API 参考

AWS Lambda



API 版本 2015-03-31

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

AWS Lambda: API 参考

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

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆或者贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

欢迎使用 1
在使用 SDK 时出现证书错误 1
操作 2
AddLayerVersionPermission 5
请求语法 5
URI 请求参数 5
请求体 6
响应语法 7
响应元素 7
错误 7
另请参阅 8
AddPermission 10
请求语法 10
URI 请求参数 10
请求正文 11
响应语法 13
响应元素 14
错误 14
另请参阅 15
CreateAlias 16
请求语法 16
URI 请求参数 16
请求体 17
响应语法 18
响应元素 18
错误 19
另请参阅 20
CreateCodeSigningConfig 21
请求语法 21
URI 请求参数 21
请求体 21
响应语法 22
响应元素 22
错误 23

另请参阅 23
CreateEventSourceMapping 24
请求语法 25
URI 请求参数 26
请求体 26
响应语法 32
响应元素 34
错误 39
另请参阅 40
CreateFunction 41
请求语法 41
URI 请求参数 43
请求体 43
响应语法 49
响应元素 51
错误 57
另请参阅 59
CreateFunctionUrlConfig 60
请求语法 60
URI 请求参数 60
请求正文 61
响应语法 62
响应元素 62
错误 63
另请参阅 64
DeleteAlias 66
请求语法 66
URI 请求参数 66
请求体 67
响应语法 67
响应元素 67
错误 67
另请参阅 67
DeleteCodeSigningConfig 69
请求语法 69
URI 请求参数 69

请求体 69
响应语法 69
响应元素 69
错误 69
另请参阅 70
DeleteEventSourceMapping 71
请求语法 71
URI 请求参数 71
请求体 71
响应语法 71
响应元素 73
错误 78
另请参阅 79
DeleteFunction 80
请求语法 80
URI 请求参数 80
请求正文 81
响应语法 81
响应元素 81
错误 81
另请参阅 82
DeleteFunctionCodeSigningConfig 83
请求语法 83
URI 请求参数 83
请求体 83
响应语法 83
响应元素 84
错误 84
另请参阅 84
DeleteFunctionConcurrency 86
请求语法 86
URI 请求参数 86
请求体 86
响应语法 86
响应元素 87
错误 87

另请参阅 87
DeleteFunctionEventInvokeConfig 89
请求语法 89
URI 请求参数 89
请求正文 90
响应语法 90
响应元素 90
错误 90
另请参阅 91
DeleteFunctionUrlConfig 92
请求语法 92
URI 请求参数 92
请求正文 93
响应语法 93
响应元素 93
错误 93
另请参阅 93
DeleteLayerVersion 95
请求语法 95
URI 请求参数 95
请求体 95
响应语法 95
响应元素 95
错误 96
另请参阅 96
DeleteProvisionedConcurrencyConfig 97
请求语法 97
URI 请求参数 97
请求体 98
响应语法 98
响应元素 98
错误 98
另请参阅 99
GetAccountSettings 100
请求语法 100
URI 请求参数 100

请求体 100
响应语法 100
响应元素 100
错误 101
另请参阅 101
GetAlias 103
请求语法 103
URI 请求参数 103
请求体 104
响应语法 104
响应元素 104
错误 105
另请参阅 106
GetCodeSigningConfig 107
请求语法 107
URI 请求参数 107
请求体 107
响应语法 107
响应元素 108
错误 108
另请参阅 108
GetEventSourceMapping 110
请求语法 110
URI 请求参数 110
请求体 110
响应语法 110
响应元素 112
错误 117
另请参阅 117
GetFunction 119
请求语法 119
URI 请求参数 119
请求正文 120
响应语法 120
响应元素 122
错误 123

另请参阅 124
GetFunctionCodeSigningConfig 125
请求语法 125
URI 请求参数 125
请求体 125
响应语法 125
响应元素 126
错误 127
另请参阅 127
GetFunctionConcurrency 129
请求语法 129
URI 请求参数 129
请求体 129
响应语法 129
响应元素 130
错误 130
另请参阅 131
GetFunctionConfiguration 132
请求语法 132
URI 请求参数 132
请求正文 133
响应语法 133
响应元素 135
错误 141
另请参阅 142
GetFunctionEventInvokeConfig 143
请求语法 143
URI 请求参数 143
请求正文 144
响应语法 144
响应元素 144
错误 145
另请参阅 146
GetFunctionUrlConfig 147
请求语法 147
URI 请求参数 147

请求正文 147
响应语法 148
响应元素 148
错误 149
另请参阅 150
GetLayerVersion 151
请求语法 151
URI 请求参数 151
请求体 151
响应语法 151
响应元素 152
错误 154
另请参阅 154
GetLayerVersionByArn 156
请求语法 156
URI 请求参数 156
请求体 156
响应语法 156
响应元素 157
错误 159
另请参阅 159
GetLayerVersionPolicy 161
请求语法 161
URI 请求参数 161
请求体 161
响应语法 161
响应元素 162
错误 162
另请参阅 163
GetPolicy 164
请求语法 164
URI 请求参数 164
请求正文 165
响应语法 165
响应元素 165
错误 165

另请参阅 166
GetProvisionedConcurrencyConfig 167
请求语法 167
URI 请求参数 167
请求体 168
响应语法 168
响应元素 168
错误 169
另请参阅 170
GetRuntimeManagementConfig 171
请求语法 171
URI 请求参数 171
请求正文 172
响应语法 172
响应元素 172
错误 173
另请参阅 173
Invoke 175
请求语法 175
URI 请求参数 176
请求正文 177
响应语法 177
响应元素 177
错误 178
另请参阅 182
InvokeAsync 183
请求语法 183
URI 请求参数 183
请求正文 184
响应语法 184
响应元素 184
错误 184
另请参阅 185
InvokeWithResponseStream 186
请求语法 186
URI 请求参数 186

请求正文 187
响应语法 187
响应元素 188
错误 189
另请参阅 193
ListAliases 194
请求语法 194
URI 请求参数 194
请求正文 195
响应语法 195
响应元素 195
错误 196
另请参阅 196
ListCodeSigningConfigs 198
请求语法 198
URI 请求参数 198
请求正文 198
响应语法 198
响应元素 199
错误 199
另请参阅 200
ListEventSourceMappings 201
请求语法 201
URI 请求参数 201
请求正文 202
响应语法 202
响应元素 204
错误 204
另请参阅 205
ListFunctionEventInvokeConfigs 206
请求语法 206
URI 请求参数 206
请求正文 207
响应语法 207
响应元素 207
错误 208

另请参阅 208
ListFunctions 210
请求语法 210
URI 请求参数 210
请求正文 211
响应语法 211
响应元素 213
错误 214
另请参阅 214
ListFunctionsByCodeSigningConfig 215
请求语法 215
URI 请求参数 215
请求正文 215
响应语法 215
响应元素 216
错误 216
另请参阅 217
ListFunctionUrlConfigs 218
请求语法 218
URI 请求参数 218
请求正文 219
响应语法 219
响应元素 219
错误 220
另请参阅 220
ListLayers 222
请求语法 222
URI 请求参数 222
请求正文 223
响应语法 223
响应元素 223
错误 224
另请参阅 224
ListLayerVersions 226
请求语法 226
URI 请求参数 226

请求正文 227
响应语法 227
响应元素 227
错误 228
另请参阅 228
ListProvisionedConcurrencyConfigs 230
请求语法 230
URI 请求参数 230
请求正文 231
响应语法 231
响应元素 231
错误 232
另请参阅 232
ListTags 234
请求语法 234
URI 请求参数 234
请求体 234
响应语法 234
响应元素 234
错误 235
另请参阅 235
ListVersionsByFunction 237
请求语法 237
URI 请求参数 237
请求正文 238
响应语法 238
响应元素 240
错误 240
另请参阅 241
PublishLayerVersion 242
请求语法 242
URI 请求参数 242
请求体 243
响应语法 244
响应元素 245
错误 247

另请参阅 247
PublishVersion 249
请求语法 249
URI 请求参数 249
请求体 250
响应语法 250
响应元素 253
错误 259
另请参阅 260
PutFunctionCodeSigningConfig 261
请求语法 261
URI 请求参数 261
请求体 261
响应语法 262
响应元素 262
错误 263
另请参阅 264
PutFunctionConcurrency 265
请求语法 265
URI 请求参数 265
请求体 266
响应语法 266
响应元素 266
错误 266
另请参阅 267
PutFunctionEventInvokeConfig 269
请求语法 269
URI 请求参数 269
请求正文 270
响应语法 271
响应元素 271
错误 273
另请参阅 273
PutProvisionedConcurrencyConfig 275
请求语法 275
URI 请求参数 275

请求体 276
响应语法 276
响应元素 276
错误 277
另请参阅 278
PutRuntimeManagementConfig 279
请求语法 279
URI 请求参数 279
请求正文 280
响应语法 281
响应元素 281
错误 282
另请参阅 282
RemoveLayerVersionPermission 284
请求语法 284
URI 请求参数 284
请求体 285
响应语法 285
响应元素 285
错误 285
另请参阅 286
RemovePermission 287
请求语法 287
URI 请求参数 287
请求体 288
响应语法 288
响应元素 288
错误 288
另请参阅 289
TagResource 290
请求语法 290
URI 请求参数 290
请求体 290
响应语法 291
响应元素 291
错误 291

另请参阅 291
UntagResource 293
请求语法 293
URI 请求参数 293
请求体 293
响应语法 293
响应元素 293
错误 294
另请参阅 294
UpdateAlias 296
请求语法 296
URI 请求参数 296
请求体 297
响应语法 298
响应元素 298
错误 299
另请参阅 300
UpdateCodeSigningConfig 302
请求语法 302
URI 请求参数 302
请求体 302
响应语法 303
响应元素 304
错误 304
另请参阅 304
UpdateEventSourceMapping 306
请求语法 307
URI 请求参数 308
请求体 308
响应语法 312
响应元素 313
错误 319
另请参阅 319
UpdateFunctionCode 321
请求语法 321
URI 请求参数 321

请求体 322
响应语法 324
响应元素 326
错误 333
另请参阅 334
UpdateFunctionConfiguration 335
请求语法 335
URI 请求参数 336
请求体 337
响应语法 341
响应元素 343
错误 349
另请参阅 351
UpdateFunctionEventInvokeConfig 352
请求语法 352
URI 请求参数 352
请求正文 353
响应语法 354
响应元素 354
错误 355
另请参阅 356
UpdateFunctionUrlConfig 357
请求语法 357
URI 请求参数 357
请求正文 358
响应语法 359
响应元素 359
错误 361
另请参阅 361
数据类型 363
AccountLimit 366
内容 366
另请参阅 367
AccountUsage 368
内容 368
另请参阅 368

AliasConfiguration 369
内容 369
另请参阅 370
AliasRoutingConfiguration 371
内容 371
另请参阅 371
AllowedPublishers 372
内容 372
另请参阅 372
AmazonManagedKafkaEventSourceConfig 373
内容 373
另请参阅 373
CodeSigningConfig 374
内容 374
另请参阅 375
CodeSigningPolicies 376
内容 376
另请参阅 376
Concurrency 377
内容 377
另请参阅 377
Cors 378
内容 378
另请参阅 380
DeadLetterConfig 381
内容 381
另请参阅 381
DestinationConfig 382
内容 382
另请参阅 382
DocumentDBEventSourceConfig 383
内容 383
另请参阅 383
Environment 385
内容 385
另请参阅 385

EnvironmentError 386
内容 386
另请参阅 386
EnvironmentResponse 387
内容 387
另请参阅 387
EphemeralStorage 388
内容 388
另请参阅 388
EventSourceMappingConfiguration 389
内容 389
另请参阅 395
FileSystemConfig 396
内容 396
另请参阅 396
Filter 397
内容 397
另请参阅 397
FilterCriteria 398
内容 398
另请参阅 398
FunctionCode 399
内容 399
另请参阅 400
FunctionCodeLocation 401
内容 401
另请参阅 401
FunctionConfiguration 403
内容 403
另请参阅 410
FunctionEventInvokeConfig 412
内容 412
另请参阅 413
FunctionUrlConfig 414
内容 414
另请参阅 415

ImageConfig 417
内容 417
另请参阅 417
ImageConfigError 419
内容 419
另请参阅 419
ImageConfigResponse 420
内容 420
另请参阅 420
InvokeResponseStreamUpdate 421
内容 421
另请参阅 421
InvokeWithResponseStreamCompleteEvent 422
内容 422
另请参阅 422
InvokeWithResponseStreamResponseEvent 423
内容 423
另请参阅 423
Layer 424
内容 424
另请参阅 425
LayersListItem 426
内容 426
另请参阅 426
LayerVersionContentInput 428
内容 428
另请参阅 429
LayerVersionContentOutput 430
内容 430
另请参阅 431
LayerVersionsListItem 432
内容 432
另请参阅 433
LoggingConfig 435
内容 435
另请参阅 436

OnFailure 437
内容 437
另请参阅 437
OnSuccess 438
内容 438
另请参阅 438
ProvisionedConcurrencyConfigListItem 439
内容 439
另请参阅 440
RuntimeVersionConfig 441
内容 441
另请参阅 441
RuntimeVersionError 442
内容 442
另请参阅 442
ScalingConfig 443
内容 443
另请参阅 443
SelfManagedEventSource 444
内容 444
另请参阅 444
SelfManagedKafkaEventSourceConfig 445
内容 445
另请参阅 445
SnapStart 446
内容 446
另请参阅 446
SnapStartResponse 447
内容 447
另请参阅 447
SourceAccessConfiguration 448
内容 448
另请参阅 449
TracingConfig 450
内容 450
另请参阅 450

TracingConfigResponse 451
内容 451
另请参阅 451
VpcConfig 452
内容 452
另请参阅 452
VpcConfigResponse 454
内容 454
另请参阅 454
常见参数 456
常见错误 459

欢迎使用

本节包含 AWS Lambda API 参考文档。我们建议使用您所使用编程语言的其中一款 AWS 软件开发工具包（SDK），而不是直接从应用程序请求使用 API。AWS SDK 负责请求身份验证、序列化和连接管理。如果不使用 AWS SDK，则需要通过提供签名对请求进行身份验证。AWS Lambda 支持签名版本 4。有关更多信息，请参阅《Amazon Web Services 一般参考》中的签名版本 4 签名流程。

