訊，請參閱「程式設計佈建」。

輸出

```
{
  "certificatePem": "string",
  "resourceArns": {
    "string": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`certificatePem`	string 長度上限：65536；下限：1	。
`resourceArns`	對應	所產生資源的 ARN。

錯誤

`InternalFailureException`

出現未預期的錯誤。

`ServiceUnavailableException`

此服務暫時無法使用。

`InvalidRequestException`

請求的內容無效。

`UnauthorizedException`

未授權您執行此操作。

`ThrottlingException`

請求頻率超過限制。

`ConflictingResourceUpdateException`

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

`ResourceRegistrationFailureException`

資源註冊失敗。

RejectCertificateTransfer

拒絕待定憑證傳送。AWS IoT 拒絕憑證傳送之後，憑證狀態會自 PENDING_TRANSFER 變更為 INACTIVE。

若要檢查待定憑證傳送，請呼叫 ListCertificates 來列舉您的憑證。

此操作僅能由傳送目的地呼叫。呼叫之後，憑證將會回傳至來源帳戶，狀態則為 INACTIVE。

概要

```
aws iot  reject-certificate-transfer \
    --certificate-id <value> \
    [--reject-reason <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string",
  "rejectReason": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)

名稱	類型	描述
rejectReason	string 長度上限：128	憑證傳送遭拒的原因。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

TransferAlreadyCompletedException

無法還原憑證傳送，因為傳送已完成。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

RemoveThingFromBillingGroup

從帳單群組移除特定物件。

概要

```
aws iot  remove-thing-from-billing-group \
    [--billing-group-name <value>] \
    [--billing-group-arn <value>] \
    [--thing-name <value>] \
    [--thing-arn <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "billingGroupName": "string",
  "billingGroupArn": "string",
  "thingName": "string",
  "thingArn": "string"
```

```
}
```

cli-input-json 欄位

名稱	類型	描述
billingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
billingGroupArn	string	帳單群組的 ARN。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	要從帳單群組移除的物件名稱。
thingArn	string	要從帳單群組移除的物件 ARN。

輸出

無

錯誤

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

ResourceNotFoundException

指定的資源不存在。

RemoveThingFromThingGroup

自指定群組移除指定物件。

概要

```
aws iot  remove-thing-from-thing-group \
    [--thing-group-name <value>] \
    [--thing-group-arn <value>] \
    [--thing-name <value>] \
    [--thing-arn <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "thingGroupName": "string",
  "thingGroupArn": "string",
  "thingName": "string",
  "thingArn": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
thingGroupName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組名稱。
thingGroupArn	string	群組 ARN。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	自群組移除的物件名稱。
thingArn	string	自群組移除的物件 ARN。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

ReplaceTopicRule

取代規則。您必須指定新規則的所有參數。規則的建立屬於管理員層級的動作。任何擁有許可而能建立規則的使用者，將能夠存取規則所處理的資料。

概要

```
aws iot  replace-topic-rule \
    --rule-name <value> \
    --topic-rule-payload <value>  \
    [--cli-input-json <value>] \
```

```
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "ruleName": "string",
  "topicRulePayload": {
    "sql": "string",
    "description": "string",
    "actions": [
      {
        "dynamoDB": {
          "tableName": "string",
          "roleArn": "string",
          "operation": "string",
          "hashKeyField": "string",
          "hashKeyValue": "string",
          "hashKeyType": "string",
          "rangeKeyField": "string",
          "rangeKeyValue": "string",
          "rangeKeyType": "string",
          "payloadField": "string"
        },
        "dynamoDBv2": {
          "roleArn": "string",
          "putItem": {
            "tableName": "string"
          }
        },
        "lambda": {
          "functionArn": "string"
        },
        "sns": {
          "targetArn": "string",
          "roleArn": "string",
          "messageFormat": "string"
        },
        "sqs": {
          "roleArn": "string",
          "queueUrl": "string",
          "useBase64": "boolean"
        },
        "kinesis": {
          "roleArn": "string",
          "streamName": "string",
          "partitionKey": "string"
        },
        "republish": {
          "roleArn": "string",
          "topic": "string",
          "qos": "integer"
        },
        "s3": {
          "roleArn": "string",
          "bucketName": "string",
          "key": "string",
          "cannedAcl": "string"
        },
        "firehose": {
          "roleArn": "string",
          "deliveryStreamName": "string",
          "separator": "string"
        },
        "cloudwatchMetric": {
          "roleArn": "string",
```

```
          "metricNamespace": "string",
          "metricName": "string",
          "metricValue": "string",
          "metricUnit": "string",
          "metricTimestamp": "string"
        },
        "cloudwatchAlarm": {
          "roleArn": "string",
          "alarmName": "string",
          "stateReason": "string",
          "stateValue": "string"
        },
        "elasticsearch": {
          "roleArn": "string",
          "endpoint": "string",
          "index": "string",
          "type": "string",
          "id": "string"
        },
        "salesforce": {
          "token": "string",
          "url": "string"
        },
        "iotAnalytics": {
          "channelArn": "string",
          "channelName": "string",
          "roleArn": "string"
        },
        "iotEvents": {
          "inputName": "string",
          "messageId": "string",
          "roleArn": "string"
        },
        "stepFunctions": {
          "executionNamePrefix": "string",
          "stateMachineName": "string",
          "roleArn": "string"
        }
      }
    ],
    "ruleDisabled": "boolean",
    "awsIotSqlVersion": "string",
    "errorAction": {
      "dynamoDB": {
        "tableName": "string",
        "roleArn": "string",
        "operation": "string",
        "hashKeyField": "string",
        "hashKeyValue": "string",
        "hashKeyType": "string",
        "rangeKeyField": "string",
        "rangeKeyValue": "string",
        "rangeKeyType": "string",
        "payloadField": "string"
      },
      "dynamoDBv2": {
        "roleArn": "string",
        "putItem": {
          "tableName": "string"
        }
      },
      "lambda": {
        "functionArn": "string"
      },
      "sns": {
        "targetArn": "string",
```

```
        "roleArn": "string",
        "messageFormat": "string"
      },
      "sqs": {
        "roleArn": "string",
        "queueUrl": "string",
        "useBase64": "boolean"
      },
      "kinesis": {
        "roleArn": "string",
        "streamName": "string",
        "partitionKey": "string"
      },
      "republish": {
        "roleArn": "string",
        "topic": "string",
        "qos": "integer"
      },
      "s3": {
        "roleArn": "string",
        "bucketName": "string",
        "key": "string",
        "cannedAcl": "string"
      },
      "firehose": {
        "roleArn": "string",
        "deliveryStreamName": "string",
        "separator": "string"
      },
      "cloudwatchMetric": {
        "roleArn": "string",
        "metricNamespace": "string",
        "metricName": "string",
        "metricValue": "string",
        "metricUnit": "string",
        "metricTimestamp": "string"
      },
      "cloudwatchAlarm": {
        "roleArn": "string",
        "alarmName": "string",
        "stateReason": "string",
        "stateValue": "string"
      },
      "elasticsearch": {
        "roleArn": "string",
        "endpoint": "string",
        "index": "string",
        "type": "string",
        "id": "string"
      },
      "salesforce": {
        "token": "string",
        "url": "string"
      },
      "iotAnalytics": {
        "channelArn": "string",
        "channelName": "string",
        "roleArn": "string"
      },
      "iotEvents": {
        "inputName": "string",
        "messageId": "string",
        "roleArn": "string"
      },
      "stepFunctions": {
        "executionNamePrefix": "string",
```