在使用 SDK 时出现证书错误

由于AWS开发工具包使用的是来自计算机的 CA 证书，因此更改AWS服务器上的证书可能会导致您在尝试使用开发工具包时无法连接。您可以通过使计算机的 CA 证书和操作系统保持最新来防止出现这些故障。如果您在公司环境中遇到这个问题而且未管理您自己的计算机，则可能需要请求管理员来协助处理更新过程。以下列表显示了最低的操作系统和 Java 版本：

- 已安装 2005 年 1 月版或更高版本更新的 Microsoft Windows 版本在其信任列表中至少包含一个必需 CA。
- 带 Java for Mac OS X 10.4 版本 5 的 Mac OS X 10.4 (2007 年 2 月版)、Mac OS X 10.5 (2007 年 10 月版) 及更高版本在其信任列表中至少包含一个必需 CA。
- Red Hat Enterprise Linux 5 (2007 年 3 月版)、6 和 7 以及 CentOS 5、6 和 7 在其默认信任 CA 列表中至少包含一个必需 CA。
- Java 1.4.2_12 (2006 年 5 月版)、5 Update 2 (2005 年 3 月版) 以及所有更高版本，包括 Java 6 (2006 年 12 月版)、7 和 8 在其默认信任 CA 列表中至少包含一个必需 CA。

在访问 AWS Lambda 管理控制台或 AWS Lambda API 端点时，无论是通过浏览器还是以编程方式，您都需要确保您的客户端计算机支持任何以下 CA：

- Amazon Root CA 1
- Starfield Services Root Certificate Authority – G2
- Starfield Class 2 Certification Authority

可以从 Amazon Trust Services 获得来自前两个颁发机构的根证书，而使您的计算机保持最新是更直接的解决方案。要了解 ACM 提供的证书的更多信息，请参阅AWS Certificate Manager 常见问题。

操作

支持以下操作：

- AddLayerVersionPermission
- AddPermission
- CreateAlias
- CreateCodeSigningConfig
- CreateEventSourceMapping
- CreateFunction
- CreateFunctionUrlConfig
- DeleteAlias
- DeleteCodeSigningConfig
- DeleteEventSourceMapping
- DeleteFunction
- DeleteFunctionCodeSigningConfig
- DeleteFunctionConcurrency
- DeleteFunctionEventInvokeConfig
- DeleteFunctionUrlConfig
- DeleteLayerVersion
- DeleteProvisionedConcurrencyConfig
- GetAccountSettings
- GetAlias
- GetCodeSigningConfig
- GetEventSourceMapping
- GetFunction
- GetFunctionCodeSigningConfig
- GetFunctionConcurrency
- GetFunctionConfiguration
- GetFunctionEventInvokeConfig
- GetFunctionUrlConfig

- GetLayerVersion
- GetLayerVersionByArn
- GetLayerVersionPolicy
- GetPolicy
- GetProvisionedConcurrencyConfig
- GetRuntimeManagementConfig
- Invoke
- InvokeAsync
- InvokeWithResponseStream
- ListAliases
- ListCodeSigningConfigs
- ListEventSourceMappings
- ListFunctionEventInvokeConfigs
- ListFunctions
- ListFunctionsByCodeSigningConfig
- ListFunctionUrlConfigs
- ListLayers
- ListLayerVersions
- ListProvisionedConcurrencyConfigs
- ListTags
- ListVersionsByFunction
- PublishLayerVersion
- PublishVersion
- PutFunctionCodeSigningConfig
- PutFunctionConcurrency
- PutFunctionEventInvokeConfig
- PutProvisionedConcurrencyConfig
- PutRuntimeManagementConfig
- RemoveLayerVersionPermission
- RemovePermission

- TagResource
- UntagResource
- UpdateAlias
- UpdateCodeSigningConfig
- UpdateEventSourceMapping
- UpdateFunctionCode
- UpdateFunctionConfiguration
- UpdateFunctionEventInvokeConfig
- UpdateFunctionUrlConfig

AddLayerVersionPermission

为某个 AWS Lambda 层的某个版本添加基于资源的策略权限。使用此操作可将层使用权限授予其他账户。您可以向企业中的单个账户、所有账户或所有AWS账户授予权限。

要撤销权限，请调用 RemoveLayerVersionPermission 与您在添加时指定的语句 ID。

请求语法

```
POST /2018-10-31/layers/LayerName/versions/VersionNumber/policy?RevisionId=RevisionId
 HTTP/1.1
Content-type: application/json

{
   "Action": "string",
   "OrganizationId": "string",
   "Principal": "string",
   "StatementId": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：是

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新策略。使用此选项以避免修改自上次读取后已发生更改的策略。

VersionNumber

版本号。

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

Action

授予层访问权限的 API 操作。例如，`lambda:GetLayerVersion`。

类型：字符串

长度约束：最大长度为 22。

模式：`lambda:GetLayerVersion`

必需：是

OrganizationId

如果将主体设置为 `*`，则为指定组织中的所有账户授予权限。

类型：字符串

长度约束：最大长度为 34。

模式：`o-[a-z0-9]{10,32}`

必需：否

Principal

账户 ID 或 `*` 为企业中的所有账户或所有AWS账户（如果 `organizationId` 未指定）授予层级使用权限。对于后一种情况，请确保您确实希望所有AWS账户都具有此层级的使用权限。

类型：字符串

模式：`\d{12}|\*|arn:(aws[a-zA-Z-]*):iam::\d{12}:root`

必需：是

StatementId

将策略与同一层版本上的其他策略区分开来的标识符。

类型：字符串

长度约束：最小长度为 1。最大长度为 100。

模式：([a-zA-Z0-9-_]+)

必需：是

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "RevisionId": "string",
   "Statement": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

RevisionId

策略当前修订版的唯一标识符。

类型：字符串

Statement

权限语句。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PolicyLengthExceededException

资源的权限策略太大。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AddPermission

授予 AWS 服务、AWS 账户 或 AWS 组织使用函数的权限。您可以在函数级别应用策略，也可以指定限定符以限制对单个版本或别名的访问。如果您使用限定符，则调用程序必须使用该版本或别名的完整 Amazon Resource Name (ARN) 来调用该函数。注意：Lambda 不支持将策略添加到版本 $LATEST。

要向另一个账户授予权限，请将账户 ID 指定为 `Principal`。要向在 AWS Organizations 中定义的组织授予权限，请将组织 ID 指定为 `PrincipalOrgID`。对于 AWS 服务，主体是服务定义的域样式标识符，例如 `s3.amazonaws.com` 或 `sns.amazonaws.com`。对于 AWS 服务，您还可以将关联资源的 ARN 指定为 `SourceArn`。如果在未指定源的情况下向服务主体授予权限，则其他账户可能会在其账户中配置资源以调用 Lambda 函数。

此操作将一条语句添加到该函数的基于资源的权限策略。有关函数策略的更多信息，请参阅将基于资源的策略用于 Lambda。

请求语法

```
POST /2015-03-31/functions/FunctionName/policy?Qualifier=Qualifier HTTP/1.1
Content-type: application/json

{
   "Action": "string",
   "EventSourceToken": "string",
   "FunctionUrlAuthType": "string",
   "Principal": "string",
   "PrincipalOrgID": "string",
   "RevisionId": "string",
   "SourceAccount": "string",
   "SourceArn": "string",
   "StatementId": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

指定版本或别名以向函数的发布版本添加权限。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

请求接受采用 JSON 格式的以下数据。

Action

主体可以对函数执行的操作。例如，`lambda:InvokeFunction` 或 `lambda:GetFunction`。

类型：字符串

模式：`(lambda:[*]|lambda:[a-zA-Z]+|[*])`

必需：是

EventSourceToken

对于 Alexa Smart Home 函数，为必须由调用程序提供的令牌。

类型：字符串

长度约束：最小长度为 0。最大长度为 256。

模式：`[a-zA-Z0-9._\-]+`

必需：否

FunctionUrlAuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 `AWS_IAM`。如果想绕过 IAM 身份验证以创建公有端点，请设置为 `NONE`。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：`NONE | AWS_IAM`

必需：否

Principal

调用函数的 AWS 服务 或 AWS 账户。如果您指定一个服务，请使用 `SourceArn` 或 `SourceAccount` 来限制哪些人可以通过该服务调用该函数。

类型：字符串

模式：`[^\s]+`

必需：是

PrincipalOrgID

AWS Organizations 中的组织标识符。使用此函数向此组织下的所有 AWS 账户 授予权限。

类型：字符串

长度限制：最小长度为 12。最大长度为 34。

模式：`^o-[a-z0-9]{10,32}$`

必需：否

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新策略。使用此选项以避免修改自上次读取后已发生更改的策略。

类型：字符串

必需：否

SourceAccount

对于 AWS 服务，为拥有此资源的 AWS 账户 ID。将其与 SourceArn 一起使用以确保资源归指定的账户所有。Amazon S3 存储桶的拥有者可以将其删除，并由其他账户重新创建。

类型：字符串

长度约束：最大长度为 12。

模式：\d{12}

必需：否

SourceArn

对于 AWS 服务，为调用该函数的 AWS 资源的 ARN。例如，Amazon S3 存储桶或 Amazon SNS 主题。

请注意，Lambda 使用 StringLike 运算符配置该比较。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

必需：否

StatementId

将语句与同一策略中的其他语句区分开来的语句标识符。

类型：字符串

长度约束：最小长度为 1。最大长度为 100。

模式：([a-zA-Z0-9-_]+)

必需：是

响应语法

```
HTTP/1.1 201
```

```
Content-type: application/json

{
   "Statement": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

Statement

添加到函数策略的权限语句。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PolicyLengthExceededException

资源的权限策略太大。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CreateAlias

为 Lambda 函数版本创建一个别名。别名用来为客户端提供函数标识符，您可以更新该标识符以调用其他版本。

您还可以映射别名以在两个版本之间拆分调用请求。使用 RoutingConfig 参数可指定另一个版本以及该版本收到的调用请求的百分比。

请求语法

```
POST /2015-03-31/functions/FunctionName/aliases HTTP/1.1
Content-type: application/json

{
   "Description": "string",
   "FunctionVersion": "string",
   "Name": "string",
   "RoutingConfig": {
      "AdditionalVersionWeights": {
         "string" : number
      }
   }
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - MyFunction。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:MyFunction。
- 部分 ARN - 123456789012:function:MyFunction。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

必需：是

Name

别名的名称。

类型：字符串

长度限制：长度下限为 1。长度上限为 128。

模式：`(?!^[0-9]+$)([a-zA-Z0-9-_]+)`

必需：是

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "AliasArn": "string",
   "Description": "string",
   "FunctionVersion": "string",
   "Name": "string",
   "RevisionId": "string",
   "RoutingConfig": {
      "AdditionalVersionWeights": {
         "string" : number
      }
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

AliasArn

是别名的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

Name

别名的名称。

类型：字符串

长度限制：长度下限为 1。长度上限为 128。

模式：(?!^[0-9]+$)([a-zA-Z0-9-_]+)

RevisionId

更新别名时更改的唯一标识符。

类型：字符串

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CreateCodeSigningConfig

创建代码签名配置。代码签名配置定义了允许的签名配置文件列表以及代码签名验证策略（在部署验证检查失败时要采取的操作）。

请求语法

```
POST /2020-04-22/code-signing-configs/ HTTP/1.1
Content-type: application/json

{
   "AllowedPublishers": {
      "SigningProfileVersionArns": [ "string" ]
   },
   "CodeSigningPolicies": {
      "UntrustedArtifactOnDeployment": "string"
   },
   "Description": "string"
}
```

URI 请求参数

该请求不使用任何 URI 参数。

请求体

请求接受采用 JSON 格式的以下数据。

AllowedPublishers

此代码签名配置的签名配置文件。

类型：AllowedPublishers 对象

必需：是

CodeSigningPolicies

代码签名策略定义验证检查失败时要执行的操作。

类型：CodeSigningPolicies 对象

必需：否

Description

此代码签名配置的描述性名称。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "CodeSigningConfig": {
      "AllowedPublishers": {
         "SigningProfileVersionArns": [ "string" ]
      },
      "CodeSigningConfigArn": "string",
      "CodeSigningConfigId": "string",
      "CodeSigningPolicies": {
         "UntrustedArtifactOnDeployment": "string"
      },
      "Description": "string",
      "LastModified": "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

CodeSigningConfig

代码签名配置。

类型：CodeSigningConfig 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CreateEventSourceMapping

在事件源和 AWS Lambda 函数之间创建映射。Lambda 从事件源读取项目并调用该函数。

有关如何配置不同事件源的详细信息，请参阅以下主题。

- Amazon DynamoDB Streams
- Amazon Kinesis
- Amazon SQS
- Amazon MQ 和 RabbitMQ
- Amazon MSK
- Apache Kafka
- Amazon DocumentDB

以下错误处理选项仅适用于流源（DynamoDB 和 Kinesis）：

- `BisectBatchOnFunctionError` – 如果函数返回错误，则将批次拆分为两批并重试。
- `DestinationConfig` – 将丢弃的记录发送到 Amazon SQS 队列或 Amazon SNS 主题。
- `MaximumRecordAgeInSeconds` – 丢弃超过指定期限的记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。
- `MaximumRetryAttempts` – 在指定重试次数后丢弃记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。
- `ParallelizationFactor` – 同时处理来自每个分片的多个批次。

有关适用于每个事件源的配置参数信息，请参阅以下主题。

- Amazon DynamoDB Streams
- Amazon Kinesis
- Amazon SQS
- Amazon MQ 和 RabbitMQ
- Amazon MSK
- Apache Kafka
- Amazon DocumentDB

请求语法

```
POST /2015-03-31/event-source-mappings/ HTTP/1.1
Content-type: application/json

{
   "AmazonManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "Enabled": boolean,
   "EventSourceArn": "string",
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
         }
      ]
   },
   "FunctionName": "string",
   "FunctionResponseTypes": [ "string" ],
   "MaximumBatchingWindowInSeconds": number,
   "MaximumRecordAgeInSeconds": number,
   "MaximumRetryAttempts": number,
   "ParallelizationFactor": number,
   "Queues": [ "string" ],
   "ScalingConfig": {
      "MaximumConcurrency": number
   },
```

```
   "SelfManagedEventSource": {
      "Endpoints": {
         "string" : [ "string" ]
      }
   },
   "SelfManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "SourceAccessConfigurations": [
      {
         "Type": "string",
         "URI": "string"
      }
   ],
   "StartingPosition": "string",
   "StartingPositionTimestamp": number,
   "Topics": [ "string" ],
   "TumblingWindowInSeconds": number
}
```