```
        "stateMachineName": "string",
        "roleArn": "string"
      }
    }
  }
}
```

cli-input-json 欄位

名稱	類型	描述
ruleName	string 長度上限：128；下限：1 模式：^[a-zA-Z0-9_]+$	規則名稱。
topicRulePayload	TopicRulePayload	規則承載。
sql	string	用於查詢主題的 SQL 陳述式。如需詳細資訊，請參閱 AWS IoT 開發人員指南的 AWS IoT SQL 參考。
description	string	規則的描述。
actions	列出 成員：Action	與規則建立關聯的動作。
dynamoDB	DynamoDBAction	寫入至 DynamoDB 資料表。
tableName	string	DynamoDB 資料表的名稱。
roleArn	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
operation	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：`INSERT`、`UPDATE` 或 `DELETE`。
hashKeyField	string	雜湊索引鍵名稱。
hashKeyValue	string	雜湊索引鍵值。
hashKeyType	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
rangeKeyField	string	範圍索引鍵名稱。
rangeKeyValue	string	範圍索引鍵值。
rangeKeyType	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER

名稱	類型	描述
`payloadField`	string	動作承載。此名稱可自訂。
`dynamoDBv2`	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
`roleArn`	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
`putItem`	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
`tableName`	string	將寫入訊息資料的資料表。
`lambda`	LambdaAction	呼叫 Lambda 函數。
`functionArn`	string	Lambda 函式的 ARN。
`sns`	SnsAction	發佈至 Amazon SNS 主題。
`targetArn`	string	SNS 主題的 ARN。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`messageFormat`	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
`sqs`	SqsAction	發佈至 Amazon SQS 佇列。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`queueUrl`	string	Amazon SQS 佇列的 URL。
`useBase64`	布林值	指定是否使用 Base64 編碼。

名稱	類型	描述
`kinesis`	KinesisAction	將資料寫入至 Amazon Kinesis 串流。
`roleArn`	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
`streamName`	string	Amazon Kinesis 串流的名稱。
`partitionKey`	string	分割區索引鍵。
`republish`	RepublishAction	發佈至另一個 MQTT 主題。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`topic`	string	MQTT 主題名稱。
`qos`	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
`s3`	S3Action	寫入至 Amazon S3 儲存貯體。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`bucketName`	string	Amazon S3 儲存貯體。
`key`	string	物件金鑰。
`cannedAcl`	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
`firehose`	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
`roleArn`	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
`deliveryStreamName`	string	交付串流名稱。
`separator`	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
`cloudwatchMetric`	CloudwatchMetricAction	擷取 CloudWatch 指標。
`roleArn`	string	允許存取 CloudWatch 指標的 IAM 角色。
`metricNamespace`	string	CloudWatch 指標命名空間名稱。

名稱	類型	描述
metricName	string	CloudWatch 指標名稱。
metricValue	string	CloudWatch 指標值。
metricUnit	string	CloudWatch 支援的指標單位。
metricTimestamp	string	Unix 時間戳記 (選用)。
cloudwatchAlarm	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
roleArn	string	允許存取 CloudWatch 警示的 IAM 角色。
alarmName	string	CloudWatch 警示名稱。
stateReason	string	警示變更的原因。
stateValue	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
elasticsearch	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
roleArn	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。
endpoint	string 模式：https?://.*	您 Elasticsearch 網域的端點。
index	string	欲用來存放資料的 Elasticsearch 索引。
type	string	欲存放文件的類型。
id	string	欲存放文件的唯一識別符。
salesforce	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
token	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
url	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w 1,20/w 1,20/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
iotAnalytics	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
channelArn	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。

名稱	類型	描述
channelName	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。
roleArn	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
iotEvents	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
inputName	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
messageId	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
roleArn	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。 ("Action":"iotevents:BatchPutMessage")。
stepFunctions	StepFunctionsAction	開始執行 Step Functions 狀態機器。
executionNamePrefix	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
stateMachineName	string	Step Functions 狀態機器的名稱會開始執行。
roleArn	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。
ruleDisabled	布林值	指定規則是否停用。
awsIotSqlVersion	string	評估規則時所用的 SQL 規則引擎版本。
errorAction	動作	發生錯誤時欲採取的動作。
dynamoDB	DynamoDBAction	寫入至 DynamoDB 資料表。
tableName	string	DynamoDB 資料表的名稱。
roleArn	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。

名稱	類型	描述
operation	string	欲執行的操作類型。其遵循的是替代範本，因此能為 `$ operation`，但替代對象需產生下列其中之一：INSERT、UPDATE 或 DELETE。
hashKeyField	string	雜湊索引鍵名稱。
hashKeyValue	string	雜湊索引鍵值。
hashKeyType	string	雜湊索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
rangeKeyField	string	範圍索引鍵名稱。
rangeKeyValue	string	範圍索引鍵值。
rangeKeyType	string	範圍索引鍵類型。有效值為「STRING」或「NUMBER」 列舉：STRING | NUMBER
payloadField	string	動作承載。此名稱可自訂。
dynamoDBv2	DynamoDBv2Action	寫入至 DynamoDB 資料表。此為 DynamoDB 動作的新版本，可讓您將 MQTT 訊息承載的每一屬性寫入至個別 DynamoDB 欄位。
roleArn	string	授予 DynamoDB 資料表存取權限的 IAM 角色 ARN。
putItem	PutItemInput	指定訊息資料將寫入的 DynamoDB 資料表。例如： `{ "dynamoDBv2": { "roleArn": "aws:iam:12341251:my-role" "putItem": { "tableName": "my-table" } } }` 訊息承載的每個屬性都將寫入 DynamoDB 資料庫不同的欄中。
tableName	string	將寫入訊息資料的資料表。
lambda	LambdaAction	呼叫 Lambda 函數。
functionArn	string	Lambda 函式的 ARN。
sns	SnsAction	發佈至 Amazon SNS 主題。
targetArn	string	SNS 主題的 ARN。
roleArn	string	授予存取權限的 IAM 角色 ARN。

名稱	類型	描述
messageFormat	string	(選用) 欲發佈訊息的格式。可接受的值為「JSON」和「RAW」。該屬性的預設值為「RAW」。SNS 會使用此設定來判斷是否應剖析承載，以及是否應擷取承載與特定平台相關的部分。若要閱讀更多關於 SNS 訊息格式的資訊，請至 https://docs.aws.amazon.com/sns/latest/dg/json-formats.html 參閱其官方文件。 列舉：RAW | JSON
sqs	SqsAction	發佈至 Amazon SQS 佇列。
roleArn	string	授予存取權限的 IAM 角色 ARN。
queueUrl	string	Amazon SQS 佇列的 URL。
useBase64	布林值	指定是否使用 Base64 編碼。
kinesis	KinesisAction	將資料寫入至 Amazon Kinesis 串流。
roleArn	string	授予 Amazon Kinesis 串流存取權限的 IAM 角色 ARN。
streamName	string	Amazon Kinesis 串流的名稱。
partitionKey	string	分割區索引鍵。
republish	RepublishAction	發佈至另一個 MQTT 主題。
roleArn	string	授予存取權限的 IAM 角色 ARN。
topic	string	MQTT 主題名稱。
qos	integer 範圍 - 上限：1；下限：0	重新發佈訊息時使用服務品質 (QoS) 層級。預設值為 0.
s3	S3Action	寫入至 Amazon S3 儲存貯體。
roleArn	string	授予存取權限的 IAM 角色 ARN。
bucketName	string	Amazon S3 儲存貯體。
key	string	物件金鑰。

名稱	類型	描述
`cannedAcl`	string	Amazon S3 固定的 ACL，負責控制物件金鑰辨識出之物件的存取權限。如需詳細資訊，請參閱 S3 固定 ACL 相關文章。 列舉：private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control | log-delivery-write
`firehose`	FirehoseAction	寫入至 Amazon Kinesis Firehose 串流。
`roleArn`	string	授予 Amazon Kinesis Firehose 串流存取權限的 IAM 角色。
`deliveryStreamName`	string	交付串流名稱。
`separator`	string 模式：([ ])|( )|(,)	用於將寫入 Firehose 串流的記錄區隔開來的字元分隔符。有效值為：'\n' (換行)、'\t' (Tab 鍵)、'\r \n' (Windows 換行)、',' (逗號)。
`cloudwatchMetric`	CloudwatchMetricAction	擷取 CloudWatch 指標。
`roleArn`	string	允許存取 CloudWatch 指標的 IAM 角色。
`metricNamespace`	string	CloudWatch 指標命名空間名稱。
`metricName`	string	CloudWatch 指標名稱。
`metricValue`	string	CloudWatch 指標值。
`metricUnit`	string	CloudWatch 支援的指標單位。
`metricTimestamp`	string	Unix 時間戳記 (選用)。
`cloudwatchAlarm`	CloudwatchAlarmAction	變更 CloudWatch 警示的狀態。
`roleArn`	string	允許存取 CloudWatch 警示的 IAM 角色。
`alarmName`	string	CloudWatch 警示名稱。
`stateReason`	string	警示變更的原因。
`stateValue`	string	警示狀態的值。可接受的值為：OK、ALARM、INSUFFICIENT_DATA。
`elasticsearch`	ElasticsearchAction	將資料寫入至 Amazon Elasticsearch Service 網域。
`roleArn`	string	能夠存取 Elasticsearch 的 IAM 角色 ARN。

名稱	類型	描述
`endpoint`	string 模式：https?://.*	您 Elasticsearch 網域的端點。
`index`	string	欲用來存放資料的 Elasticsearch 索引。
`type`	string	欲存放文件的類型。
`id`	string	欲存放文件的唯一識別符。
`salesforce`	SalesforceAction	傳送訊息到 Salesforce IoT 雲端輸入串流。
`token`	string 長度下限：40	用於驗證 Salesforce IoT 雲端輸入串流存取的符記。建立輸入串流後，Salesforce IoT 雲端平台會提供該符記。
`url`	string 長度上限：2000 pattern: https://ingestion-[a-zA-Z0-9]{1,12}.[a-zA-Z0-9]+.((sfdc-matrix.net)|(sfdcnow.com))/streams/w *1,20*/w *1,20*/event	Salesforce IoT 雲端輸入串流所公開的 URL。建立輸入串流後，Salesforce IoT 雲端平台會提供該 URL。
`iotAnalytics`	IotAnalyticsAction	傳送訊息資料至 AWS IoT Analytics 頻道。
`channelArn`	string	(已棄用) 要傳送訊息資料的 IoT Analytics 頻道的 ARN。
`channelName`	string	要傳送訊息資料的 IoT Analytics 頻道的名稱。
`roleArn`	string	政策允許 IoT Analytics 透過 IoT Analytics (iotanalytics:BatchPutMessage) 傳送訊息資料之角色的 ARN。
`iotEvents`	IotEventsAction	將輸入傳送至 AWS IoT 事件偵錯工具。
`inputName`	string 長度上限：128；下限：1	AWS IoT 事件輸入的名稱。
`messageId`	string 長度上限：128	[選用] 使用此項以確保只有一個指定 messageId 的輸入 (訊息) 會由 AWS IoT 事件偵錯工具進行處理。
`roleArn`	string	授予 AWS IoT 將輸入傳送到 AWS IoT 事件偵錯工具許可的角色 ARN。("Action":"iotevents:BatchPutMessage")。

名稱	類型	描述
stepFunctions	StepFunctionsAction	開始執行 Step Functions 狀態機器。
executionNamePrefix	string	(選用) 將指定名稱給狀態機器，其中包含此字首後接 UUID。如果未提供名稱，Step Functions 會自動建立唯一的名稱給每個狀態機器。
stateMachineName	string	Step Functions 狀態機器的名稱會開始執行。
roleArn	string	授與物聯網 (IoT) 許可以開始執行狀態機器角色的 ARN ("Action":"states:StartExecution")。

輸出

無

錯誤

`SqlParseException`

無法正確剖析 SQL 表達式規則。

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

`UnauthorizedException`

未授權您執行此操作。

`ConflictingResourceUpdateException`

資源更新例外狀況相衝突。當兩個待定更新造成衝突時，就會擲出此例外狀況。

SearchIndex

查詢搜尋索引。

概要

```
aws iot  search-index \
    [--index-name <value>] \
    --query-string <value> \
    [--next-token <value>] \
    [--max-results <value>] \
    [--query-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "indexName": "string",
  "queryString": "string",
  "nextToken": "string",
  "maxResults": "integer",
  "queryVersion": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`indexName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	搜尋索引名稱。
`queryString`	string 長度下限：1	搜尋的查詢字串。
`nextToken`	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。
`maxResults`	integer 範圍 - 上限：500；下限：1	一次可回傳結果的數量上限。
`queryVersion`	string	查詢的版本。

輸出

```
{
  "nextToken": "string",
  "things": [
    {
      "thingName": "string",
      "thingId": "string",
      "thingTypeName": "string",
      "thingGroupNames": [
        "string"
      ],
      "attributes": {
        "string": "string"
      },
      "shadow": "string",
      "connectivity": {
        "connected": "boolean",
        "timestamp": "long"
      }
    }
  ],
  "thingGroups": [
    {
      "thingGroupName": "string",
      "thingGroupId": "string",
      "thingGroupDescription": "string",
      "attributes": {
```

```
        "string": "string"
      },
      "parentGroupNames": [
        "string"
      ]
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
nextToken	string	用於取得下一組結果的符記，如果沒有其他結果則為 null。
things	列出 成員：ThingDocument Java 類別：java.util.List	符合搜尋查詢的物件。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
thingId	string	物件 ID。
thingTypeName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
thingGroupNames	列出 成員：ThingGroupName Java 類別：java.util.List	物件群組名稱。
attributes	對應	屬性。
shadow	string	影子。
connectivity	ThingConnectivity	指出事物是否連接至 AWS IoT 服務。
connected	布林值	若事物已連接至 AWS IoT 服務，則為 true；若未連接則為 false。
timestamp	長整數	物件上次連線或中斷連線的 epoch 時間 (毫秒)。若該事物已超過數週未連線，則時間值可能遺失。
thingGroups	列出 成員：ThingGroupDocument Java 類別：java.util.List	符合搜尋查詢的實物群組。

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件群組名稱。
`thingGroupId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9-]+	物件群組 ID。
`thingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
`attributes`	對應	實物群組屬性。
`parentGroupNames`	列出 成員：ThingGroupName Java 類別：java.util.List	父群組名稱。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`InvalidQueryException`

查詢無效。

`IndexNotReadyException`

索引尚未就緒。

SetDefaultAuthorizer

設定預設授權方。若透過 WebSocket 進行連線且未指定授權方，將使用此授權方。

概要

```
aws iot  set-default-authorizer \
    --authorizer-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。

輸出

```
{
  "authorizerName": "string",
  "authorizerArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerArn`	string	授權方 ARN。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceAlreadyExistsException`

此資源已存在。

SetDefaultPolicyVersion

將指定政策的指定版本做為政策的預設 (實施) 版本。此動作會影響本政策連接的所有憑證。若要列出本政策連接的委託人，請使用 ListPrincipalPolicy API。

概要

```
aws iot  set-default-policy-version \
    --policy-name <value> \
    --policy-version-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "policyName": "string",
  "policyVersionId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`policyVersionId`	string 模式：[0-9]+	政策版本 ID。

輸出

無

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

SetLoggingOptions

設定記錄選項。

注意：不建議使用此命令。請改用 `SetV2LoggingOptions`。

概要

```
aws iot  set-logging-options \
    --logging-options-payload <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "loggingOptionsPayload": {
    "roleArn": "string",
    "logLevel": "string"
  }
}
```

`cli-input-json` 欄位

名稱	類型	描述
`loggingOptionsPayload`	LoggingOptionsPayload	記錄選項承載。
`roleArn`	string	授予存取權限的 IAM 角色 ARN。
`logLevel`	string	日誌層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED

輸出

無

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

SetV2LoggingLevel

設定記錄層級。

概要

```
aws iot  set-v2-logging-level \
    --log-target <value> \
    --log-level <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "logTarget": {
    "targetType": "string",
    "targetName": "string"
  },
  "logLevel": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`logTarget`	LogTarget	日誌目標。
`targetType`	string	目標類型。 列舉：DEFAULT | THING_GROUP
`targetName`	string	目標名稱。
`logLevel`	string	日誌層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED

輸出

無

錯誤

`InternalException`

出現未預期的錯誤。

`NotConfiguredException`

資源未設定。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

SetV2LoggingOptions

設定 V2 記錄服務的記錄選項。

概要

```
aws iot  set-v2-logging-options \
    [--role-arn <value>] \
    [--default-log-level <value>] \
    [--disable-all-logs | --no-disable-all-logs]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleArn": "string",
  "defaultLogLevel": "string",
  "disableAllLogs": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`roleArn`	string	角色的 ARN 允許 IoT 寫入 CloudWatch Logs。
`defaultLogLevel`	string	預設記錄層級。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
`disableAllLogs`	布林值	如果為 true 則停用所有日誌為。預設值為 false。

輸出

無

錯誤

`InternalException`

出現未預期的錯誤。

`InvalidRequestException`

請求的內容無效。

`ServiceUnavailableException`

此服務暫時無法使用。

StartAuditMitigationActionsTask

啟動將一組緩解動作套用至指定目標的任務。

概要

```
aws iot  start-audit-mitigation-actions-task \
    --task-id <value> \
    --target <value> \
    --audit-check-to-actions-mapping <value> \
    --client-request-token <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string",
  "target": {
    "auditTaskId": "string",
    "findingIds": [
      "string"
    ],
    "auditCheckToReasonCodeFilter": {
      "string": [
        "string"
      ]
    }
  },
  "auditCheckToActionsMapping": {
    "string": [
      "string"
    ]
  },
  "clientRequestToken": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	任務的唯一識別符。您可以使用此識別符來檢查任務的狀態或取消任務。
`target`	AuditMitigationActionsTaskTarget	指定稽核結果套用到哪些緩解行動。您可以將它們套用到一個類型的稽核檢查、到來自稽核的所

名稱	類型	描述
		有發現結果，或到一組特定的發現結果。
`auditTaskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	如果任務將從特定稽核對發現結果套用緩解動作，則此值可唯一識別稽核。
`findingIds`	列出 成員：FindingId	如果任務將套用緩解動作到一或多個列出的發現結果，這個值可唯一識別其發現結果。
`auditCheckToReasonCodeFilter`	對應	以稽核檢查與一組原因代碼的形式指定篩選條件，該原因代碼可識別來自稽核緩解動作任務適用之稽核的發現結果。
`auditCheckToActionsMapping`	對應	針對稽核檢查，指定要套用的緩解動作。這些動作必須在您的 AWS 帳戶中定義。
`clientRequestToken`	string 長度上限：64；下限：1 模式：^[a-zA-Z0-9-_]+$	每個稽核緩解任務都必須有唯一的用戶端請求字符。如果您嘗試使用與已存在的任務相同的字符啟動新任務，則會發生例外狀況。如果您省略這個值，會自動產生唯一用戶端請求字符。

輸出

```
{
  "taskId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`taskId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	稽核緩解任務的唯一識別符。這符合您在請求中指定的 `taskId`。

錯誤

`InvalidRequestException`

請求的內容無效。

`TaskAlreadyExistsException`

如果您嘗試開始已存在相同任務 id 的任務，但是不同的 clientRequestToken，則會發生此例外狀況。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

StartNextPendingJobExecution

取得並開始物件下一個待定 (狀態為 IN_PROGRESS 或 QUEUED) 的工作執行。

概要

```
aws iot-jobs-data  start-next-pending-job-execution \
    --thing-name <value> \
    [--status-details <value>] \
    [--step-timeout-in-minutes <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "statusDetails": {
    "string": "string"
  },
  "stepTimeoutInMinutes": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	與裝置關聯的物件名稱。
`statusDetails`	對應	名稱/值對的集合，能夠描述工作執行的狀態。如果未指定，則 statusDetails 不會改變。
`stepTimeoutInMinutes`	長整數	指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (藉由呼叫 `UpdateJobExecution`，將狀態設為 `IN_PROGRESS`，並在 `stepTimeoutInMinutes` 欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會自動設為 `TIMED_OUT`。請注意，設定步驟逾時不會影響可能已在建立任務時指定的進行中逾時 (使用欄位 `timeoutConfig` 的 `CreateJob`)。

名稱	類型	描述
		此參數的有效值範圍是 1 到 10080 (1 分鐘到 7 天)。

輸出

```
{
  "execution": {
    "jobId": "string",
    "thingName": "string",
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "queuedAt": "long",
    "startedAt": "long",
    "lastUpdatedAt": "long",
    "approximateSecondsBeforeTimedOut": "long",
    "versionNumber": "long",
    "executionNumber": "long",
    "jobDocument": "string"
  }
}
```

CLI 輸出欄位

名稱	類型	描述
`execution`	JobExecution	JobExecution 物件。
`jobId`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	您在建立此工作時所指派的獨特識別符。
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	執行工作的物件名稱。
`status`	string	工作執行的狀態。可以是這些狀態其中之一：「QUEUED」、「IN_PROGRESS」、「FAILED」、「SUCCESS」、「CANCELED」、「TIMED_OUT」、「REJECTED」或「REMOVED」。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
`statusDetails`	對應	名稱/值對的集合，能夠描述工作執行的狀態。

名稱	類型	描述
queuedAt	長整數	任務排入佇列的 epoch 時間，以秒為單位。
startedAt	長整數	任務執行開始的 epoch 時間，以秒為單位。
lastUpdatedAt	長整數	任務執行最近一次更新的 epoch 時間，以秒為單位。
approximateSecondsBeforeTimedOut	長整數	將任務執行狀態變更為 TIMED_OUT 之前要保留狀態的預估秒數。實際任務執行逾時可能會比預估持續時間最多晚 60 秒發生。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
executionNumber	長整數	可識別在特定裝置上之特定工作執行的編號。這可在隨後於命令中使用，以傳回或更新工作執行的資訊。
jobDocument	string 長度上限：32768	工作文件的內容。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

`CertificateValidationException`

憑證無效。

StartOnDemandAuditTask

開始執行 Device Defender 稽核。

概要