URI 请求参数

该请求不使用任何 URI 参数。

请求体

请求接受采用 JSON 格式的以下数据。

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

必需：否

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

- Amazon Kinesis – 默认值为 100。最大值为 10,000。
- Amazon DynamoDB Streams – 默认值为 100。最大值为 10,000。

- Amazon Simple Queue Service – 默认值为 10。对于标准队列，最大值为 10,000。对于 FIFO 队列，最大值为 10。
- Amazon Managed Streaming for Apache Kafka – 默认值为 100。最大值为 10,000。
- 自行管理的 Apache Kafka – 默认值为 100。最大值为 10,000。
- Amazon MQ（ActiveMQ 和 RabbitMQ）– 默认值为 100。最大值为 10,000。
- DocumentDB – 默认值 100。最大值为 10,000。

类型：整数

有效范围：最小值为 1。最大值为 10000。

必需：否

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。

类型：布尔值

必需：否

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Kafka）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

必需：否

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

必需：否

Enabled

当为 true 时，事件源映射处于活动状态。当为 false 时，Lambda 暂停轮询和调用。

默认值：True

类型：布尔值

必需：否

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

- Amazon Kinesis – 数据流或流使用者的 ARN。
- Amazon DynamoDB Streams – 流的 ARN。
- Amazon Simple Queue Service – 队列的 ARN。
- Amazon Managed Streaming for Apache Kafka – 集群的 ARN 或 VPC 连接的 ARN（用于跨账户事件源映射）。
- Amazon MQ – 代理的 ARN。
- Amazon DocumentDB – DocumentDB 更改流的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

必需：否

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `MyFunction`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 版本或别名 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction:PROD`。
- 部分 ARN – `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度将限制为 64 个字符。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

必需：否

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

必需：否

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。默认值为无限 (-1)。

类型：整数

有效范围：最小值为 -1。最大值为 604800。

必需：否

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

必需：否

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。

类型：整数

有效范围：最小值为 1。最大值为 10。

必需：否

Queues

(MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：`[\s\S]*`

必需：否

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

必需：否

SelfManagedEventSource

从中接收记录的自行管理的 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

必需：否

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

必需：否

SourceAccessConfigurations

用于保护事件源的身份验证协议数组或 VPC 组件。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

必需：否

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

必需：否

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

必需：否

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

必需：否

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

必需：否

响应语法

```
HTTP/1.1 202
Content-type: application/json

{
   "AmazonManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
```

```
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "EventSourceArn": "string",
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
         }
      ]
   },
   "FunctionArn": "string",
   "FunctionResponseTypes": [ "string" ],
   "LastModified": number,
   "LastProcessingResult": "string",
   "MaximumBatchingWindowInSeconds": number,
   "MaximumRecordAgeInSeconds": number,
   "MaximumRetryAttempts": number,
   "ParallelizationFactor": number,
   "Queues": [ "string" ],
   "ScalingConfig": {
      "MaximumConcurrency": number
   },
   "SelfManagedEventSource": {
      "Endpoints": {
         "string" : [ "string" ]
      }
   },
   "SelfManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "SourceAccessConfigurations": [
      {
         "Type": "string",
         "URI": "string"
      }
   ],
   "StartingPosition": "string",
   "StartingPositionTimestamp": number,
   "State": "string",
   "StateTransitionReason": "string",
   "Topics": [ "string" ],
```

```
    "TumblingWindowInSeconds": number,
    "UUID": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 202 响应。

服务以 JSON 格式返回以下数据。

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

默认值：因服务而异。对于 Amazon SQS，默认值为 10。对于所有其他服务，默认值为 100。

相关设置：当将 `BatchSize` 设置为一个大于 10 的值时，必须至少将 `MaximumBatchingWindowInSeconds` 设置为 1。

类型：整数

有效范围：最小值为 1。最大值为 10000。

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。默认值为 False。

类型：布尔值

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Apache Kafka 事件源）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

FunctionArn

Lambda 函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

LastModified

上次更新事件源映射或其状态更改的日期（以 Unix 时间秒为单位）。

类型：时间戳

LastProcessingResult

函数的最后一次 Lambda 调用结果。

类型：字符串

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。原定设置值为 -1，该值将最大年龄设置为无限。将该值设置为无限后，Lambda 永远不会删除旧记录。

> **Note**
>
> 最长记录期限的最小有效值为 60 秒。尽管小于 60 和大于 -1 的值都在参数的绝对范围内，但不允许使用它们

类型：整数

有效范围：最小值为 -1。最大值为 604800。

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。原定设置值为 -1，该值将最大重试次数设置为无限。将 MaximumRetryAttempts 设置为无限后，Lambda 会重试失败的记录，直到事件源中的记录过期为止。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。默认值是 1。

类型：整数

有效范围：最小值为 1。最大值为 10。

Queues

(Amazon MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：`[\s\S]*`

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

SourceAccessConfigurations

用于保护与定义事件源的身份验证协议数组 VPC 组件或虚拟化主机。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

State

事件源映射的状态。此状态可以是以下值之一：Creating、Enabling、Enabled、Disabling、Disabled、Updating 或 Deleting。

类型：字符串

StateTransitionReason

表明是用户还是 Lambda 对事件源映射进行了最后一次更改。

类型：字符串

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

UUID

事件源映射的标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CreateFunction

创建 Lambda 函数。要创建函数，需要部署程序包和执行角色。部署包是包含函数代码的 .zip 文件存档或容器映像。执行角色授予该函数使用 AWS 服务 的权限，例如用于日志流的 Amazon CloudWatch Logs 和用于请求跟踪的 AWS X-Ray。

如果部署包是容器映像，则可将部署包类型设置为 `Image`。对于容器映像，代码属性必须在 Amazon ECR 注册表中包含容器映像的 URI。您无需指定处理程序和运行时属性。

如果部署包是 .zip 文件存档，则可将部署包类型设置为 `Zip`。对于 .zip 文件存档，代码属性会指定 .zip 文件的位置。您还必须指定处理程序和运行时属性。部署包中的代码必须与函数的目标指令集架构兼容（`x86-64` 或 `arm64`）。如果未指定架构，则默认值为 `x86-64`。

创建函数时，Lambda 会预置该函数及其支持资源的实例。如果您的函数连接到 VPC，此过程可能需要一分钟左右。在此期间，您无法调用或修改函数。来自 GetFunctionConfiguration 的响应中的 `State`、`StateReason` 和 `StateReasonCode` 字段指示函数何时可进行调用。有关更多信息，请参阅 Lambda 函数状态。

函数具有未发布的版本，并且可以具有已发布的版本和别名。更新函数的代码和配置时，未发布的版本将会更改。已发布版本是函数代码和配置的快照，无法更改。别名是映射到版本的命名资源，并且可以更改为映射到其他版本。使用 `Publish` 参数，以通过其初始配置创建函数的版本 1。

其他参数允许您配置特定于版本和函数级别的设置。稍后可以使用 UpdateFunctionConfiguration 修改特定于版本的设置。函数级别设置同时应用于函数的未发布版本和已发布版本，并且包含标签 (TagResource) 和每函数并发限制 (PutFunctionConcurrency)。

如果部署包是 .zip 文件存档，则可以使用 代码签名 。要为此函数启用代码签名，请指定代码签名配置的 ARN。当用户尝试使用 UpdateFunctionCode部署代码包时，Lambda 会检查代码包是否具有可信发布者的有效签名。代码签名配置包括一组签名配置文件，这些配置文件定义了此函数的可信发布者。

如果另一个 AWS 账户 或 AWS 服务 调用您的函数，请使用 AddPermission 通过创建基于资源的 AWS Identity and Access Management（IAM）policy 授予权限。您可以在函数级别授予版本或别名的权限。

要直接调用函数，请使用 Invoke。要调用您的函数以响应其他 AWS 服务，请创建事件源映射（CreateEventSourceMapping），或者在其他服务中配置函数触发器。有关更多信息，请参阅调用 Lambda 函数。

请求语法

```
POST /2015-03-31/functions HTTP/1.1
```

```
Content-type: application/json

{
   "Architectures": [ "string" ],
   "Code": {
      "ImageUri": "string",
      "S3Bucket": "string",
      "S3Key": "string",
      "S3ObjectVersion": "string",
      "ZipFile": blob
   },
   "CodeSigningConfigArn": "string",
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfig": {
      "Command": [ "string" ],
      "EntryPoint": [ "string" ],
      "WorkingDirectory": "string"
   },
   "KMSKeyArn": "string",
   "Layers": [ "string" ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
```

```
    },
    "MemorySize": number,
    "PackageType": "string",
    "Publish": boolean,
    "Role": "string",
    "Runtime": "string",
    "SnapStart": {
       "ApplyOn": "string"
    },
    "Tags": {
       "string" : "string"
    },
    "Timeout": number,
    "TracingConfig": {
       "Mode": "string"
    },
    "VpcConfig": {
       "Ipv6AllowedForDualStack": boolean,
       "SecurityGroupIds": [ "string" ],
       "SubnetIds": [ "string" ]
    }
}
```

URI 请求参数

该请求不使用任何 URI 参数。

请求体

请求接受采用 JSON 格式的以下数据。

Architectures

该函数支持的指令集架构。输入具有其中一个有效值（arm64 或 x86_64）的字符串数组。默认值为 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：x86_64 | arm64

必需：否

Code

函数的代码。

类型：FunctionCode 对象

必需：是

CodeSigningConfigArn

要为此函数启用代码签名，请指定代码签名配置的 ARN。代码签名配置包括一组签名配置文件，这些配置文件定义了此函数的可信发布者。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：否

DeadLetterConfig

死信队列配置，用于指定 Lambda 在处理失败时发送异步事件的队列或主题。有关更多信息，请参阅死信队列。

类型：DeadLetterConfig 对象

必需：否

Description

该函数的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

Environment

执行期间可从函数代码访问的环境变量。

类型：Environment 对象

必需：否

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

必需：否

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

必需：否

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – my-function。
- 函数 ARN – arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN – 123456789012:function:my-function。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Handler

代码中方法的名称，Lambda 可调用该方法来运行您的函数。如果部署包是 .zip 文件归档，则需要使用处理程序。该格式包括文件名，也可以包括命名空间和其他限定符，具体取决于运行时。有关更多信息，请参阅 Lambda 编程模型。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

必需：否

ImageConfig

覆盖容器镜像 Dockerfile 中的值的容器镜像配置值。

类型：ImageConfig 对象

必需：否

KMSKeyArn

AWS Key Management Service (AWS KMS) 客户托管密钥的 ARN，用于加密函数的环境变量。激活 Lambda SnapStart 后，Lambda 还使用此密钥对函数的快照进行加密。如果您使用容器映像部署函数，Lambda 还会在部署函数后使用此密钥对其进行加密。请注意，此密钥与在 Amazon Elastic Container Registry（Amazon ECR）中用于保护容器映像的密钥不同。如果您未提供客户托管密钥，Lambda 将使用默认服务密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

必需：否

Layers

要添加到函数执行环境中的函数层列表。通过 ARN 指定每一层，包括版本。

类型：字符串数组

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

必需：否

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

必需：否

MemorySize

在运行时 可用于函数的内存 量。提高该函数的内存也会增加它的 CPU 分配。默认值为 128 MB。该值可以是 1 MB 的任意倍数。

类型：整数

有效范围：最小值为 128。最大值为 10240。

必需：否

PackageType

部署程序包的类型。将容器映像设置为 Image，然后将 .zip 文件归档设置为 Zip。

类型：字符串

有效值：Zip | Image

必需：否

Publish

设置为 true 以在创建期间发布函数的第一个版本。

类型：布尔值

必需：否

Role

函数的执行角色的 Amazon Resource Name (ARN)。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

必需：是

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

必需：否

SnapStart

函数的 SnapStart 设置。

类型：SnapStart 对象

必需：否

Tags

要应用于函数的标签列表。

类型：字符串到字符串映射

必需：否

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。默认值为 3 秒。允许的最大值为 900 秒。有关更多信息，请参阅 Lambda 执行环境。

类型：整数

有效范围：最小值为 1。

必需：否

TracingConfig

将 Mode 设置为 Active，以使用 X-Ray 对传入请求的子集进行采样和跟踪。

类型：TracingConfig 对象

必需：否

VpcConfig

对于与 VPC 中的AWS资源的网络连接，指定 VPC 中的安全组和子网列表。当您将函数连接到 VPC 时，它只能通过该 VPC 访问资源和互联网。有关更多信息，请参阅配置 Lambda 函数以访问 VPC 中的资源。

类型：VpcConfig 对象

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "Architectures": [ "string" ],
   "CodeSha256": "string",
   "CodeSize": number,
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionArn": "string",
```

```
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "RuntimeVersionArn": "string"
```

```
    },
    "SigningJobArn": "string",
    "SigningProfileVersionArn": "string",
    "SnapStart": {
       "ApplyOn": "string",
       "OptimizationStatus": "string"
    },
    "State": "string",
    "StateReason": "string",
    "StateReasonCode": "string",
    "Timeout": number,
    "TracingConfig": {
       "Mode": "string"
    },
    "Version": "string",
    "VpcConfig": {
       "Ipv6AllowedForDualStack": boolean,
       "SecurityGroupIds": [ "string" ],
       "SubnetIds": [ "string" ],
       "VpcId": "string"
    }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：x86_64 | arm64

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 `Successful`。

类型：字符串

有效值：Successful | Failed | InProgress

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Layers

函数的层。

类型：Layer 对象数组

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

PackageType

部署程序包的类型。将容器映像设置为 Image，然后将 .zip 文件存档设置为 Zip。

类型：字符串

有效值：Zip | Image

RevisionId

函数或别名的最新更新版本。

类型：字符串

Role

函数的执行角色。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

RuntimeVersionConfig

运行时的 ARN 和发生的任何错误。

类型：RuntimeVersionConfig 对象

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SnapStart

发布函数版本时，将 `ApplyOn` 设置为 `PublishedVersions` 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

State

函数的当前状态。当状态为 `Inactive`，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：`Pending | Active | Inactive | Failed`

StateReason

函数当前状态的原因。

类型：字符串

StateReasonCode

函数当前状态的原因代码。当代码为 `Creating` 时，无法调用或修改该函数。

类型：字符串

有效值：`Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError`

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeSigningConfigNotFoundException

指定的代码签名配置不存在。

HTTP 状态代码：404

CodeStorageExceededException

您的 AWS 账户 已超出最大总代码大小。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

CodeVerificationFailedException

代码签名失败，一个或多个签名验证检查不匹配或过期，并且代码签名策略设置为“ENFORCE”。Lambda 阻止部署。

HTTP 状态代码：400

InvalidCodeSignatureException

代码签名未通过完整性检查。如果完整性检查失败，Lambda 会阻止部署，即使代码签名策略设置为 WARN 也是如此。

HTTP 状态代码：400

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CreateFunctionUrlConfig

使用指定的配置参数创建 Lambda 函数 URL。函数 URL 是专用的 HTTP(S) 端点，可用于调用函数。

请求语法