```
aws iot  start-on-demand-audit-task \
```

```
    --target-check-names <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "targetCheckNames": [
    "string"
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`targetCheckNames`	列出 成員：AuditCheckName	稽核期間執行哪些檢查。必須為您的帳戶啟用您指定的檢查，或發生例外。使用 `DescribeAccountAuditConfiguration` 查看所有檢查清單，包括已啟用的，或使用 `UpdateAccountAuditConfiguration` 選取啟用哪些檢查。

輸出

```
{
  "taskId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`taskId`	string 長度上限：40；下限：1 模式：[a-zA-Z0-9-]+	您開始執行之隨需稽核的 ID。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

StartThingRegistrationTask

建立大量物件佈建任務。

概要

```
aws iot  start-thing-registration-task \
    --template-body <value> \
    --input-file-bucket <value> \
    --input-file-key <value> \
    --role-arn <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "templateBody": "string",
  "inputFileBucket": "string",
  "inputFileKey": "string",
  "roleArn": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`templateBody`	string	該佈建範本。
`inputFileBucket`	string 長度上限：256；下限：3 模式：[a-zA-Z0-9._-]+	內含輸入檔案的 S3 儲存貯體。
`inputFileKey`	string 長度上限：1024；下限：1 模式：[a-zA-Z0-9!_.*'()-/]+	S3 儲存貯體內的輸入檔案名稱。該檔案包含換行分隔的 JSON 檔案，每行均包含佈建單一裝置 (物件) 的參數值。
`roleArn`	字串 長度上限：2048；下限：20	授予輸入檔案許可權限的 IAM 角色 ARN。

輸出

```
{
  "taskId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`taskId`	string	大量物件佈建任務 ID。

名稱	類型	描述
	長度上限：40	

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

StopThingRegistrationTask

取消大量物件佈建任務。

概要

```
aws iot  stop-thing-registration-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "taskId": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`taskId`	string 長度上限：40	大量物件佈建任務 ID。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

TagResource

新增或修改特定資源的標籤。標籤是可用來管理資源的中繼資料。

概要

```
aws iot  tag-resource \
    --resource-arn <value> \
    --tags <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "resourceArn": "string",
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

`cli-input-json` 欄位

名稱	類型	描述
`resourceArn`	string	資源的 ARN。
`tags`	列出 成員：Tag Java 類別：java.util.List	欲新增或修改的資源標籤。
`Key`	string	標籤的金鑰。
`Value`	string	標籤的值。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`LimitExceededException`

已超出限制。

TestAuthorization

測試指定的委託人是否授權在指定資源上執行 AWS IoT 動作。使用此方法來測試並對連接到 AWS IoT 裝置閘道的裝置之授權行為進行偵錯。

概要

```
aws iot  test-authorization \
    [--principal <value>] \
    [--cognito-identity-pool-id <value>] \
    --auth-infos <value> \
    [--client-id <value>] \
    [--policy-names-to-add <value>] \
    [--policy-names-to-skip <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "principal": "string",
  "cognitoIdentityPoolId": "string",
  "authInfos": [
    {
      "actionType": "string",
      "resources": [
        "string"
      ]
    }
  ],
  "clientId": "string",
  "policyNamesToAdd": [
    "string"
  ],
  "policyNamesToSkip": [
    "string"
```

```
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
principal	string	委託人。
cognitoIdentityPoolId	string	Cognito Identity Pool 的 ID。
authInfos	列出 成員：AuthInfo	授權資訊物件清單。模擬授權將針對清單中的每一 authInfo物件建立回應。
actionType	string	正授權委託人的動作類型。 列舉：PUBLISH | SUBSCRIBE | RECEIVE | CONNECT
resources	列出 成員：Resource	正授權委託人以執行指定動作的資源。
clientId	string	MQTT 用戶端 ID。
policyNamesToAdd	列出 成員：PolicyName Java 類別：java.util.List	測試自訂授權時，此處指定的政策會視為已連接至正在授權的委託人。
policyNamesToSkip	列出 成員：PolicyName Java 類別：java.util.List	測試自訂授權時，此處指定的政策會視為尚未連接至正在授權的委託人。

輸出

```
{
  "authResults": [
    {
      "authInfo": {
        "actionType": "string",
        "resources": [
          "string"
        ]
      },
      "allowed": {
        "policies": [
          {
            "policyName": "string",
            "policyArn": "string"
          }
        ]
      },
      "denied": {
        "implicitDeny": {
          "policies": [
            {
```

```
            "policyName": "string",
            "policyArn": "string"
          }
        ]
      },
      "explicitDeny": {
        "policies": [
          {
            "policyName": "string",
            "policyArn": "string"
          }
        ]
      }
    },
    "authDecision": "string",
    "missingContextValues": [
      "string"
    ]
  }
]
}
```

CLI 輸出欄位

名稱	類型	描述
authResults	列出 成員：AuthResult	身分驗證結果。
authInfo	AuthInfo	授權資訊。
actionType	string	正授權委託人的動作類型。 列舉：PUBLISH | SUBSCRIBE | RECEIVE | CONNECT
resources	列出 成員：Resource	正授權委託人以執行指定動作的資源。
allowed	Allowed	允許指定動作的政策和陳述式。
policies	列出 成員：Policy Java 類別：java.util.List	允許身分驗證的政策清單。
policyName	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
policyArn	string	政策 ARN。
denied	Denied	拒絕指定動作的政策和陳述式。
implicitDeny	ImplicitDeny	隱含拒絕授權的資訊。當政策未明確拒絕或允許對資源的動作，將視為隱含拒絕。

名稱	類型	描述
`policies`	列出 成員：Policy Java 類別：java.util.List	針對特定資源所進行的特定動作，未含符合允許或拒絕陳述式的政策。
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`policyArn`	string	政策 ARN。
`explicitDeny`	ExplicitDeny	明確拒絕授權的資訊。
`policies`	列出 成員：Policy Java 類別：java.util.List	拒絕授權的政策。
`policyName`	string 長度上限：128；下限：1 模式：[w+=,.@-]+	政策名稱。
`policyArn`	string	政策 ARN。
`authDecision`	string	此情境的最終授權決策。進行授權決策時，會考量多個陳述式。明確拒絕的陳述式能夠覆寫多個允許陳述式。 列舉：ALLOWED | EXPLICIT_DENY | IMPLICIT_DENY
`missingContextValues`	列出 成員：MissingContextValue Java 類別：java.util.List	包含評估政策時發現的任何缺少的內容值。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`LimitExceededException`

已超出限制。

TestInvokeAuthorizer

叫用指定自訂授權方來測試自訂授權行為。使用此方法來測試並對連接到 AWS IoT 裝置閘道的裝置之自訂授權行為進行偵錯。

概要

```
aws iot  test-invoke-authorizer \
    --authorizer-name <value> \
    --token <value> \
    --token-signature <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string",
  "token": "string",
  "tokenSignature": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	自訂授權方名稱。
`token`	string 長度上限：6144；下限：1	自訂身分驗證服務回傳的符記。
`tokenSignature`	string 長度上限：2560；下限：1 模式：[A-Za-z0-9+/]+={0,2}	符記與您自訂身分驗證服務的私有金鑰完成的簽章。

輸出

```
{
  "isAuthenticated": "boolean",
  "principalId": "string",
  "policyDocuments": [
    "string"
  ],
  "refreshAfterInSeconds": "integer",
  "disconnectAfterInSeconds": "integer"
}
```

CLI 輸出欄位

名稱	類型	描述
`isAuthenticated`	布林值	若符記已驗證則此值為 true，否則為 false。
`principalId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9]+	委託人 ID。
`policyDocuments`	列出 成員：PolicyDocument	IAM 政策文件。
`refreshAfterInSeconds`	integer	暫時登入資料重新整理後的等待秒數。
`disconnectAfterInSeconds`	integer	連線終止後的等待秒數。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`InvalidResponseException`

回應無效。

TransferCertificate

將指定憑證傳送至指定 AWS 帳戶。

收件人確認之前，您可以取消傳送。

不會發送通知給傳送目的地帳戶，發起人可決定是否通知傳送目標。

所傳送的憑證不得處於 ACTIVE 狀態，您可使用 UpdateCertificate API 來加以停用。

憑證不得連接至任何政策，您可使用 DetachPrincipalPolicy API 以將兩者分離。

概要

```
aws iot  transfer-certificate \
    --certificate-id <value> \
    --target-aws-account <value> \
    [--transfer-message <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string",
  "targetAwsAccount": "string",
  "transferMessage": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)
`targetAwsAccount`	string 長度上限：12；下限：12 模式：[0-9]+	AWS 帳戶。
`transferMessage`	string 長度上限：128	傳輸的訊息。

輸出

```
{
  "transferredCertificateArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`transferredCertificateArn`	string	憑證的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`CertificateStateException`

不允許此憑證操作。

`TransferConflictException`

憑證無法傳送，因為仍連接至授權政策。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UntagResource

從資源移除特定標籤 (中繼資料)。

概要

```
aws iot  untag-resource \
    --resource-arn <value> \
    --tag-keys <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "resourceArn": "string",
  "tagKeys": [
    "string"
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
resourceArn	string	資源的 ARN。
tagKeys	列出 成員：TagKey Java 類別：java.util.List	要從資源移除的標籤金鑰清單。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