```
POST /2021-10-31/functions/FunctionName/url?Qualifier=Qualifier HTTP/1.1
Content-type: application/json

{
   "AuthType": "string",
   "Cors": {
      "AllowCredentials": boolean,
      "AllowHeaders": [ "string" ],
      "AllowMethods": [ "string" ],
      "AllowOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAge": number
   },
   "InvokeMode": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(^\$LATEST$)|((?!^[0-9]+$)([a-zA-Z0-9-_]+))

请求正文

请求接受采用 JSON 格式的以下数据。

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

必需：是

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

必需：否

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：BUFFERED | RESPONSE_STREAM

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "AuthType": "string",
   "Cors": {
      "AllowCredentials": boolean,
      "AllowHeaders": [ "string" ],
      "AllowMethods": [ "string" ],
      "AllowOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAge": number
   },
   "CreationTime": "string",
   "FunctionArn": "string",
   "FunctionUrl": "string",
   "InvokeMode": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

CreationTime

创建函数 URL 时，使用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionUrl

函数的 HTTP URL 端点。

类型：字符串

长度限制：最小长度为 40。最大长度为 100。

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：`BUFFERED | RESPONSE_STREAM`

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK

- 适用于 Ruby V3 的 AWS SDK

DeleteAlias

删除 Lambda 函数别名

请求语法

```
DELETE /2015-03-31/functions/FunctionName/aliases/Name HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Name

别名的名称。

长度限制：最小长度为 1。长度上限为 128。

模式：`(?!^[0-9]+$)([a-zA-Z0-9-_]+)`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteCodeSigningConfig

删除代码签名配置。只有在任何函数都未使用代码签名配置的情况下，才能删除该配置。

请求语法

```
DELETE /2020-04-22/code-signing-configs/CodeSigningConfigArn HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteEventSourceMapping

删除事件源映射。您可以从 ListEventSourceMappings 的输出中获取映射的标识符。

删除事件源映射时，它会输入 Deleting 状态，并且可能不会在几秒钟内完全删除。

请求语法

```
DELETE /2015-03-31/event-source-mappings/UUID HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

UUID

事件源映射的标识符。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 202
Content-type: application/json

{
   "AmazonManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
```

```
      }
   },
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "EventSourceArn": "string",
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
         }
      ]
   },
   "FunctionArn": "string",
   "FunctionResponseTypes": [ "string" ],
   "LastModified": number,
   "LastProcessingResult": "string",
   "MaximumBatchingWindowInSeconds": number,
   "MaximumRecordAgeInSeconds": number,
   "MaximumRetryAttempts": number,
   "ParallelizationFactor": number,
   "Queues": [ "string" ],
   "ScalingConfig": {
      "MaximumConcurrency": number
   },
   "SelfManagedEventSource": {
      "Endpoints": {
         "string" : [ "string" ]
      }
   },
   "SelfManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "SourceAccessConfigurations": [
      {
         "Type": "string",
         "URI": "string"
      }
   ],
   "StartingPosition": "string",
   "StartingPositionTimestamp": number,
   "State": "string",
```

```
    "StateTransitionReason": "string",
    "Topics": [ "string" ],
    "TumblingWindowInSeconds": number,
    "UUID": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 202 响应。

服务以 JSON 格式返回以下数据。

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

默认值：因服务而异。对于 Amazon SQS，默认值为 10。对于所有其他服务，默认值为 100。

相关设置：当将 `BatchSize` 设置为一个大于 10 的值时，必须至少将 `MaximumBatchingWindowInSeconds` 设置为 1。

类型：整数

有效范围：最小值为 1。最大值为 10000。

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。默认值为 False。

类型：布尔值

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Apache Kafka 事件源）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

FunctionArn

Lambda 函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

LastModified

上次更新事件源映射或其状态更改的日期（以 Unix 时间秒为单位）。

类型：时间戳

LastProcessingResult

函数的最后一次 Lambda 调用结果。

类型：字符串

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。原定设置值为 -1，该值将最大年龄设置为无限。将该值设置为无限后，Lambda 永远不会删除旧记录。

> **Note**
>
> 最长记录期限的最小有效值为 60 秒。尽管小于 60 和大于 -1 的值都在参数的绝对范围内，但不允许使用它们

类型：整数

有效范围：最小值为 -1。最大值为 604800。

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。原定设置值为 -1，该值将最大重试次数设置为无限。将 MaximumRetryAttempts 设置为无限后，Lambda 会重试失败的记录，直到事件源中的记录过期为止。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。默认值是 1。

类型：整数

有效范围：最小值为 1。最大值为 10。

Queues

(Amazon MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：`[\s\S]*`

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

SourceAccessConfigurations

用于保护与定义事件源的身份验证协议数组 VPC 组件或虚拟化主机。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

State

事件源映射的状态。此状态可以是以下值之一：Creating、Enabling、Enabled、Disabling、Disabled、Updating 或 Deleting。

类型：字符串

StateTransitionReason

表明是用户还是 Lambda 对事件源映射进行了最后一次更改。

类型：字符串

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

UUID

事件源映射的标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceInUseException

该操作与资源的可用性冲突。例如，您试图在 CREATING 状态中更新事件源映射，或尝试删除当前处于 UPDATING 状态的事件源映射。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteFunction

删除 Lambda 函数。要删除特定函数版本，请使用 Qualifier 参数。否则，将删除所有版本和别名。这并不要求用户具有 DeleteAlias 的显式权限。

要删除调用函数的 Lambda 事件源映射，请使用 DeleteEventSourceMapping。对于直接调用函数的 AWS 服务 和资源，请在最初配置它的服务中删除触发器。

请求语法

```
DELETE /2015-03-31/functions/FunctionName?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称或版本。

名称格式

- 函数名称 – my-function（仅限名称）、my-function:1（包含版本）。
- 函数 ARN – arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN – 123456789012:function:my-function。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

指定要删除的版本。您无法删除别名引用的版本。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteFunctionCodeSigningConfig

从函数中删除代码签名配置。

请求语法

```
DELETE /2020-06-30/functions/FunctionName/code-signing-config HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeSigningConfigNotFoundException

指定的代码签名配置不存在。

HTTP 状态代码：404

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteFunctionConcurrency

从函数中删除并发执行限制。

请求语法

```
DELETE /2017-10-31/functions/FunctionName/concurrency HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteFunctionEventInvokeConfig

删除函数、版本或别名的异步调用配置。

要配置异步调用的选项，请使用 PutFunctionEventInvokeConfig。

请求语法

```
DELETE /2019-09-25/functions/FunctionName/event-invoke-config?Qualifier=Qualifier
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – my-function（仅限名称）、my-function:v1（具有别名）。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN - 123456789012:function:my-function。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteFunctionUrlConfig

删除 Lambda 函数 URL。删除函数 URL 后，将无法恢复。创建一个新函数 URL 会导致不同的 URL 地址。

请求语法

```
DELETE /2021-10-31/functions/FunctionName/url?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

别名。

长度限制：最小长度为 1。长度上限为 128。

模式：`(^\$LATEST$)|((?!^[0-9]+$)([a-zA-Z0-9-_]+))`

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteLayerVersion

删除某个 AWS Lambda 层的某个版本。已删除的版本无法再查看或添加到函数中。为了避免中断函数，版本的副本将保留在 Lambda 中，直到没有函数引用它。

请求语法

```
DELETE /2018-10-31/layers/LayerName/versions/VersionNumber HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：是

VersionNumber

版本号。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeleteProvisionedConcurrencyConfig

删除为函数预置的并发配置。

请求语法

```
DELETE /2019-09-30/functions/FunctionName/provisioned-concurrency?Qualifier=Qualifier
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetAccountSettings

检索有关您账户限制和在AWS区域中用量的详细信息。

请求语法

```
GET /2016-08-19/account-settings/ HTTP/1.1
```

URI 请求参数

该请求不使用任何 URI 参数。

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AccountLimit": {
      "CodeSizeUnzipped": number,
      "CodeSizeZipped": number,
      "ConcurrentExecutions": number,
      "TotalCodeSize": number,
      "UnreservedConcurrentExecutions": number
   },
   "AccountUsage": {
      "FunctionCount": number,
      "TotalCodeSize": number
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AccountLimit

与并发和代码存储相关的限制。

类型：AccountLimit 对象

AccountUsage

正在使用的函数数量和存储容量。

类型：AccountUsage 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK

- 适用于 Ruby V3 的 AWS SDK

GetAlias

返回有关 Lambda 函数别名的详细信息。

请求语法

```
GET /2015-03-31/functions/FunctionName/aliases/Name HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - MyFunction。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:MyFunction。
- 部分 ARN - 123456789012:function:MyFunction。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Name

别名的名称。

长度限制：最小长度为 1。长度上限为 128。

模式：(?!^[0-9]+$)([a-zA-Z0-9-_]+)

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AliasArn": "string",
   "Description": "string",
   "FunctionVersion": "string",
   "Name": "string",
   "RevisionId": "string",
   "RoutingConfig": {
      "AdditionalVersionWeights": {
         "string" : number
      }
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AliasArn

是别名的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

Name

别名的名称。

类型：字符串

长度限制：长度下限为 1。长度上限为 128。

模式：(?!^[0-9]+$)([a-zA-Z0-9-_]+)

RevisionId

更新别名时更改的唯一标识符。

类型：字符串

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetCodeSigningConfig

返回有关指定代码签名配置的信息。

请求语法

```
GET /2020-04-22/code-signing-configs/CodeSigningConfigArn HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CodeSigningConfig": {
      "AllowedPublishers": {
         "SigningProfileVersionArns": [ "string" ]
      },
      "CodeSigningConfigArn": "string",
      "CodeSigningConfigId": "string",
      "CodeSigningPolicies": {
         "UntrustedArtifactOnDeployment": "string"
```

```
        },
        "Description": "string",
        "LastModified": "string"
    }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CodeSigningConfig

代码签名配置

类型：CodeSigningConfig 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetEventSourceMapping

返回有关事件源映射的详细信息。您可以从 ListEventSourceMappings 的输出中获取映射的标识符。

请求语法

```
GET /2015-03-31/event-source-mappings/UUID HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

UUID

事件源映射的标识符。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AmazonManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
```

```
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "EventSourceArn": "string",
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
         }
      ]
   },
   "FunctionArn": "string",
   "FunctionResponseTypes": [ "string" ],
   "LastModified": number,
   "LastProcessingResult": "string",
   "MaximumBatchingWindowInSeconds": number,
   "MaximumRecordAgeInSeconds": number,
   "MaximumRetryAttempts": number,
   "ParallelizationFactor": number,
   "Queues": [ "string" ],
   "ScalingConfig": {
      "MaximumConcurrency": number
   },
   "SelfManagedEventSource": {
      "Endpoints": {
         "string" : [ "string" ]
      }
   },
   "SelfManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "SourceAccessConfigurations": [
      {
         "Type": "string",
         "URI": "string"
      }
   ],
   "StartingPosition": "string",
   "StartingPositionTimestamp": number,
   "State": "string",
   "StateTransitionReason": "string",
   "Topics": [ "string" ],
```

```
    "TumblingWindowInSeconds": number,
    "UUID": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

默认值：因服务而异。对于 Amazon SQS，默认值为 10。对于所有其他服务，默认值为 100。

相关设置：当将 `BatchSize` 设置为一个大于 10 的值时，必须至少将 `MaximumBatchingWindowInSeconds` 设置为 1。

类型：整数

有效范围：最小值为 1。最大值为 10000。

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。默认值为 False。

类型：布尔值

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Apache Kafka 事件源）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

FunctionArn

Lambda 函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

LastModified

上次更新事件源映射或其状态更改的日期（以 Unix 时间秒为单位）。

类型：时间戳

LastProcessingResult

函数的最后一次 Lambda 调用结果。

类型：字符串

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。原定设置值为 -1，该值将最大年龄设置为无限。将该值设置为无限后，Lambda 永远不会删除旧记录。

> **Note**
>
> 最长记录期限的最小有效值为 60 秒。尽管小于 60 和大于 -1 的值都在参数的绝对范围内，但不允许使用它们

类型：整数

有效范围：最小值为 -1。最大值为 604800。

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。原定设置值为 -1，该值将最大重试次数设置为无限。将 MaximumRetryAttempts 设置为无限后，Lambda 会重试失败的记录，直到事件源中的记录过期为止。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。默认值是 1。

类型：整数

有效范围：最小值为 1。最大值为 10。

Queues

(Amazon MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：`[\s\S]*`

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

SourceAccessConfigurations

用于保护与定义事件源的身份验证协议数组 VPC 组件或虚拟化主机。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

State

事件源映射的状态。此状态可以是以下值之一：Creating、Enabling、Enabled、Disabling、Disabled、Updating 或 Deleting。

类型：字符串

StateTransitionReason

表明是用户还是 Lambda 对事件源映射进行了最后一次更改。

类型：字符串

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

UUID

事件源映射的标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunction

返回有关函数或函数版本的信息，其中包含部署包下载链接，有效时间 10 分钟。如果指定函数版本，则仅返回特定于该版本的详细信息。

请求语法

```
GET /2015-03-31/functions/FunctionName?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

指定版本或别名以获取有关函数发布版本的详细信息。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Code": {
      "ImageUri": "string",
      "Location": "string",
      "RepositoryType": "string",
      "ResolvedImageUri": "string"
   },
   "Concurrency": {
      "ReservedConcurrentExecutions": number
   },
   "Configuration": {
      "Architectures": [ "string" ],
      "CodeSha256": "string",
      "CodeSize": number,
      "DeadLetterConfig": {
         "TargetArn": "string"
      },
      "Description": "string",
      "Environment": {
         "Error": {
            "ErrorCode": "string",
            "Message": "string"
         },
         "Variables": {
            "string" : "string"
         }
      },
      "EphemeralStorage": {
         "Size": number
      },
      "FileSystemConfigs": [
         {
            "Arn": "string",
            "LocalMountPath": "string"
```

```
      }
   ],
   "FunctionArn": "string",
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
```