UpdateAccountAuditConfiguration

為此帳戶設定或重新設定 Device Defender 稽核設定。設定值包括如何傳送稽核通知，以及啟用或停用哪些稽核檢查。

概要

```
aws iot  update-account-audit-configuration \
    [--role-arn <value>] \
    [--audit-notification-target-configurations <value>] \
    [--audit-check-configurations <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
    "string": {
```

```
      "enabled": "boolean"
    }
  }
}
```

cli-input-json 欄位

名稱	類型	描述
roleArn	字串 長度上限：2048；下限：20	角色的 ARN 授與 AWS IoT 許可，可在執行稽核時存取裝置、政策、憑證和其他必要項目的相關資訊。
auditNotificationTargetConfigurations	對應	有關傳送稽核通知的目標的資訊。
targetArn	string	傳送稽核通知的目標 ARN (SNS 主題)。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，以傳送通知至目標。
enabled	布林值	若啟用通知至目標則為 True。
auditCheckConfigurations	對應	指定為此帳戶啟用和停用哪些稽核檢查。使用 DescribeAccountAuditConfiguration 查看所有檢查清單，其中包括目前已啟用的。 有些資料收集可能會在啟用特定檢查時立即開始。當停用檢查時，目前為止與檢查相關的任何收集資料會加以刪除。 若是由任何排定的稽核使用，則不能停用檢查。您必須先從排定的稽核刪除檢查，或刪除排定的稽核本身。 在對 UpdateAccountAuditConfiguration 的第一個呼叫，需使用此參數，且必須指定至少一個啟用的檢查。
enabled	布林值	若為此帳戶啟用此稽核檢查，則為 True。

輸出

無

錯誤

InvalidRequestException

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

UpdateAuthorizer

更新授權方。

概要

```
aws iot  update-authorizer \
    --authorizer-name <value> \
    [--authorizer-function-arn <value>] \
    [--token-key-name <value>] \
    [--token-signing-public-keys <value>] \
    [--status <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "authorizerName": "string",
  "authorizerFunctionArn": "string",
  "tokenKeyName": "string",
  "tokenSigningPublicKeys": {
    "string": "string"
  },
  "status": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerFunctionArn`	string	授權方 Lambda 函數的 ARN。
`tokenKeyName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	用於自 HTTP 標頭擷取符記的金鑰。
`tokenSigningPublicKeys`	對應	用於驗證符記簽章的公開金鑰。
`status`	string	所更新授權方的要求狀態。 列舉：ACTIVE | INACTIVE

輸出

```
{
  "authorizerName": "string",
  "authorizerArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`authorizerName`	string 長度上限：128；下限：1 模式：[w=,@-]+	授權方名稱。
`authorizerArn`	string	授權方 ARN。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`LimitExceededException`

已超出限制。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UpdateBillingGroup

更新關於帳單群組的資訊。

概要

```
aws iot  update-billing-group \
    --billing-group-name <value> \
    --billing-group-properties <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "billingGroupName": "string",
  "billingGroupProperties": {
    "billingGroupDescription": "string"
  },
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`billingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	帳單群組的名稱。
`billingGroupProperties`	BillingGroupProperties	帳單群組的屬性。
`billingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	帳單群組的描述。
`expectedVersion`	長整數	帳單群組的預期版本。若帳單群組版本與請求中指定的預期版本不符，則會使用 `VersionConflictException` 來拒絕 `DeleteBillingGroup` 請求。

輸出

```
{
  "version": "long"
}
```

CLI 輸出欄位

名稱	類型	描述
`version`	長整數	帳單群組的最新版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

UpdateCACertificate

更新已註冊的 CA 憑證。

概要

```
aws iot  update-ca-certificate \
    --certificate-id <value> \
    [--new-status <value>] \
    [--new-auto-registration-status <value>] \
    [--registration-config <value>] \
    [--remove-auto-registration | --no-remove-auto-registration]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "certificateId": "string",
  "newStatus": "string",
  "newAutoRegistrationStatus": "string",
  "registrationConfig": {
    "templateBody": "string",
    "roleArn": "string"
  },
  "removeAutoRegistration": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`certificateId`	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	CA 憑證識別符。
`newStatus`	string	CA 憑證的更新狀態。 備註：狀態值 REGISTER_INACTIVE 已棄用，請不要使用。 列舉：ACTIVE | INACTIVE
`newAutoRegistrationStatus`	string	自動註冊狀態的新值。有效值為：「ENABLE」或「DISABLE」。 列舉：ENABLE | DISABLE

名稱	類型	描述
registrationConfig	RegistrationConfig	註冊組態的資訊。
templateBody	string	範本本身。
roleArn	字串 長度上限：2048；下限：20	此角色的 ARN。
removeAutoRegistration	布林值	若為 true，則移除自動註冊。

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

InternalFailureException

出現未預期的錯誤。

UpdateCertificate

更新指定憑證的狀態。此為等冪操作。

改變憑證的 ACTIVE 狀態 (包括 REVOKED)，不會中斷目前的連線裝置，但這些裝置將無法重新連線。

必須為 ACTIVE 狀態，才可驗證使用憑證連接至 AWS IoT 的裝置。

概要

```
aws iot  update-certificate \
    --certificate-id <value> \
    --new-status <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
```

```
  "certificateId": "string",
  "newStatus": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
certificateId	string 長度上限：64；下限：64 模式：(0x)?[a-fA-F0-9]+	憑證的 ID。(憑證 ARN 的最後一部分內含憑證 ID。)
newStatus	string	新的狀態。 備註：將狀態設定為 PENDING_TRANSFER，將導致擲出例外狀況。PENDING_TRANSFER 為 AWS IoT 內部使用的狀態，不適用開發人員。 備註：狀態值 REGISTER_INACTIVE 已棄用，請不要使用。 列舉：ACTIVE | INACTIVE | REVOKED | PENDING_TRANSFER | REGISTER_INACTIVE | PENDING_ACTIVATION

輸出

無

錯誤

ResourceNotFoundException

指定的資源不存在。

CertificateStateException

不允許此憑證操作。

InvalidRequestException

請求的內容無效。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UpdateDynamicThingGroup

更新動態物件群組。

概要

```
aws iot  update-dynamic-thing-group \
    --thing-group-name <value> \
    --thing-group-properties <value> \
    [--expected-version <value>] \
    [--index-name <value>] \
    [--query-string <value>] \
    [--query-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "thingGroupProperties": {
    "thingGroupDescription": "string",
    "attributePayload": {
      "attributes": {
        "string": "string"
      },
      "merge": "boolean"
    }
  },
  "expectedVersion": "long",
  "indexName": "string",
  "queryString": "string",
  "queryVersion": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲更新的動態物件群組名稱。
`thingGroupProperties`	ThingGroupProperties	欲更新的動態物件群組屬性。
`thingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
`attributePayload`	AttributePayload	JSON 格式的物件群組屬性。

名稱	類型	描述
attributes	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： `\"attributes\": {\"string1\": \"string2\"}`
merge	布林值	指定 AttributePayload提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 UpdateThing 加上一個空白屬性值。 Note merge 屬性僅於呼叫 UpdateThing 或 UpdateThingGroup 時有效。
expectedVersion	長整數	欲更新的動態物件群組預期版本。
indexName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲更新的動態物件群組索引。 Note 目前僅支援一個索引：'AWS_Things'。
queryString	string 長度下限：1	欲更新的動態物件群組搜尋查詢字串。
queryVersion	string	欲更新的動態物件群組查詢版本。 Note 目前僅支援一個查詢版本："2017-09-30"。如果未指定，則查詢版本預設為此值。

輸出

```
{
  "version": "long"
}
```

CLI 輸出欄位

名稱	類型	描述
version	長整數	動態物件群組版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

`InvalidQueryException`

查詢無效。

UpdateEventConfigurations

更新事件組態。

概要

```
aws iot  update-event-configurations \
    [--event-configurations <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "eventConfigurations": {
    "string": {
      "Enabled": "boolean"
    }
  }
}
```

`cli-input-json` 欄位

名稱	類型	描述
`eventConfigurations`	對應	新的事件組態值。
`Enabled`	布林值	True 將會啟用此設定。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`InternalFailureException`

出現未預期的錯誤。

`ThrottlingException`

請求頻率超過限制。

UpdateIndexingConfiguration

更新搜尋組態。

概要

```
aws iot  update-indexing-configuration \
    [--thing-indexing-configuration <value>] \
    [--thing-group-indexing-configuration <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingIndexingConfiguration": {
    "thingIndexingMode": "string",
    "thingConnectivityIndexingMode": "string"
  },
  "thingGroupIndexingConfiguration": {
    "thingGroupIndexingMode": "string"
  }
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingIndexingConfiguration`	ThingIndexingConfiguration	物件索引組態。
`thingIndexingMode`	string	物件索引模式。有效值為： • REGISTRY - 您的事物索引僅包含登錄檔資料。 • REGISTRY_AND_SHADOW - 您的事物索引包含登錄檔和陰影資料。 • OFF - 物件索引已停用。 列舉：OFF | REGISTRY | REGISTRY_AND_SHADOW
`thingConnectivityIndexingMode`	string	物件連線索引模式。有效值為： • STATUS - 您的事物索引包含連線能力狀態。若要啟用事物連線能力索引建立，則

名稱	類型	描述
		thingIndexMode 必不得設為 OFF。 • OFF - 物件連線狀態索引已停用。 enum: OFF | STATUS
thingGroupIndexingConfiguration	ThingGroupIndexingConfiguration	實物群組索引組態。
thingGroupIndexingMode	string	實物群組索引模式。 列舉：OFF | ON

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UpdateJob

更新指定任務的支援欄位。

概要

```
aws iot  update-job \
    --job-id <value> \
    [--description <value>] \
    [--presigned-url-config <value>] \
    [--job-executions-rollout-config <value>] \
    [--abort-config <value>] \
    [--timeout-config <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "jobId": "string",
```

```
  "description": "string",
  "presignedUrlConfig": {
    "roleArn": "string",
    "expiresInSec": "long"
  },
  "jobExecutionsRolloutConfig": {
    "maximumPerMinute": "integer",
    "exponentialRate": {
      "baseRatePerMinute": "integer",
      "incrementFactor": "double",
      "rateIncreaseCriteria": {
        "numberOfNotifiedThings": "integer",
        "numberOfSucceededThings": "integer"
      }
    }
  },
  "abortConfig": {
    "criteriaList": [
      {
        "failureType": "string",
        "action": "string",
        "thresholdPercentage": "double",
        "minNumberOfExecutedThings": "integer"
      }
    ]
  },
  "timeoutConfig": {
    "inProgressTimeoutInMinutes": "long"
  }
}
```

cli-input-json 欄位

名稱	類型	描述
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	欲更新的任務 ID。
description	string 長度上限：2028 模式：[^\\p{C}]+	關於工作的簡短文字描述。
presignedUrlConfig	PresignedUrlConfig	預先簽章的 S3 URL 之組態資訊。
roleArn	字串 長度上限：2048；下限：20	授予自工作資料/更新儲存位置 S3 儲存貯體下載檔案的許可之 IAM 角色 ARN。此角色也必須授權讓 IoT 下載檔案。
expiresInSec	長整數 範圍 - 上限：3600；下限：60	預先簽章的 URL 有效時間 (以秒為單位)。有效值為 60 - 3600，預設值為 3600 秒。當 Jobs 收到對工作文件的 MQTT 請求，就會產生預先簽章的 URL。
jobExecutionsRolloutConfig	JobExecutionsRolloutConfig	可讓您建立工作的分階段推展。

名稱	類型	描述
`maximumPerMinute`	integer 範圍 - 分鐘：1	每分鐘接收待定工作的物件之數量上限。此參數可讓您建立分階段推展的任務。
`exponentialRate`	ExponentialRolloutRate	任務推展的增加率。此參數可讓您為任務推展定義指數率。
`baseRatePerMinute`	integer 範圍 - 上限：1000；下限：1	在任務推展開始時，每分鐘通知待處理任務的最低物件數量。此參數可讓您定義推展的初始速率。
`rateIncreaseCriteria`	RateIncreaseCriteria	任務啟動增加推展速率的條件。 AWS IoT 最多支援小數點後一位數 (例如，1.5，但不支援 1.55)。
`numberOfNotifiedThings`	integer 範圍 - 分鐘：1	啟動增加推展速率通知物件數量的閾值。
`numberOfSucceededThings`	integer 範圍 - 分鐘：1	啟動增加推展速率成功物件數量的閾值。
`abortConfig`	AbortConfig	可讓您建立中止任務的條件。
`criteriaList`	列出 成員：AbortCriteria Java 類別：java.util.List	用於定義中止任務規則的中止條件清單。
`failureType`	string	用於定義初始化任務中止的任務執行失敗類型。 enum: FAILED | REJECTED | TIMED_OUT | ALL
`action`	string	用於初始化任務中止的中止動作類型。 enum: CANCEL
`minNumberOfExecutedThings`	integer 範圍 - 分鐘：1	評估中止規則之前的已執行物件最低數量。
`timeoutConfig`	TimeoutConfig	指定每個裝置必須完成其任務執行的時間。任務執行狀態設定為 `IN_PROGRESS` 時，計時器即會開始。在時間過期之前，如果任務執行狀態未設定為其他結束狀態，就會自動設為 `TIMED_OUT`。

名稱	類型	描述
inProgressTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間 (以分鐘為單位)。逾時間隔可介於 1 分鐘到 7 天之間 (1 到 10080 分鐘)。進行中的計時器無法更新，並會套用到任務的所有任務執行。每當任務執行維持在 IN_PROGRESS 狀態超過此間隔時，任務執行就會失敗，並切換到結束 TIMED_OUT 狀態。

輸出

無

錯誤

InvalidRequestException

請求的內容無效。

ResourceNotFoundException

指定的資源不存在。

ThrottlingException

請求頻率超過限制。

ServiceUnavailableException

此服務暫時無法使用。

UpdateJobExecution

更新工作執行的狀態。

概要

```
aws iot-jobs-data  update-job-execution \
    --job-id <value> \
    --thing-name <value> \
    --status <value> \
    [--status-details <value>] \
    [--step-timeout-in-minutes <value>] \
    [--expected-version <value>] \
    [--include-job-execution-state | --no-include-job-execution-state] \
    [--include-job-document | --no-include-job-document] \
    [--execution-number <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json 格式

```
{
  "jobId": "string",
  "thingName": "string",
  "status": "string",
  "statusDetails": {
```

```
    "string": "string"
  },
  "stepTimeoutInMinutes": "long",
  "expectedVersion": "long",
  "includeJobExecutionState": "boolean",
  "includeJobDocument": "boolean",
  "executionNumber": "long"
}
```

cli-input-json 欄位

名稱	類型	描述
jobId	string 長度上限：64；下限：1 模式：[a-zA-Z0-9_-]+	此工作在建立時被指派的獨特識別符。
thingName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	與裝置關聯的物件名稱。
status	string	新的工作執行狀態 (IN_PROGRESS、FAILED、SUCCESS 或 REJECTED)。這必須在每次更新時指定。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED
statusDetails	對應	選用。名稱/值對的集合，能夠描述工作執行的狀態。如果未指定，則 statusDetails 不會改變。
stepTimeoutInMinutes	長整數	指定此裝置必須完成這項任務執行的時間。在此計時器到期或計時器重設之前 (同樣地，藉由呼叫 UpdateJobExecution，將狀態設為 IN_PROGRESS，並在此欄位中指定新的逾時值)，如果任務執行狀態未設定為結束狀態，任務執行狀態就會自動設為 TIMED_OUT。請注意，設定或重設步驟逾時不會影響可能已在建立任務時指定的進行中逾時 (使用欄位 timeoutConfig 的 CreateJob)。 此參數的有效值範圍是 1 到 10080 (1 分鐘到 7 天)。-1 值也有效，並會取消目前的步驟計時器 (之前使用 UpdateJobExecutionRequest而建立)。

名稱	類型	描述
expectedVersion	長整數	選用。預期的目前工作執行版本。每次您更新工作執行，其版本編號就會增加。如果存放於 Jobs 的工作執行版本不符，則更新會被 VersionMismatch 錯誤拒絕，並傳回包含目前工作執行狀態資料的 ErrorResponse。(這樣就不必另外執行 DescribeJobExecution 請求以獲得工作執行狀態資料。)
includeJobExecutionState	布林值	選用。若被包括在內並設定為 true，則回應會包含 JobExecutionState 資料。預設值為 false。
includeJobDocument	布林值	選用。當設定為 True，回應就會包含工作文件。預設值為 false。
executionNumber	長整數	選用。可識別在特定裝置上之特定工作執行的編號。

輸出

```
{
  "executionState": {
    "status": "string",
    "statusDetails": {
      "string": "string"
    },
    "versionNumber": "long"
  },
  "jobDocument": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
executionState	JobExecutionState	JobExecutionState 物件。
status	string	工作執行的狀態。可以是這些狀態其中之一：「QUEUED」、「IN_PROGRESS」、「FAILED」、「SUCCESS」、「CANCELED」、「TIMED_OUT」、「REJECTED」或「REMOVED」。 列舉：QUEUED | IN_PROGRESS | SUCCEEDED | FAILED | TIMED_OUT | REJECTED | REMOVED | CANCELED

名稱	類型	描述
statusDetails	對應	名稱/值對的集合，能夠描述工作執行的狀態。
versionNumber	長整數	工作執行的版本。每次裝置更新工作時，工作執行的版本編號就會增加。
jobDocument	string 長度上限：32768	工作文件的內容。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`ServiceUnavailableException`

此服務暫時無法使用。

`CertificateValidationException`

憑證無效。

`InvalidStateTransitionException`

由於工作執行的當前狀態而使嘗試變更工作執行狀態的更新無效 (例如，嘗試變更將 SUCCESS 狀態改為 IN_PROGRESS 的請求)。在此情況下，錯誤訊息的本文也會包含 executionState 欄位。

UpdateMitigationAction

更新指定緩解動作的定義。

概要

```
aws iot  update-mitigation-action \
    --action-name <value> \
    [--role-arn <value>] \
    [--action-params <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "actionName": "string",
  "roleArn": "string",
  "actionParams": {
    "updateDeviceCertificateParams": {
      "action": "string"
    },
```

```
    "updateCACertificateParams": {
      "action": "string"
    },
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "string"
      ],
      "overrideDynamicGroups": "boolean"
    },
    "replaceDefaultPolicyVersionParams": {
      "templateName": "string"
    },
    "enableIoTLoggingParams": {
      "roleArnForLogging": "string",
      "logLevel": "string"
    },
    "publishFindingToSnsParams": {
      "topicArn": "string"
    }
  }
}
```

cli-input-json 欄位

名稱	類型	描述
actionName	string 長度上限：128 模式：[a-zA-Z0-9_-]+	緩解動作的易記名稱。您無法使用 UpdateMitigationAction 變更名稱。反之，您必須使用新名稱刪除並重新建立緩解動作。
roleArn	字串 長度上限：2048；下限：20	用來套用緩解動作的 IAM 角色的 ARN。
actionParams	MitigationActionParams	定義動作的類型和該動作的參數。
updateDeviceCertificateParams	UpdateDeviceCertificateParams	參數定義緩解行動，變更裝置的憑證到未啟用的狀態。
action	string	您要套用到裝置憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
updateCACertificateParams	UpdateCACertificateParams	參數定義緩解行動，變更的 CA 憑證處於未啟用的狀態。
action	string	您要套用到 CA 憑證的動作。唯一支援的值為 DEACTIVATE。 列舉：DEACTIVATE
addThingsToThingGroupParams	AddThingsToThingGroupParams	參數定義緩解行動，將裝置關聯的憑證移動到一或多個指定的物件群組，通常用於隔離。
thingGroupNames	列出 成員：ThingGroupName	觸發緩解行動新增物件的群組清單。您可以新增物件最多到 10 個群組，但是在相同的層級內，您

名稱	類型	描述
		無法讓一個物件加入超過一個群組。
overrideDynamicGroups	布林值	指定此緩解動作是否可以移動觸發緩解動作的實物，即使該實物為一或多個動態物件群組的一部分。
replaceDefaultPolicyVersionParams	ReplaceDefaultPolicyVersionParams	參數定義緩解行動，新增空白政策來限制權限。
templateName	字串	要套用範本的名稱。唯一支援的值為 BLANK_POLICY。 列舉：BLANK_POLICY
enableIoTLoggingParams	EnableIoTLoggingParams	定義可以指定的細節層級啟用 AWS IoT 記錄的緩解動作的參數。
roleArnForLogging	字串 長度上限：2048；下限：20	用於記錄 IAM 角色的 ARN。
logLevel	string	指定要記錄的資訊類型。 列舉：DEBUG | INFO | ERROR | WARN | DISABLED
publishFindingToSnsParams	PublishFindingToSnsParams	用來定義發佈結果到 Amazon SNS 緩解動作的參數。您可以實作自訂的動作來回應 Amazon SNS 訊息。
topicArn	string 長度上限：350	您要發佈結果主題的 ARN。

輸出