```
            "Message": "string"
         },
         "RuntimeVersionArn": "string"
      },
      "SigningJobArn": "string",
      "SigningProfileVersionArn": "string",
      "SnapStart": {
         "ApplyOn": "string",
         "OptimizationStatus": "string"
      },
      "State": "string",
      "StateReason": "string",
      "StateReasonCode": "string",
      "Timeout": number,
      "TracingConfig": {
         "Mode": "string"
      },
      "Version": "string",
      "VpcConfig": {
         "Ipv6AllowedForDualStack": boolean,
         "SecurityGroupIds": [ "string" ],
         "SubnetIds": [ "string" ],
         "VpcId": "string"
      }
   },
   "Tags": {
      "string" : "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Code

函数或版本的部署包。

类型：FunctionCodeLocation 对象

Concurrency

函数的预留并发。

类型：Concurrency 对象

Configuration

函数或版本的配置。

类型：FunctionConfiguration 对象

Tags

函数的标签。

类型：字符串到字符串映射

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunctionCodeSigningConfig

返回指定函数的代码签名配置。

请求语法

```
GET /2020-06-30/functions/FunctionName/code-signing-config HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "CodeSigningConfigArn": "string",
   "FunctionName": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunctionConcurrency

返回有关函数的预留并发配置的详细信息。要为函数设置并发限制，请使用 PutFunctionConcurrency。

请求语法

```
GET /2019-09-30/functions/FunctionName/concurrency HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json
```

```
{
   "ReservedConcurrentExecutions": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

ReservedConcurrentExecutions

为函数预留的同时执行次数。

类型：整数

有效范围：最小值为 0。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunctionConfiguration

返回 Lambda 函数或版本的版本特定设置。输出仅包含在函数版本之间可能有所不同的选项。要修改这些设置，请使用 UpdateFunctionConfiguration。

要获取某个函数的所有详细信息，包括函数级别设置，请使用 GetFunction。

请求语法

```
GET /2015-03-31/functions/FunctionName/configuration?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

指定版本或别名以获取有关函数发布版本的详细信息。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Architectures": [ "string" ],
   "CodeSha256": "string",
   "CodeSize": number,
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionArn": "string",
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
```

```
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "RuntimeVersionArn": "string"
   },
   "SigningJobArn": "string",
   "SigningProfileVersionArn": "string",
   "SnapStart": {
      "ApplyOn": "string",
      "OptimizationStatus": "string"
   },
```

```
   "State": "string",
   "StateReason": "string",
   "StateReasonCode": "string",
   "Timeout": number,
   "TracingConfig": {
      "Mode": "string"
   },
   "Version": "string",
   "VpcConfig": {
      "Ipv6AllowedForDualStack": boolean,
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcId": "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：`x86_64 | arm64`

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 `Successful`。

类型：字符串

有效值：`Successful | Failed | InProgress`

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：`EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError`

Layers

函数的层。

类型：Layer 对象数组

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

PackageType

部署程序包的类型。将容器映像设置为 Image，然后将 .zip 文件存档设置为 Zip。

类型：字符串

有效值：Zip | Image

RevisionId

函数或别名的最新更新版本。

类型：字符串

Role

函数的执行角色。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

RuntimeVersionConfig

运行时的 ARN 和发生的任何错误。

类型：RuntimeVersionConfig 对象

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SnapStart

发布函数版本时，将 `ApplyOn` 设置为 `PublishedVersions` 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

State

函数的当前状态。当状态为 `Inactive`，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：`Pending | Active | Inactive | Failed`

StateReason

函数当前状态的原因。

类型：字符串

StateReasonCode

函数当前状态的原因代码。当代码为 `Creating` 时，无法调用或修改该函数。

类型：字符串

有效值：`Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError`

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunctionEventInvokeConfig

检索函数、版本或别名的异步调用配置。

要配置异步调用的选项，请使用 PutFunctionEventInvokeConfig。

请求语法

```
GET /2019-09-25/functions/FunctionName/event-invoke-config?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – my-function（仅限名称）、my-function:v1（具有别名）。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN - 123456789012:function:my-function。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "FunctionArn": "string",
   "LastModified": number,
   "MaximumEventAgeInSeconds": number,
   "MaximumRetryAttempts": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。
- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

LastModified

上次更新配置的日期和时间（以 Unix 时间秒为单位）。

类型：时间戳

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetFunctionUrlConfig

返回有关 Lambda 函数 URL 的详细信息。

请求语法

```
GET /2021-10-31/functions/FunctionName/url?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

别名。

长度限制：最小长度为 1。长度上限为 128。

模式：`(^\$LATEST$)|((?!^[0-9]+$)([a-zA-Z0-9-_]+))`

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AuthType": "string",
   "Cors": {
      "AllowCredentials": boolean,
      "AllowHeaders": [ "string" ],
      "AllowMethods": [ "string" ],
      "AllowOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAge": number
   },
   "CreationTime": "string",
   "FunctionArn": "string",
   "FunctionUrl": "string",
   "InvokeMode": "string",
   "LastModifiedTime": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

CreationTime

创建函数 URL 时，使用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionUrl

函数的 HTTP URL 端点。

类型：字符串

长度限制：最小长度为 40。最大长度为 100。

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：`BUFFERED | RESPONSE_STREAM`

LastModifiedTime

上次更新函数 URL 时，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetLayerVersion

返回有关某个 AWS Lambda 层的某个版本的信息，其中包含层归档的下载链接，有效时间 10 分钟。

请求语法

```
GET /2018-10-31/layers/LayerName/versions/VersionNumber HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：是

VersionNumber

版本号。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CompatibleArchitectures": [ "string" ],
```

```
   "CompatibleRuntimes": [ "string" ],
   "Content": {
      "CodeSha256": "string",
      "CodeSize": number,
      "Location": "string",
      "SigningJobArn": "string",
      "SigningProfileVersionArn": "string"
   },
   "CreatedDate": "string",
   "Description": "string",
   "LayerArn": "string",
   "LayerVersionArn": "string",
   "LicenseInfo": "string",
   "Version": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CompatibleArchitectures

兼容指令集架构列表。

类型：字符串数组

数组成员：最多 15 项。

有效值：`x86_64 | arm64`

CompatibleRuntimes

层的兼容运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串数组

数组成员：最多 15 项。

有效值：`nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11`

```
| python2.7 | python3.6 | python3.7 | python3.8 | python3.9 |
dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6
| nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2
| nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x |
provided.al2023 | python3.12 | java21
```

Content

有关层版本的详细信息。

类型：LayerVersionContentOutput 对象

CreatedDate

层版本的创建日期，为 ISO-8601 格式 (YYYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

Description

关于版本的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

LayerArn

层的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+`

LayerVersionArn

层版本的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

LicenseInfo

层的软件许可证。

类型：字符串

长度约束：最大长度为 512。

Version

版本号。

类型：长整型

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetLayerVersionByArn

返回有关某个 AWS Lambda 层的某个版本的信息，其中包含层归档的下载链接，有效时间 10 分钟。

请求语法

```
GET /2018-10-31/layers?find=LayerVersion&Arn=Arn HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

Arn

层版本的 ARN。

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CompatibleArchitectures": [ "string" ],
   "CompatibleRuntimes": [ "string" ],
   "Content": {
      "CodeSha256": "string",
      "CodeSize": number,
      "Location": "string",
      "SigningJobArn": "string",
```

```
      "SigningProfileVersionArn": "string"
   },
   "CreatedDate": "string",
   "Description": "string",
   "LayerArn": "string",
   "LayerVersionArn": "string",
   "LicenseInfo": "string",
   "Version": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CompatibleArchitectures

兼容指令集架构列表。

类型：字符串数组

数组成员：最多 15 项。

有效值：x86_64 | arm64

CompatibleRuntimes

层的兼容运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串数组

数组成员：最多 15 项。

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

Content

有关层版本的详细信息。

类型：LayerVersionContentOutput 对象

CreatedDate

层版本的创建日期，为 ISO-8601 格式 (YYYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

Description

关于版本的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

LayerArn

层的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+`

LayerVersionArn

层版本的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

LicenseInfo

层的软件许可证。

类型：字符串

长度约束：最大长度为 512。

Version

版本号。

类型：长整型

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetLayerVersionPolicy

返回某个 AWS Lambda 层的某个版本的权限策略。有关更多信息，请参阅AddLayerVersionPermission。

请求语法

```
GET /2018-10-31/layers/LayerName/versions/VersionNumber/policy HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：是

VersionNumber

版本号。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
```

```
    "Policy": "string",
    "RevisionId": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Policy

策略文档。

类型：字符串

RevisionId

策略当前修订版的唯一标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetPolicy

返回函数、版本或别名的基于资源的 IAM policy。

请求语法

```
GET /2015-03-31/functions/FunctionName/policy?Qualifier=Qualifier HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

指定版本或别名以获取该资源的策略。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Policy": "string",
   "RevisionId": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Policy

基于资源的策略。

类型：字符串

RevisionId

策略当前修订版的唯一标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetProvisionedConcurrencyConfig

检索函数的别名或版本的预置并发配置。

请求语法

```
GET /2019-09-30/functions/FunctionName/provisioned-concurrency?Qualifier=Qualifier
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AllocatedProvisionedConcurrentExecutions": number,
   "AvailableProvisionedConcurrentExecutions": number,
   "LastModified": "string",
   "RequestedProvisionedConcurrentExecutions": number,
   "Status": "string",
   "StatusReason": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AllocatedProvisionedConcurrentExecutions

输入要分配的预置并发数量。在线性部署和金丝雀部署期间使用加权别名时，该值会根据为函数版本预配置的并发量而波动。

类型：整数

有效范围：最小值为 0。

AvailableProvisionedConcurrentExecutions

可用的预置并发数量。

类型：整数

有效范围：最小值为 0。

LastModified

用户上次更新配置的日期和时间，格式为 ISO 8601。

类型：字符串

RequestedProvisionedConcurrentExecutions

请求的预置并发数量。

类型：整数

有效范围：最小值为 1。

Status

分配过程的状态。

类型：字符串

有效值：`IN_PROGRESS | READY | FAILED`

StatusReason

分配失败的原因，即无法分配预置并发的原因。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ProvisionedConcurrencyConfigNotFoundException

指定的配置不存在。

HTTP 状态代码：404

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

GetRuntimeManagementConfig

检索函数版本的运行时管理配置。如果运行时更新模式为手动，则将包括运行时版本的 ARN 和运行时更新模式。如果运行时更新模式为自动或函数更新，则将包括运行时更新模式，并且会为 ARN 返回 null。有关更多信息，请参阅 Runtime updates（运行时更新）。

请求语法

```
GET /2021-07-20/functions/FunctionName/runtime-management-config?Qualifier=Qualifier
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – my-function。
- 函数 ARN – arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN – 123456789012:function:my-function。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

指定该函数的版本。可以是 $LATEST 或已发布的版本号。如果未指定任何值，则将返回 $LATEST 版本的配置。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "FunctionArn": "string",
   "RuntimeVersionArn": "string",
   "UpdateRuntimeOn": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

RuntimeVersionArn

配置函数所使用的运行时的 ARN。如果运行时更新模式为手动，则将返回 ARN，否则将返回 `null`。

类型：字符串

长度限制：最小长度为 26。最大长度为 2048。

模式：`^arn:(aws[a-zA-Z-]*):lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}::runtime:.+$`

UpdateRuntimeOn

函数的当前运行时更新模式。

类型：字符串

有效值：`Auto | Manual | FunctionUpdate`

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Invoke

调用 Lambda 函数。您可以同步调用函数（并等待响应），也可以异步调用函数。默认情况下，Lambda 会同步调用函数（即 `InvocationType` 是 `RequestResponse`）。要异步调用函数，请将 `InvocationType` 设置为 `Event`。Lambda 将 `ClientContext` 对象传递给函数（仅限同步调用）。

对于同步调用，有关函数响应的详细信息（包括错误）包含在响应正文和标题中。对于任一调用类型，您可以在执行日志和跟踪中找到更多信息。

当发生错误时，您的函数可能会被多次调用。重试行为因错误类型、客户端、事件源和调用类型而异。例如，如果您异步调用函数并返回错误，则 Lambda 最多再执行两次该函数。有关更多信息，请参阅 Lambda 中的错误处理和自动重试。

对于异步调用，Lambda 会将事件添加到队列，然后将它们发送到函数。如果您的函数没有足够的容量来跟上队列，事件可能会丢失。有时，您的函数可能会多次收到相同的事件，即使没有发生错误。要保留未处理的事件，请使用死信队列配置函数。

API 响应中的状态代码不反映函数错误。错误代码是为阻止函数执行的错误而预留的，例如权限错误、限额错误或与函数的代码和配置有关的问题。例如，如果运行该函数会导致您超出账户级别（`ConcurrentInvocationLimitExceeded`）或函数级别（`ReservedFunctionConcurrentInvocationLimitExceeded`）的并发限制，则 Lambda 将返回 `TooManyRequestsException`。

对于超时很长的函数，在等待响应的同步调用期间，客户端可能会断开连接。配置您的 HTTP 客户端、软件开发工具包、防火墙、代理或操作系统，以允许针对超时或保持活动设置保持长时间的连接。

此操作需要 lambda:InvokeFunction 操作的权限。有关如何设置跨账户调用权限的详细信息，请参阅向其他账户授予函数访问权。

请求语法

```
POST /2015-03-31/functions/FunctionName/invocations?Qualifier=Qualifier HTTP/1.1
X-Amz-Invocation-Type: InvocationType
X-Amz-Log-Type: LogType
X-Amz-Client-Context: ClientContext

Payload
```

URI 请求参数

请求使用以下 URI 参数。

ClientContext

有关要传递给上下文对象中的函数的调用客户端的 base64 编码数据，最多为 3,583 字节。Lambda 将 `ClientContext` 对象传递给函数（仅限同步调用）。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

InvocationType

从以下选项中进行选择。

- `RequestResponse`（默认）– 同步调用函数。保持连接打开，直到函数返回响应或超时。API 响应包括函数响应和其他数据。
- `Event` – 异步调用函数。将多次失败的事件发送到函数的死信队列（如果已配置）。API 响应仅包含状态代码。
- `DryRun` – 验证参数值并验证用户或角色是否具有调用函数的权限。

有效值：`Event | RequestResponse | DryRun`

LogType

设置 `Tail` 以在响应中包含执行日志。仅适用于同步调用的函数。

有效值：None | Tail

Qualifier

指定版本或别名以调用函数的已发布版本。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

请求接受以下二进制数据。

Payload

您想要作为输入提供到您的 Lambda 函数的 JSON。

您可以直接输入 JSON。例如，--payload '{ "key": "value" }'。此外，您还可以指定文件路径。例如，--payload file://payload.json。

响应语法

```
HTTP/1.1 StatusCode
X-Amz-Function-Error: FunctionError
X-Amz-Log-Result: LogResult
X-Amz-Executed-Version: ExecutedVersion

Payload
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 响应。

StatusCode

对于成功请求，HTTP 状态代码在 200 范围内。对于 RequestResponse 调用类型，状态代码为 200。对于 Event 调用类型，状态代码为 202。对于 DryRun 调用类型，状态代码为 204。

响应将返回以下 HTTP 标头。

ExecutedVersion

执行的函数的版本。当您调用带有别名的函数时，这将指示别名解析为哪个版本。

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

FunctionError

如果存在，则表示函数执行过程中发生错误。响应负载中包含有关错误的详细信息。

LogResult

Base64 编码执行日志的最后 4 KB。

响应将以下内容作为 HTTP 正文返回。

Payload

来自函数或错误对象的响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

EC2AccessDeniedException

需要额外的权限才能配置 VPC 设置。

HTTP 状态代码：502

EC2ThrottledException

AWS Lambda 是由 Amazon EC2 在 Lambda 函数初始化期间使用为函数提供的执行角色进行节流的。

HTTP 状态代码：502

EC2UnexpectedException

AWS Lambda 在设置 Lambda 函数时收到意外的 Amazon EC2 客户端异常。

HTTP 状态代码：502

EFSIOException

从连接的文件系统读取或向其中写入时发生了错误。

HTTP 状态代码：410

EFSMountConnectivityException

Lambda 函数无法与配置的文件系统建立网络连接。

HTTP 状态代码：408

EFSMountFailureException

由于权限或配置问题，Lambda 函数无法挂载配置的文件系统。

HTTP 状态代码：403

EFSMountTimeoutException

Lambda 函数能够连接到配置的文件系统，但挂载操作超时。

HTTP 状态代码：408

ENILimitReachedException

AWS Lambda 无法在 VPC 中创建指定为 Lambda 函数配置一部分的弹性网络接口，因为已达到网络接口限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：502

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

InvalidRequestContentException

请求正文无法解析为 JSON。

HTTP 状态代码：400

InvalidRuntimeException

运行时或运行时版本不受支持。

HTTP 状态代码：502

InvalidSecurityGroupIDException

Lambda 函数 VPC 配置中提供的安全组 ID 无效。

HTTP 状态代码：502

InvalidSubnetIDException

Lambda 函数 VPC 配置中提供的子网 ID 无效。

HTTP 状态代码：502

InvalidZipFileException

AWS Lambda 无法解压缩部署程序包。

HTTP 状态代码：502

KMSAccessDeniedException

Lambda 无法解密环境变量，因为对 AWS KMS 的访问已被拒绝。检查 Lambda 函数的 KMS 权限。

HTTP 状态代码：502

KMSDisabledException

Lambda 无法解密环境变量，因为使用的 AWS KMS key 已被禁用。检查 Lambda 函数的 KMS 密钥设置。

HTTP 状态代码：502

KMSInvalidStateException

Lambda 无法解密环境变量，因为所使用的 AWS KMS key 的状态对解密无效。检查函数的 KMS 密钥设置。

HTTP 状态代码：502

KMSNotFoundException

Lambda 无法解密环境变量，因为找不到 AWS KMS key。检查函数的 KMS 密钥设置。

HTTP 状态代码：502

RecursiveInvocationException

Lambda 检测到您的函数在涉及其他 AWS 资源的递归循环中被调用，因而停止了调用您的函数。

HTTP 状态代码：400

RequestTooLargeException

请求负载已超出 Invoke 请求正文 JSON 输入限额。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：413

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ResourceNotReadyException

函数处于非活动状态，其 VPC 连接不再可用。等待 VPC 连接重新建立，然后重试。

HTTP 状态代码：502

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

SnapStartException

afterRestore() 运行时挂钩遇到错误。有关更多信息，请查看 Amazon CloudWatch 日志。

HTTP 状态代码：400

SnapStartNotReadyException

Lambda 正在初始化函数。您可以在函数状态变为 Active 时调用该函数。

HTTP 状态代码：409

SnapStartTimeoutException

Lambda 无法在超时限制内还原快照。

HTTP 状态代码：408

SubnetIPAddressLimitReachedException

由于配置的一个或多个子网没有可用的 IP 地址，因此 AWS Lambda 无法为 Lambda 函数设置 VPC 访问。

HTTP 状态代码：502

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

UnsupportedMediaTypeException

`Invoke` 的内容类型请求体不是 JSON。

HTTP 状态代码：415

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

InvokeAsync

此操作已弃用。

> **Important**
>
> 对于异步函数调用，请使用 Invoke。

异步调用函数。

> **Note**
>
> 如果您确实使用了 InvokeAsync 操作，请注意它不支持使用 X-Ray 活动跟踪。即使开启了 X-Ray 活动追踪，也不会将跟踪 ID 传播到该函数。

请求语法

```
POST /2014-11-13/functions/FunctionName/invoke-async/ HTTP/1.1

InvokeArgs
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求正文

请求接受以下二进制数据。

InvokeArgs

您想要作为输入提供到您的 Lambda 函数的 JSON。

必需：是

响应语法

```
HTTP/1.1 Status
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 响应。

Status

状态代码。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidRequestContentException

请求正文无法解析为 JSON。

HTTP 状态代码：400

InvalidRuntimeException

运行时或运行时版本不受支持。

HTTP 状态代码：502

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

InvokeWithResponseStream

配置 Lambda 函数以将响应负载流式传输回客户端。有关更多信息，请参阅配置 Lambda 函数以流式处理响应。

此操作需要 lambda:InvokeFunction 操作的权限。有关如何设置跨账户调用权限的详细信息，请参阅向其他账户授予函数访问权。

请求语法

```
POST /2021-11-15/functions/FunctionName/response-streaming-invocations?
Qualifier=Qualifier HTTP/1.1
X-Amz-Invocation-Type: InvocationType
X-Amz-Log-Type: LogType
X-Amz-Client-Context: ClientContext

Payload
```

URI 请求参数

请求使用以下 URI 参数。

ClientContext

有关要传递给上下文对象中的函数的调用客户端的 base64 编码数据，最多为 3,583 字节。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

InvocationType

使用以下选项之一：

- RequestResponse（默认）– 同步调用函数。保持连接打开，直到函数返回响应或超时。API 操作响应包括函数响应和其他数据。
- DryRun – 验证参数值并验证 IAM 用户或角色是否具有调用函数的权限。

有效值：RequestResponse | DryRun

LogType

设置 Tail 以在响应中包含执行日志。仅适用于同步调用的函数。

有效值：None | Tail

Qualifier

别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

请求接受以下二进制数据。

Payload

您想要作为输入提供到您的 Lambda 函数的 JSON。

您可以直接输入 JSON。例如，--payload '{ "key": "value" }'。此外，您还可以指定文件路径。例如，--payload file://payload.json。

响应语法

```
HTTP/1.1 StatusCode
X-Amz-Executed-Version: ExecutedVersion
Content-Type: ResponseStreamContentType
Content-type: application/json
```

```
{
   "InvokeComplete": {
      "ErrorCode": "string",
      "ErrorDetails": "string",
      "LogResult": "string"
   },
   "PayloadChunk": {
      "Payload": blob
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 响应。

StatusCode

对于成功请求，HTTP 状态代码在 200 范围内。对于 RequestResponse 调用类型，状态代码为 200。对于 DryRun 调用类型，状态代码为 204。

响应将返回以下 HTTP 标头。

ExecutedVersion

执行的函数的版本。当您调用带有别名的函数时，这将指示别名解析为哪个版本。

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

ResponseStreamContentType

流返回的数据类型。

服务以 JSON 格式返回以下数据。

InvokeComplete

当流结束并且所有负载区块均已返回后所返回的对象。

类型：InvokeWithResponseStreamCompleteEvent 对象

PayloadChunk

流式处理响应负载的区块。

类型：InvokeResponseStreamUpdate 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

EC2AccessDeniedException

需要额外的权限才能配置 VPC 设置。

HTTP 状态代码：502

EC2ThrottledException

AWS Lambda 是由 Amazon EC2 在 Lambda 函数初始化期间使用为函数提供的执行角色进行节流的。

HTTP 状态代码：502

EC2UnexpectedException

AWS Lambda 在设置 Lambda 函数时收到意外的 Amazon EC2 客户端异常。

HTTP 状态代码：502

EFSIOException

从连接的文件系统读取或向其中写入时发生了错误。

HTTP 状态代码：410

EFSMountConnectivityException

Lambda 函数无法与配置的文件系统建立网络连接。

HTTP 状态代码：408

EFSMountFailureException

由于权限或配置问题，Lambda 函数无法挂载配置的文件系统。

HTTP 状态代码：403

EFSMountTimeoutException

Lambda 函数能够连接到配置的文件系统，但挂载操作超时。

HTTP 状态代码：408

ENILimitReachedException

AWS Lambda 无法在 VPC 中创建指定为 Lambda 函数配置一部分的弹性网络接口，因为已达到网络接口限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：502

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

InvalidRequestContentException

请求正文无法解析为 JSON。

HTTP 状态代码：400

InvalidRuntimeException

运行时或运行时版本不受支持。

HTTP 状态代码：502

InvalidSecurityGroupIDException

Lambda 函数 VPC 配置中提供的安全组 ID 无效。

HTTP 状态代码：502

InvalidSubnetIDException

Lambda 函数 VPC 配置中提供的子网 ID 无效。

HTTP 状态代码：502

InvalidZipFileException

AWS Lambda 无法解压缩部署程序包。

HTTP 状态代码：502

KMSAccessDeniedException

Lambda 无法解密环境变量，因为对 AWS KMS 的访问已被拒绝。检查 Lambda 函数的 KMS 权限。

HTTP 状态代码：502

KMSDisabledException

Lambda 无法解密环境变量，因为使用的 AWS KMS key 已被禁用。检查 Lambda 函数的 KMS 密钥设置。

HTTP 状态代码：502

KMSInvalidStateException

Lambda 无法解密环境变量，因为所使用的 AWS KMS key 的状态对解密无效。检查函数的 KMS 密钥设置。

HTTP 状态代码：502

KMSNotFoundException

Lambda 无法解密环境变量，因为找不到 AWS KMS key。检查函数的 KMS 密钥设置。

HTTP 状态代码：502

RecursiveInvocationException

Lambda 检测到您的函数在涉及其他 AWS 资源的递归循环中被调用，因而停止了调用您的函数。

HTTP 状态代码：400

RequestTooLargeException

请求负载已超出 `Invoke` 请求正文 JSON 输入限额。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：413

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ResourceNotReadyException

函数处于非活动状态，其 VPC 连接不再可用。等待 VPC 连接重新建立，然后重试。

HTTP 状态代码：502

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

SnapStartException

`afterRestore()` 运行时挂钩遇到错误。有关更多信息，请查看 Amazon CloudWatch 日志。

HTTP 状态代码：400

SnapStartNotReadyException

Lambda 正在初始化函数。您可以在函数状态变为 `Active` 时调用该函数。

HTTP 状态代码：409

SnapStartTimeoutException

Lambda 无法在超时限制内还原快照。

HTTP 状态代码：408

SubnetIPAddressLimitReachedException

由于配置的一个或多个子网没有可用的 IP 地址，因此 AWS Lambda 无法为 Lambda 函数设置 VPC 访问。

HTTP 状态代码：502

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

UnsupportedMediaTypeException

`Invoke` 的内容类型请求体不是 JSON。

HTTP 状态代码：415

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListAliases

返回为 Lambda 函数创建的别名列表。

请求语法

```
GET /2015-03-31/functions/FunctionName/aliases?
FunctionVersion=FunctionVersion&Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

FunctionVersion

指定函数版本以仅列出调用该版本的别名。

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

限制别名返回的数量。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Aliases": [
      {
         "AliasArn": "string",
         "Description": "string",
         "FunctionVersion": "string",
         "Name": "string",
         "RevisionId": "string",
         "RoutingConfig": {
            "AdditionalVersionWeights": {
               "string" : number
            }
         }
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Aliases

别名列表。

类型：AliasConfiguration 对象数组

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListCodeSigningConfigs

返回代码签名配置列表。每次调用最多返回 10000 个配置。您可以使用 MaxItems 参数以便在每次调用时返回较少的配置。

请求语法

```
GET /2020-04-22/code-signing-configs/?Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

要返回的最大项目数量。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CodeSigningConfigs": [
      {
         "AllowedPublishers": {
            "SigningProfileVersionArns": [ "string" ]
         },
         "CodeSigningConfigArn": "string",
         "CodeSigningConfigId": "string",
```

```
            "CodeSigningPolicies": {
               "UntrustedArtifactOnDeployment": "string"
            },
            "Description": "string",
            "LastModified": "string"
         }
      ],
      "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CodeSigningConfigs

代码签名配置

类型：CodeSigningConfig 对象数组

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListEventSourceMappings

列出事件源映射。指定 EventSourceArn 以仅显示单个事件源的事件源映射。

请求语法

```
GET /2015-03-31/event-source-mappings/?
EventSourceArn=EventSourceArn&FunctionName=FunctionName&Marker=Marker&MaxItems=MaxItems
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

EventSourceArn

事件源的 Amazon 资源名称（ARN）。

- Amazon Kinesis – 数据流或流使用者的 ARN。
- Amazon DynamoDB Streams – 流的 ARN。
- Amazon Simple Queue Service – 队列的 ARN。
- Amazon Managed Streaming for Apache Kafka – 集群的 ARN 或 VPC 连接的 ARN（用于跨账户事件源映射）。
- Amazon MQ – 代理的 ARN。
- Amazon DocumentDB – DocumentDB 更改流的 ARN。

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `MyFunction`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 版本或别名 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction:PROD`。
- 部分 ARN – `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度将限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Marker

由上一次调用返回的分页令牌。

MaxItems

要返回的事件源映射最大数量。请注意，即使您设置的数字较高，ListEventSourceMappings 在每个响应中最多返回 100 个项目。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "EventSourceMappings": [
      {
         "AmazonManagedKafkaEventSourceConfig": {
            "ConsumerGroupId": "string"
         },
         "BatchSize": number,
         "BisectBatchOnFunctionError": boolean,
         "DestinationConfig": {
            "OnFailure": {
               "Destination": "string"
            },
            "OnSuccess": {
               "Destination": "string"
            }
         },
         "DocumentDBEventSourceConfig": {
```