```
{
  "actionArn": "string",
  "actionId": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
actionArn	字串	新的緩解行動的 ARN。
actionId	字串	緩解行動的唯一識別符。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

UpdateRoleAlias

更新角色別名。

概要

```
aws iot  update-role-alias \
    --role-alias <value> \
    [--role-arn <value>] \
    [--credential-duration-seconds <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "roleAlias": "string",
  "roleArn": "string",
  "credentialDurationSeconds": "integer"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`roleAlias`	string 長度上限：128；下限：1 模式：[w=,@-]+	欲更新的角色別名。
`roleArn`	字串 長度上限：2048；下限：20	角色 ARN。
`credentialDurationSeconds`	integer 範圍 - 上限：3600；下限：900	登入資料有效秒數。

輸出

```
{
  "roleAlias": "string",
```

```
  "roleAliasArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
roleAlias	string 長度上限：128；下限：1 模式：[w=,@-]+	角色別名。
roleAliasArn	string	角色別名 ARN。

錯誤

`ResourceNotFoundException`

指定的資源不存在。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UpdateScheduledAudit

更新已排程的稽核，包含執行哪些檢查，以及執行稽核的頻率。

概要

```
aws iot  update-scheduled-audit \
    [--frequency <value>] \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    [--target-check-names <value>] \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "frequency": "string",
```

```
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string"
}
```

cli-input-json 欄位

名稱	類型	描述
frequency	string	已排程的稽核執行的頻率。可以是「每日」、「每週」、「每兩週」或「每月」的其中之一。每個稽核的開始時間取決於系統。 列舉：每日 | 每週 | 每兩週 | 每月
dayOfMonth	string 模式：^([1-9]|[12][0-9]|3[01])$\|^LAST$	當月執行已排程稽核的日期。可以是「1」到「31」或「最後一天」。如果「頻率」參數設定為「每月」，則此欄位為必填。如果指定的是 29-31 的日期，但當月卻沒有這幾天時，稽核會在該月的「最後一天」執行。
dayOfWeek	string	當週執行已排程稽核的日期。可以是「週日」、「週一」、「週二」、「週三」、「週四」、「週五」或「週六」的其中之一。如果「頻率」參數設定為「每週」或「每兩週」，則此欄位為必填。 列舉：週日 | 週一 | 週二 | 週三 | 週四 | 週五 | 週六
targetCheckNames	列出 成員：AuditCheckName	已排程的稽核期間執行哪些檢查。必須為您的帳戶啟用檢查。(使用 DescribeAccountAuditConfiguration 查看所有檢查清單，包括已啟用的檢查，或使用 UpdateAccountAuditConfiguration 選取啟用哪些檢查。)
scheduledAuditName	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	已排程的稽核的名稱。(上限128個字元)

輸出

```
{
  "scheduledAuditArn": "string"
}
```

CLI 輸出欄位

名稱	類型	描述
`scheduledAuditArn`	string	已排程的稽核的 ARN。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

UpdateSecurityProfile

更新 Device Defender 安全性描述檔。

概要

```
aws iot  update-security-profile \
    --security-profile-name <value> \
    [--security-profile-description <value>] \
    [--behaviors <value>] \
    [--alert-targets <value>] \
    [--additional-metrics-to-retain <value>] \
    [--delete-behaviors | --no-delete-behaviors] \
    [--delete-alert-targets | --no-delete-alert-targets] \
    [--delete-additional-metrics-to-retain | --no-delete-additional-metrics-to-retain] \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "securityProfileName": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
```

```
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "deleteBehaviors": "boolean",
  "deleteAlertTargets": "boolean",
  "deleteAdditionalMetricsToRetain": "boolean",
  "expectedVersion": "long"
}
```

cli-input-json 欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您要更新之安全性描述檔的名稱。
`securityProfileDescription`	string 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的說明。
`behaviors`	列出 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
`name`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
`metric`	string	行為測量的內容。
`criteria`	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
`comparisonOperator`	string	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或 `statisticalThreshold`) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-

名稱	類型	描述
		than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
`value`	MetricValue	相較於 `metric` 的值。
`count`	長整數 範圍 - 分鐘：0	如果 `comparisonOperator` 呼叫數值，請使用此項來指定相較於 `metric` 的數值。
`cidrs`	列出 成員：Cidr	如果 `comparisonOperator` 呼叫一組 CIDR，請使用此項來指定相較於 `metric` 的組合。
`ports`	列出 成員：Port	如果 `comparisonOperator` 呼叫一組連接埠，請使用此項來指定相較於 `metric` 的組合。
`durationSeconds`	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，`NUM_MESSAGES_SENT`)。對於 `statisticalThreshhold` 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
`consecutiveDatapointsToAlarm`	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
`consecutiveDatapointsToClear`	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
`statisticalThreshold`	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
`statistic`	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (`durationSeconds`) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (`comparisonOperator`)，則該裝置會被視為符合行為，否則就是違反行為。

名稱	類型	描述
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。
alertTargetArn	string	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	列出 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在設定檔 behaviors 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
deleteBehaviors	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 behaviors。如果在目前叫用中定義任何 behaviors，則會發生例外狀況。
deleteAlertTargets	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 alertTargets。如果在目前叫用中定義任何 alertTargets，則會發生例外狀況。
deleteAdditionalMetricsToRetain	布林值	如果為 true，則刪除為此安全性設定檔定義的所有 additionalMetricsToRetain。如果在目前叫用中定義任何 additionalMetricsToRetain，則會發生例外狀況。
expectedVersion	長整數	安全性描述檔的預期版本。只要更新安全性描述檔便產生新版本。如果指定的值與實際版本不同，便擲出 VersionConflictException。

輸出

```
{
  "securityProfileName": "string",
  "securityProfileArn": "string",
  "securityProfileDescription": "string",
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
```

```
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ],
  "alertTargets": {
    "string": {
      "alertTargetArn": "string",
      "roleArn": "string"
    }
  },
  "additionalMetricsToRetain": [
    "string"
  ],
  "version": "long",
  "creationDate": "timestamp",
  "lastModifiedDate": "timestamp"
}
```

CLI 輸出欄位

名稱	類型	描述
`securityProfileName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	已更新之安全性描述檔的名稱。
`securityProfileArn`	string	已更新之安全性描述檔的 ARN。
`securityProfileDescription`	string 長度上限：1000 模式：[\\p{Graph}]*	安全性描述檔的描述。
`behaviors`	列出 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
`name`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
`metric`	string	行為測量的內容。
`criteria`	BehaviorCriteria	可判斷裝置對於 `metric` 的操作是否正確的條件。
`comparisonOperator`	string	將測量的實物 (`metric`) 關聯到條件 (包含 `value` 或

名稱	類型	描述
		statisticalThreshold) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
value	MetricValue	相較於 metric 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，NUM_MESSAGES_SENT)。對於 statisticalThreshhold 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。

名稱	類型	描述
statistic	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (durationSeconds) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (comparisonOperator)，則該裝置會被視為符合行為，否則就是違反行為。
alertTargets	對應	傳送提醒之處。(提醒始終傳送到主控台)。
alertTargetArn	string	傳送提醒所至的通知目標的 ARN。
roleArn	字串 長度上限：2048；下限：20	授與許可的角色 ARN，將提醒傳送至通知目標。
additionalMetricsToRetain	列出 成員：BehaviorMetric	其資料受保留 (存放) 的指標清單。在預設情況下，系統會為在安全性描述檔 behaviors 中所用的任何指標保留資料，但也會為此處指定的任何指標保留資料。
version	長整數	安全性描述檔的更新版本。
creationDate	timestamp	安全性描述檔的建立時間。
lastModifiedDate	timestamp	安全性描述檔的最後建立時間。

錯誤

InvalidRequestException

請求的內容無效。

ResourceNotFoundException

指定的資源不存在。

VersionConflictException

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

ThrottlingException

請求頻率超過限制。

InternalFailureException

出現未預期的錯誤。

UpdateStream

更新現有串流，串流版本會增加一。

概要

```
aws iot  update-stream \
    --stream-id <value> \
    [--description <value>] \
    [--files <value>] \
    [--role-arn <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "streamId": "string",
  "description": "string",
  "files": [
    {
      "fileId": "integer",
      "s3Location": {
        "bucket": "string",
        "key": "string",
        "version": "string"
      }
    }
  ],
  "roleArn": "string"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`streamId`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
`description`	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。
`files`	列出 成員：StreamFile	與串流建立關聯的檔案。
`fileId`	integer 範圍 - 上限：255；下限：0	檔案 ID。
`s3Location`	S3Location	檔案於 S3 中的位置。
`bucket`	string 長度下限：1	S3 儲存貯體。

名稱	類型	描述
key	string 長度下限：1	S3 金鑰。
version	string	S3 儲存貯體版本。
roleArn	字串 長度上限：2048；下限：20	此 IAM 角色允許 IoT 服務委託人存取您的 S3 檔案。

輸出