```
            "CollectionName": "string",
            "DatabaseName": "string",
            "FullDocument": "string"
         },
         "EventSourceArn": "string",
         "FilterCriteria": {
            "Filters": [
               {
                  "Pattern": "string"
               }
            ]
         },
         "FunctionArn": "string",
         "FunctionResponseTypes": [ "string" ],
         "LastModified": number,
         "LastProcessingResult": "string",
         "MaximumBatchingWindowInSeconds": number,
         "MaximumRecordAgeInSeconds": number,
         "MaximumRetryAttempts": number,
         "ParallelizationFactor": number,
         "Queues": [ "string" ],
         "ScalingConfig": {
            "MaximumConcurrency": number
         },
         "SelfManagedEventSource": {
            "Endpoints": {
               "string" : [ "string" ]
            }
         },
         "SelfManagedKafkaEventSourceConfig": {
            "ConsumerGroupId": "string"
         },
         "SourceAccessConfigurations": [
            {
               "Type": "string",
               "URI": "string"
            }
         ],
         "StartingPosition": "string",
         "StartingPositionTimestamp": number,
         "State": "string",
         "StateTransitionReason": "string",
         "Topics": [ "string" ],
         "TumblingWindowInSeconds": number,
```

```
            "UUID": "string"
        }
    ],
    "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

EventSourceMappings

事件源映射的列表。

类型：EventSourceMappingConfiguration 对象数组

NextMarker

当响应不包含所有事件源映射时返回的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListFunctionEventInvokeConfigs

检索函数异步调用的配置列表。

要配置异步调用的选项，请使用 PutFunctionEventInvokeConfig。

请求语法

```
GET /2019-09-25/functions/FunctionName/event-invoke-config/list?
Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `my-function`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN - `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

要返回的配置的最大数量。

有效范围：最小值为 1。最大值为 50。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "FunctionEventInvokeConfigs": [
      {
         "DestinationConfig": {
            "OnFailure": {
               "Destination": "string"
            },
            "OnSuccess": {
               "Destination": "string"
            }
         },
         "FunctionArn": "string",
         "LastModified": number,
         "MaximumEventAgeInSeconds": number,
         "MaximumRetryAttempts": number
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

FunctionEventInvokeConfigs

配置的列表。

类型：FunctionEventInvokeConfig 对象数组

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK

- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListFunctions

返回 Lambda 函数列表，以及每个函数的版本特定配置。Lambda 每次调用最多返回 50 个函数。

将 `FunctionVersion` 设置为 `ALL`，以包括每个函数的所有已发布版本以及未发布的版本。

> **Note**
>
> `ListFunctions` 操作将返回 FunctionConfiguration 字段的子集。要获取函数或版本的附加字段
> （State、StateReasonCode、StateReason、LastUpdateStatus、LastUpdateStatusReason、LastUpda
> 请使用 GetFunction。

请求语法

```
GET /2015-03-31/functions/?
FunctionVersion=FunctionVersion&Marker=Marker&MasterRegion=MasterRegion&MaxItems=MaxItems
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionVersion

设置为 `ALL`，以包含每个函数的所有已发布版本的条目。

有效值：`ALL`

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MasterRegion

对于 Lambda@Edge 函数，为主函数的 AWS 区域。例如，`us-east-1` 筛选函数列表，以仅包括从美国东部（弗吉尼亚州北部）的主函数复制的 Lambda@Edge 函数。如果已指定，则必须将 `FunctionVersion` 设置为 `ALL`。

模式：`ALL|[a-z]{2}(-gov)?-[a-z]+-\d{1}`

MaxItems

要在响应中返回的函数的最大数量。请注意，即使您设置的数字较高，ListFunctions 在每个响应中最多返回 50 个项目。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Functions": [
      {
         "Architectures": [ "string" ],
         "CodeSha256": "string",
         "CodeSize": number,
         "DeadLetterConfig": {
            "TargetArn": "string"
         },
         "Description": "string",
         "Environment": {
            "Error": {
               "ErrorCode": "string",
               "Message": "string"
            },
            "Variables": {
               "string" : "string"
            }
         },
         "EphemeralStorage": {
            "Size": number
         },
         "FileSystemConfigs": [
            {
               "Arn": "string",
               "LocalMountPath": "string"
```

```
      }
   ],
   "FunctionArn": "string",
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
```

```
                "Message": "string"
            },
            "RuntimeVersionArn": "string"
         },
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string",
         "SnapStart": {
            "ApplyOn": "string",
            "OptimizationStatus": "string"
         },
         "State": "string",
         "StateReason": "string",
         "StateReasonCode": "string",
         "Timeout": number,
         "TracingConfig": {
            "Mode": "string"
         },
         "Version": "string",
         "VpcConfig": {
            "Ipv6AllowedForDualStack": boolean,
            "SecurityGroupIds": [ "string" ],
            "SubnetIds": [ "string" ],
            "VpcId": "string"
         }
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Functions

Lambda 函数的列表。

类型：FunctionConfiguration 对象数组

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListFunctionsByCodeSigningConfig

列出使用指定代码签名配置的函数。您可以在删除代码签名配置之前使用此方法，以验证没有函数正在使用该配置。

请求语法

```
GET /2020-04-22/code-signing-configs/CodeSigningConfigArn/functions?
Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

要返回的最大项目数量。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "FunctionArns": [ "string" ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

FunctionArns

函数 ARN。

类型：字符串数组

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListFunctionUrlConfigs

返回指定函数的 Lambda 函数 URL 的列表。

请求语法

```
GET /2021-10-31/functions/FunctionName/urls?Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

要在响应中返回的函数 URL 的最大数量。请注意，即使您设置的数字较高，`ListFunctionUrlConfigs` 在每个响应中最多返回 50 个项目。

有效范围：最小值为 1。最大值为 50。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "FunctionUrlConfigs": [
      {
         "AuthType": "string",
         "Cors": {
            "AllowCredentials": boolean,
            "AllowHeaders": [ "string" ],
            "AllowMethods": [ "string" ],
            "AllowOrigins": [ "string" ],
            "ExposeHeaders": [ "string" ],
            "MaxAge": number
         },
         "CreationTime": "string",
         "FunctionArn": "string",
         "FunctionUrl": "string",
         "InvokeMode": "string",
         "LastModifiedTime": "string"
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

FunctionUrlConfigs

函数 URL 配置的列表。

类型：FunctionUrlConfig 对象数组

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK

- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListLayers

列出 AWS Lambda 层并显示有关每个函数层的最新版本的信息。指定运行时标识符，以仅列出指示它们与该运行时兼容的层。指定兼容架构以仅包含与该指令集架构兼容的层。

请求语法

```
GET /2018-10-31/layers?
CompatibleArchitecture=CompatibleArchitecture&CompatibleRuntime=CompatibleRuntime&Marker=Marker
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

CompatibleArchitecture

兼容指令集架构。

有效值：x86_64 | arm64

CompatibleRuntime

运行时标识符。例如，java21。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

Marker

由上一次调用返回的分页令牌。

MaxItems

要返回的层最大数量。

有效范围：最小值为 1。最大值为 50。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Layers": [
      {
         "LatestMatchingVersion": {
            "CompatibleArchitectures": [ "string" ],
            "CompatibleRuntimes": [ "string" ],
            "CreatedDate": "string",
            "Description": "string",
            "LayerVersionArn": "string",
            "LicenseInfo": "string",
            "Version": number
         },
         "LayerArn": "string",
         "LayerName": "string"
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Layers

函数层的列表。

类型：LayersListItem 对象数组

NextMarker

当响应不包含所有层时返回的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK

- 适用于 Ruby V3 的 AWS SDK

ListLayerVersions

列出某个 AWS Lambda 层的版本。未列出已删除的版本。指定运行时标识符，以仅列出指示它们与该运行时兼容的版本。指定兼容架构以仅包含与该架构兼容的层版本。

请求语法

```
GET /2018-10-31/layers/LayerName/versions?
CompatibleArchitecture=CompatibleArchitecture&CompatibleRuntime=CompatibleRuntime&Marker=Marker
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

CompatibleArchitecture

兼容指令集架构。

有效值：x86_64 | arm64

CompatibleRuntime

运行时标识符。例如，java21。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+

必需：是

Marker

由上一次调用返回的分页令牌。

MaxItems

要返回的版本最大数量。

有效范围：最小值为 1。最大值为 50。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "LayerVersions": [
      {
         "CompatibleArchitectures": [ "string" ],
         "CompatibleRuntimes": [ "string" ],
         "CreatedDate": "string",
         "Description": "string",
         "LayerVersionArn": "string",
         "LicenseInfo": "string",
         "Version": number
      }
   ],
   "NextMarker": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

LayerVersions

版本列表。

类型：LayerVersionsListItem 对象数组

NextMarker

当响应不包含所有版本时返回的分页令牌。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListProvisionedConcurrencyConfigs

检索函数的预置并发配置列表。

请求语法

```
GET /2019-09-30/functions/FunctionName/provisioned-concurrency?
List=ALL&Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

指定一个数字以限制返回的配置数。

有效范围：最小值为 1。最大值为 50。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "NextMarker": "string",
   "ProvisionedConcurrencyConfigs": [
      {
         "AllocatedProvisionedConcurrentExecutions": number,
         "AvailableProvisionedConcurrentExecutions": number,
         "FunctionArn": "string",
         "LastModified": "string",
         "RequestedProvisionedConcurrentExecutions": number,
         "Status": "string",
         "StatusReason": "string"
      }
   ]
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

ProvisionedConcurrencyConfigs

预置的并发配置列表。

类型：ProvisionedConcurrencyConfigListItem 对象数组

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListTags

返回函数的标签。您也可以将标签与 GetFunction 一起使用。

请求语法

```
GET /2017-03-31/tags/ARN HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

ARN

函数的 Amazon Resource Name (ARN)。注意：Lambda 不支持向别名或版本添加标签。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Tags": {
      "string" : "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Tags

函数的标签。

类型：字符串到字符串映射

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ListVersionsByFunction

返回版本列表，以及每个版本的版本特定配置。Lambda 每次调用最多返回 50 个版本。

请求语法

```
GET /2015-03-31/functions/FunctionName/versions?Marker=Marker&MaxItems=MaxItems
 HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Marker

指定上次请求返回的分页令牌以检索下一页结果。

MaxItems

要返回的版本最大数量。请注意，即使您设置的数字较高，`ListVersionsByFunction` 在每个响应中最多返回 50 个项目。

有效范围：最小值为 1。最大值为 10000。

请求正文

该请求没有请求正文。

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "NextMarker": "string",
   "Versions": [
      {
         "Architectures": [ "string" ],
         "CodeSha256": "string",
         "CodeSize": number,
         "DeadLetterConfig": {
            "TargetArn": "string"
         },
         "Description": "string",
         "Environment": {
            "Error": {
               "ErrorCode": "string",
               "Message": "string"
            },
            "Variables": {
               "string" : "string"
            }
         },
         "EphemeralStorage": {
            "Size": number
         },
         "FileSystemConfigs": [
            {
               "Arn": "string",
               "LocalMountPath": "string"
            }
         ],
         "FunctionArn": "string",
         "FunctionName": "string",
         "Handler": "string",
         "ImageConfigResponse": {
            "Error": {
```

```
            "ErrorCode": "string",
            "Message": "string"
         },
         "ImageConfig": {
            "Command": [ "string" ],
            "EntryPoint": [ "string" ],
            "WorkingDirectory": "string"
         }
      },
      "KMSKeyArn": "string",
      "LastModified": "string",
      "LastUpdateStatus": "string",
      "LastUpdateStatusReason": "string",
      "LastUpdateStatusReasonCode": "string",
      "Layers": [
         {
            "Arn": "string",
            "CodeSize": number,
            "SigningJobArn": "string",
            "SigningProfileVersionArn": "string"
         }
      ],
      "LoggingConfig": {
         "ApplicationLogLevel": "string",
         "LogFormat": "string",
         "LogGroup": "string",
         "SystemLogLevel": "string"
      },
      "MasterArn": "string",
      "MemorySize": number,
      "PackageType": "string",
      "RevisionId": "string",
      "Role": "string",
      "Runtime": "string",
      "RuntimeVersionConfig": {
         "Error": {
            "ErrorCode": "string",
            "Message": "string"
         },
         "RuntimeVersionArn": "string"
      },
      "SigningJobArn": "string",
      "SigningProfileVersionArn": "string",
      "SnapStart": {
```

```
            "ApplyOn": "string",
            "OptimizationStatus": "string"
         },
         "State": "string",
         "StateReason": "string",
         "StateReasonCode": "string",
         "Timeout": number,
         "TracingConfig": {
            "Mode": "string"
         },
         "Version": "string",
         "VpcConfig": {
            "Ipv6AllowedForDualStack": boolean,
            "SecurityGroupIds": [ "string" ],
            "SubnetIds": [ "string" ],
            "VpcId": "string"
         }
      }
   ]
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

NextMarker

有更多结果可用时包含的分页令牌。

类型：字符串

Versions

Lambda 函数版本的列表。

类型：FunctionConfiguration 对象数组

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PublishLayerVersion

从 ZIP 归档中创建某个 AWS Lambda 层。每次使用相同的层名称调用 PublishLayerVersion 时，将会创建新的版本。

使用 CreateFunction 或 UpdateFunctionConfiguration 将层添加到您的函数。

请求语法

```
POST /2018-10-31/layers/LayerName/versions HTTP/1.1
Content-type: application/json

{
   "CompatibleArchitectures": [ "string" ],
   "CompatibleRuntimes": [ "string" ],
   "Content": {
      "S3Bucket": "string",
      "S3Key": "string",
      "S3ObjectVersion": "string",
      "ZipFile": blob
   },
   "Description": "string",
   "LicenseInfo": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

CompatibleArchitectures

兼容指令集架构列表。

类型：字符串数组

数组成员：最多 15 项。

有效值：`x86_64 | arm64`

必需：否

CompatibleRuntimes

兼容的函数运行时列表。用于使用 ListLayers 和 ListLayerVersions 进行筛选。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串数组

数组成员：最多 15 项。

有效值：`nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21`

必需：否

Content

函数层存档。

类型：LayerVersionContentInput 对象

必需：是

Description

关于版本的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

LicenseInfo

层的软件许可证。它可以是以下任一种：

- SPDX 许可证标识符。例如，MIT。
- 在 Internet 上托管的许可证的 URL。例如，https://opensource.org/licenses/MIT。
- 许可证全文。