```
{
  "streamId": "string",
  "streamArn": "string",
  "description": "string",
  "streamVersion": "integer"
}
```

CLI 輸出欄位

名稱	類型	描述
streamId	string 長度上限：128；下限：1 模式：[a-zA-Z0-9_-]+	串流 ID。
streamArn	string	串流 ARN。
description	string 長度上限：2028 模式：[^\\p{C}]+	串流的描述。
streamVersion	integer 範圍 - 上限：65535；下限：0	此串流的版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`ResourceNotFoundException`

指定的資源不存在。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

UpdateThing

更新物件資料。

概要

```
aws iot  update-thing \
    --thing-name <value> \
    [--thing-type-name <value>] \
    [--attribute-payload <value>] \
    [--expected-version <value>] \
    [--remove-thing-type | --no-remove-thing-type]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "thingTypeName": "string",
  "attributePayload": {
    "attributes": {
      "string": "string"
    },
    "merge": "boolean"
  },
  "expectedVersion": "long",
  "removeThingType": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲更新的物件名稱。
`thingTypeName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件類型名稱。
`attributePayload`	AttributePayload	內含名稱/值對之 JSON 字串的物件屬性清單。例如： `\"attributes\": {\"name1\":\"value2\"}`

名稱	類型	描述
		此資料可用來新增屬性或更新現有屬性。
attributes	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如： *\"attributes\": {\"string1\": \"string2\"}*
merge	布林值	指定 AttributePayload提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 UpdateThing 加上一個空白屬性值。 Note merge 屬性僅於呼叫 UpdateThing 或 UpdateThingGroup 時有效。
expectedVersion	長整數	登錄檔內物件記錄的預期版本。若登錄檔內的記錄版本與請求指定的預期版本不符，則會使用 VersionConflictException 來拒絕 UpdateThing 請求。
removeThingType	布林值	移除物件類型關聯。若此值為 true，將移除關聯。

輸出

無

錯誤

InvalidRequestException

請求的內容無效。

VersionConflictException

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

ThrottlingException

請求頻率超過限制。

UnauthorizedException

未授權您執行此操作。

ServiceUnavailableException

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

UpdateThingGroup

更新物件群組。

概要

```
aws iot  update-thing-group \
    --thing-group-name <value> \
    --thing-group-properties <value> \
    [--expected-version <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingGroupName": "string",
  "thingGroupProperties": {
    "thingGroupDescription": "string",
    "attributePayload": {
      "attributes": {
        "string": "string"
      },
      "merge": "boolean"
    }
  },
  "expectedVersion": "long"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingGroupName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	欲更新的物件群組。
`thingGroupProperties`	ThingGroupProperties	物件群組屬性。
`thingGroupDescription`	string 長度上限：2028 模式：[\\p{Graph}]*	物件群組描述。
`attributePayload`	AttributePayload	JSON 格式的物件群組屬性。
`attributes`	對應	包含至多三個 JSON 格式的金鑰/值對之 JSON 字串。例如：

名稱	類型	描述
		`\"attributes\": {\"string1\": \"string2\"}`
`merge`	布林值	指定 `AttributePayload`提供的屬性清單是否與登錄檔內的屬性合併，而非覆寫。 若要移除一項屬性，請呼叫 `UpdateThing` 加上一個空白屬性值。 Note `merge` 屬性僅於呼叫 `UpdateThing` 或 `UpdateThingGroup` 時有效。
`expectedVersion`	長整數	物件群組的預期版本。若此值不符欲更新的物件群組版本，更新將會失敗。

輸出

```
{
  "version": "long"
}
```

CLI 輸出欄位

名稱	類型	描述
`version`	長整數	更新物件群組的版本。

錯誤

`InvalidRequestException`

請求的內容無效。

`VersionConflictException`

當傳送至命令的物件版本與版本參數所指定的不符，將擲出此例外狀況。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

UpdateThingGroupsForThing

更新某物件所屬的群組。

概要

```
aws iot  update-thing-groups-for-thing \
    [--thing-name <value>] \
    [--thing-groups-to-add <value>] \
    [--thing-groups-to-remove <value>] \
    [--override-dynamic-groups | --no-override-dynamic-groups]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "thingGroupsToAdd": [
    "string"
  ],
  "thingGroupsToRemove": [
    "string"
  ],
  "overrideDynamicGroups": "boolean"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	群組成員資格將更新的物件。
`thingGroupsToAdd`	列出 成員：ThingGroupName	將新增該物件的群組。
`thingGroupsToRemove`	列出 成員：ThingGroupName	將移除該物件的群組。
`overrideDynamicGroups`	布林值	達到 10 個群組的上限時，以靜態物件群組覆寫動態物件群組。如果物件屬於 10 個物件群組，而其中有一或多個群組為動態物件群組，則對靜態群組新增物件會從最後一個動態群組移除物件。

輸出

無

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。

`ResourceNotFoundException`

指定的資源不存在。

UpdateThingShadow

更新指定之物件的影子。

如需詳細資訊，請參閱《AWS IoT 開發人員指南》中的 UpdateThingShadow。

概要

```
aws iot-data  update-thing-shadow \
    --thing-name <value> \
    [--shadow-name <value>] \
    --payload <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "thingName": "string",
  "shadowName": "string",
  "payload": "blob"
}
```

`cli-input-json` 欄位

名稱	類型	描述
`thingName`	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	物件名稱。
`shadowName`	string 長度上限：64；下限：1 模式：[a-zA-Z0-9:_-]+	影子的名稱。
`payload`	blob	JSON 格式的狀態資訊。

輸出

```
{
```

```
  "payload": "blob"
}
```

CLI 輸出欄位

名稱	類型	描述
payload	blob	JSON 格式的狀態資訊。

錯誤

`ConflictException`

指定版本與文件版本不符。

`RequestEntityTooLargeException`

承載超過允許的最大值。

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`UnauthorizedException`

未授權您執行此操作。

`ServiceUnavailableException`

此服務暫時無法使用。

`InternalFailureException`

出現未預期的錯誤。

`MethodNotAllowedException`

不支援所指定的 HTTP 動詞和 URI 組合。

`UnsupportedDocumentEncodingException`

不支援該編碼。

ValidateSecurityProfileBehaviors

驗證 Device Defender 安全性描述檔行為規格。

概要

```
aws iot  validate-security-profile-behaviors \
    --behaviors <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json` 格式

```
{
  "behaviors": [
    {
      "name": "string",
      "metric": "string",
```

```
      "criteria": {
        "comparisonOperator": "string",
        "value": {
          "count": "long",
          "cidrs": [
            "string"
          ],
          "ports": [
            "integer"
          ]
        },
        "durationSeconds": "integer",
        "consecutiveDatapointsToAlarm": "integer",
        "consecutiveDatapointsToClear": "integer",
        "statisticalThreshold": {
          "statistic": "string"
        }
      }
    }
  ]
}
```

cli-input-json 欄位

名稱	類型	描述
behaviors	列出 成員：Behavior	指定當裝置 (實物) 違反時造成提醒的行為。
name	string 長度上限：128；下限：1 模式：[a-zA-Z0-9:_-]+	您已指定給行為的名稱。
metric	string	行為測量的內容。
criteria	BehaviorCriteria	可判斷裝置對於 metric 的操作是否正確的條件。
comparisonOperator	string	將測量的實物 (metric) 關聯到條件 (包含 value 或 statisticalThreshold) 的運算子。 列舉：less-than | less-than-equals | greater-than | greater-than-equals | in-cidr-set | not-in-cidr-set | in-port-set | not-in-port-set
value	MetricValue	相較於 metric 的值。
count	長整數 範圍 - 分鐘：0	如果 comparisonOperator 呼叫數值，請使用此項來指定相較於 metric 的數值。
cidrs	列出 成員：Cidr	如果 comparisonOperator 呼叫一組 CIDR，請使用此項來指定相較於 metric 的組合。

名稱	類型	描述
ports	列出 成員：Port	如果 comparisonOperator 呼叫一組連接埠，請使用此項來指定相較於 metric 的組合。
durationSeconds	integer	針對具有時間維度的這些條件，使用此項來指定評估行為的持續時間 (例如，NUM_MESSAGES_SENT)。對於 statisticalThreshhold 指標比較，系統會在此期間累積來自所有裝置的衡量值，在將這些值用來計算百分比；接著，系統也會在此期間收集來自個別裝置的量測值，在為這些值提供百分比排名。
consecutiveDatapointsToAlarm	integer 範圍 - 上限：10；下限：1	如果裝置違反特定數量連續資料點的行為，則會產生警示。如果未指定，則預設值為 1。
consecutiveDatapointsToClear	integer 範圍 - 上限：10；下限：1	如果發生警示，以及違例裝置不再違反指定數量的連續資料點行為，則會清除警示。如果未指定，則預設值為 1。
statisticalThreshold	StatisticalThreshold	表示閾值的統計排名 (百分比)，系統會使用該值來判斷某個行為是否符合或違反行為。
statistic	string 模式：(p0|p0.1|p0.01|p1|p10|p50|p90|p99|p99.9|p99.99|p100)	解析閾值的百分比，該閾值會用來判斷是否符合某個行為。系統會在指定期間 (durationSeconds) 從您帳戶中的所有報告裝置收集指標並計算統計排名。然後，系統會在相同期間收集來自裝置的衡量值。如果來自裝置累積的衡量值超過或低於與指定百分比關聯的值 (comparisonOperator)，則該裝置會被視為符合行為，否則就是違反行為。

輸出

```
{
  "valid": "boolean",
  "validationErrors": [
    {
      "errorMessage": "string"
    }
  ]
}
```

CLI 輸出欄位

名稱	類型	描述
`valid`	布林值	行為若有效則為 True。
`validationErrors`	清單 成員：ValidationError	行為中找到的任何錯誤清單。
`errorMessage`	字串 長度上限：2048	行為中找到對錯誤的描述。

錯誤

`InvalidRequestException`

請求的內容無效。

`ThrottlingException`

請求頻率超過限制。

`InternalFailureException`

出現未預期的錯誤。