类型：字符串

长度约束：最大长度为 512。

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "CompatibleArchitectures": [ "string" ],
   "CompatibleRuntimes": [ "string" ],
   "Content": {
      "CodeSha256": "string",
      "CodeSize": number,
      "Location": "string",
      "SigningJobArn": "string",
      "SigningProfileVersionArn": "string"
   },
   "CreatedDate": "string",
   "Description": "string",
   "LayerArn": "string",
   "LayerVersionArn": "string",
```

```
    "LicenseInfo": "string",
    "Version": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

CompatibleArchitectures

兼容指令集架构列表。

类型：字符串数组

数组成员：最多 15 项。

有效值：x86_64 | arm64

CompatibleRuntimes

层的兼容运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串数组

数组成员：最多 15 项。

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

Content

有关层版本的详细信息。

类型：LayerVersionContentOutput 对象

CreatedDate

层版本的创建日期，为 ISO-8601 格式 (YYYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

Description

关于版本的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

LayerArn

层的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+`

LayerVersionArn

层版本的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

LicenseInfo

层的软件许可证。

类型：字符串

长度约束：最大长度为 512。

Version

版本号。

类型：长整型

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeStorageExceededException

您的 AWS 账户 已超出最大总代码大小。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PublishVersion

从一个函数的当前代码和配置创建一个版本。使用版本创建不会更改的函数代码和配置的快照。

如果该函数的配置和代码自上一个版本以来没有更改，则 AWS Lambda 不会发布版本。使用 UpdateFunctionCode 或 UpdateFunctionConfiguration 在发布版本之前更新函数。

客户端可以直接调用版本或使用别名调用版本。要创建别名，请使用 CreateAlias。

请求语法

```
POST /2015-03-31/functions/FunctionName/versions HTTP/1.1
Content-type: application/json

{
   "CodeSha256": "string",
   "Description": "string",
   "RevisionId": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

CodeSha256

仅当哈希值与指定的值匹配时才发布版本。如果自上次更新后函数代码发生更改，请使用此选项以避免发布版本。您可以从 UpdateFunctionCode 的输出获得您上载的版本的哈希值。

类型：字符串

必需：否

Description

用于覆盖函数配置中的描述的版本描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新函数。如果自上次更新后函数配置发生了更改，请使用此选项以避免发布版本。

类型：字符串

必需：否

响应语法

```
HTTP/1.1 201
Content-type: application/json

{
   "Architectures": [ "string" ],
   "CodeSha256": "string",
   "CodeSize": number,
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
```

```
   "Description": "string",
   "Environment": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionArn": "string",
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
```

```
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "RuntimeVersionArn": "string"
   },
   "SigningJobArn": "string",
   "SigningProfileVersionArn": "string",
   "SnapStart": {
      "ApplyOn": "string",
      "OptimizationStatus": "string"
   },
   "State": "string",
   "StateReason": "string",
   "StateReasonCode": "string",
   "Timeout": number,
   "TracingConfig": {
      "Mode": "string"
   },
   "Version": "string",
   "VpcConfig": {
      "Ipv6AllowedForDualStack": boolean,
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcId": "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 201 响应。

服务以 JSON 格式返回的以下数据。

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：x86_64 | arm64

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 `Successful`。

类型：字符串

有效值：`Successful | Failed | InProgress`

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：`EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied |`

KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Layers

函数的层。

类型：Layer 对象数组

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

PackageType

部署程序包的类型。将容器映像设置为 Image，然后将 .zip 文件存档设置为 Zip。

类型：字符串

有效值：Zip | Image

RevisionId

函数或别名的最新更新版本。

类型：字符串

Role

函数的执行角色。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

RuntimeVersionConfig

运行时的 ARN 和发生的任何错误。

类型：RuntimeVersionConfig 对象

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

SnapStart

发布函数版本时，将 ApplyOn 设置为 PublishedVersions 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

State

函数的当前状态。当状态为 Inactive，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：Pending | Active | Inactive | Failed

StateReason

函数当前状态的原因。

类型：字符串

StateReasonCode

函数当前状态的原因代码。当代码为 Creating 时，无法调用或修改该函数。

类型：字符串

有效值：Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeStorageExceededException

您的 AWS 账户 已超出最大总代码大小。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PutFunctionCodeSigningConfig

更新函数的代码签名配置。对代码签名配置的更改将在用户下次尝试将代码包部署到函数时生效。

请求语法

```
PUT /2020-06-30/functions/FunctionName/code-signing-config HTTP/1.1
Content-type: application/json

{
   "CodeSigningConfigArn": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - MyFunction。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:MyFunction。
- 部分 ARN - 123456789012:function:MyFunction。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CodeSigningConfigArn": "string",
   "FunctionName": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeSigningConfigNotFoundException

指定的代码签名配置不存在。

HTTP 状态代码：404

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PutFunctionConcurrency

设置函数的最大同时执行数，并为该并发级别预留容量。

并发设置适用于整个函数，包括所有已发布版本和未发布版本。预留并发可确保您的函数具有同时处理指定数量事件的容量，并防止其扩展到超出该级别。使用 GetFunction 查看函数的当前设置。

使用 GetAccountSettings 以查看您的区域并发限制。您可以根据需要为任意数量的函数预留并发，只需为未配置每函数限制的函数预留至少 100 个并发执行。有关更多信息，请参阅 Lambda 函数扩展。

请求语法

```
PUT /2017-10-31/functions/FunctionName/concurrency HTTP/1.1
Content-type: application/json

{
   "ReservedConcurrentExecutions": number
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

ReservedConcurrentExecutions

为函数保留的同时执行次数。

类型：整数

有效范围：最小值为 0。

必需：是

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "ReservedConcurrentExecutions": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

ReservedConcurrentExecutions

为此函数预留的并发执行数。有关更多信息，请参阅管理 Lambda 预留并发。

类型：整数

有效范围：最小值为 0。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK

- 适用于 Ruby V3 的 AWS SDK

PutFunctionEventInvokeConfig

配置对函数、版本或别名的异步调用选项。如果函数、版本或别名的配置已存在，则此操作会覆盖该配置。如果排除任何设置，则会删除这些设置。要设置一个选项而不影响其他选项的现有设置，请使用 UpdateFunctionEventInvokeConfig。

默认情况下，如果函数返回错误，Lambda 将重试两次异步调用。它将事件在队列中保留最多 6 个小时。如果事件未完成所有处理尝试，或者在异步调用队列中保留太长的时间，Lambda 将丢弃该事件。要保留已丢弃的事件，请使用 UpdateFunctionConfiguration 配置死信队列。

要将调用记录发送到队列、主题、函数或事件总线，请指定目的地。您可以为成功调用（成功时）和所有处理尝试均失败（失败时）的事件配置单独的目的地。除死信队列之外，或者作为替代死信队列的方式，您还可以配置目的地。

请求语法

```
PUT /2019-09-25/functions/FunctionName/event-invoke-config?Qualifier=Qualifier HTTP/1.1
Content-type: application/json

{
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "MaximumEventAgeInSeconds": number,
   "MaximumRetryAttempts": number
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – my-function（仅限名称）、my-function:v1（具有别名）。
- 函数 ARN - arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN - 123456789012:function:my-function。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

请求正文

请求接受采用 JSON 格式的以下数据。

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。
- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

必需：否

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

必需：否

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "FunctionArn": "string",
   "LastModified": number,
   "MaximumEventAgeInSeconds": number,
   "MaximumRetryAttempts": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。
- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

LastModified

上次更新配置的日期和时间（以 Unix 时间秒为单位）。

类型：时间戳

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK

- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PutProvisionedConcurrencyConfig

将预置的并发配置添加到函数的别名或版本。

请求语法

```
PUT /2019-09-30/functions/FunctionName/provisioned-concurrency?Qualifier=Qualifier
 HTTP/1.1
Content-type: application/json

{
   "ProvisionedConcurrentExecutions": number
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

ProvisionedConcurrentExecutions

要为版本或别名分配的预置并发数量。

类型：整数

有效范围：最小值为 1。

必需：是

响应语法

```
HTTP/1.1 202
Content-type: application/json

{
   "AllocatedProvisionedConcurrentExecutions": number,
   "AvailableProvisionedConcurrentExecutions": number,
   "LastModified": "string",
   "RequestedProvisionedConcurrentExecutions": number,
   "Status": "string",
   "StatusReason": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 202 响应。

服务以 JSON 格式返回以下数据。

AllocatedProvisionedConcurrentExecutions

输入要分配的预置并发数量。在线性部署和金丝雀部署期间使用加权别名时，该值会根据为函数版本预配置的并发量而波动。

类型：整数

有效范围：最小值为 0。

AvailableProvisionedConcurrentExecutions

可用的预置并发数量。

类型：整数

有效范围：最小值为 0。

LastModified

用户上次更新配置的日期和时间，格式为 ISO 8601。

类型：字符串

RequestedProvisionedConcurrentExecutions

请求的预置并发数量。

类型：整数

有效范围：最小值为 1。

Status

分配过程的状态。

类型：字符串

有效值：`IN_PROGRESS | READY | FAILED`

StatusReason

分配失败的原因，即无法分配预置并发的原因。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

PutRuntimeManagementConfig

设置函数版本的运行时管理配置。有关更多信息，请参阅 Runtime updates（运行时更新）。

请求语法

```
PUT /2021-07-20/functions/FunctionName/runtime-management-config?Qualifier=Qualifier
 HTTP/1.1
Content-type: application/json

{
   "RuntimeVersionArn": "string",
   "UpdateRuntimeOn": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – my-function。
- 函数 ARN – arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN – 123456789012:function:my-function。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

指定该函数的版本。可以是 $LATEST 或已发布的版本号。如果未指定任何值，则将返回 $LATEST 版本的配置。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

请求接受采用 JSON 格式的以下数据。

RuntimeVersionArn

您希望函数使用的运行时版本的 ARN。

> **Note**
>
> 仅在使用手动运行时更新模式时需要。

类型：字符串

长度限制：最小长度为 26。最大长度为 2048。

模式：`^arn:(aws[a-zA-Z-]*):lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}::runtime:.+$`

必需：否

UpdateRuntimeOn

指定运行时更新模式。

- 自动（默认）– 通过两阶段运行时版本推出，自动更新到最新的安全运行时版本。对于大多数客户来说，这是确保其始终从运行时更新中受益的最佳选择。
- 函数更新 – 当您更新函数时，Lambda 会将函数的运行时更新为最新的安全运行时版本。这种方法可将运行时更新与函数部署同步，这样，您就可以控制应用运行时更新的时间，并及早检测和缓解少数运行时更新不兼容问题。使用此设置时，您需要定期更新函数以保持最新的函数运行时。
- 手动 – 在函数配置中指定运行时版本。该函数将无限期使用此运行时版本。在极少数情况下，新的运行时版本与现有函数不兼容，这样您就可以将函数回滚到早期运行时版本。有关更多信息，请参阅 Roll back a runtime version（回滚运行时版本）。

类型：字符串

有效值：`Auto | Manual | FunctionUpdate`

必需：是

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "FunctionArn": "string",
   "RuntimeVersionArn": "string",
   "UpdateRuntimeOn": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

FunctionArn

函数的 ARN

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

RuntimeVersionArn

配置函数所使用的运行时的 ARN。如果运行时更新模式为手动，则将返回 ARN，否则将返回 `null`。

类型：字符串

长度限制：最小长度为 26。最大长度为 2048。

模式：`^arn:(aws[a-zA-Z-]*):lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}::runtime:.+$`

UpdateRuntimeOn

运行时更新模式。

类型：字符串

有效值：`Auto | Manual | FunctionUpdate`

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

RemoveLayerVersionPermission

从某个 AWS Lambda 层的某个版本的权限策略中删除语句。有关更多信息，请参阅AddLayerVersionPermission。

请求语法

```
DELETE /2018-10-31/layers/LayerName/versions/VersionNumber/policy/StatementId?
RevisionId=RevisionId HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

LayerName

层的名称或 Amazon Resource Name (ARN)。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：是

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新策略。使用此选项以避免修改自上次读取后已发生更改的策略。

StatementId

添加语句时指定的标识符。

长度限制：最小长度为 1。最大长度为 100。

模式：`([a-zA-Z0-9-_]+)`

必需：是

VersionNumber

版本号。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

RemovePermission

从 AWS 服务 或其他 AWS 账户 中撤销函数使用权限。从 GetPolicy 的输出中获取语句的 ID。

请求语法

```
DELETE /2015-03-31/functions/FunctionName/policy/StatementId?
Qualifier=Qualifier&RevisionId=RevisionId HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – my-function（仅限名称）、my-function:v1（具有别名）。
- 函数 ARN – arn:aws:lambda:us-west-2:123456789012:function:my-function。
- 部分 ARN – 123456789012:function:my-function。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

Qualifier

指定版本或别名以从函数的已发布版本中删除权限。

长度限制：最小长度为 1。长度上限为 128。

模式：(|[a-zA-Z0-9$_-]+)

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新策略。使用此选项以避免修改自上次读取后已发生更改的策略。

StatementId

要删除的权限的语句 ID。

长度限制：最小长度为 1。最大长度为 100。

模式：`([a-zA-Z0-9-_.]+)`

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

TagResource

将标签添加到函数。

请求语法

```
POST /2017-03-31/tags/ARN HTTP/1.1
Content-type: application/json

{
   "Tags": {
      "string" : "string"
   }
}
```

URI 请求参数

请求使用以下 URI 参数。

ARN

函数的 Amazon Resource Name (ARN)。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

Tags

要应用于函数的标签列表。

类型：字符串到字符串映射

必需：是

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UntagResource

从函数中删除标签。

请求语法

```
DELETE /2017-03-31/tags/ARN?tagKeys=TagKeys HTTP/1.1
```

URI 请求参数

请求使用以下 URI 参数。

ARN

函数的 Amazon Resource Name (ARN)。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

TagKeys

要从函数中删除的标签键的列表。

必需：是

请求体

该请求没有请求正文。

响应语法

```
HTTP/1.1 204
```

响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 204 响应。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK

- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateAlias

更新 Lambda 函数别名的配置。

请求语法

```
PUT /2015-03-31/functions/FunctionName/aliases/Name HTTP/1.1
Content-type: application/json

{
   "Description": "string",
   "FunctionVersion": "string",
   "RevisionId": "string",
   "RoutingConfig": {
      "AdditionalVersionWeights": {
         "string" : number
      }
   }
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 - `MyFunction`。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。
- 部分 ARN - `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Name

别名的名称。

长度限制：最小长度为 1。长度上限为 128。

模式：(?!^[0-9]+$)([a-zA-Z0-9-_]+)

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

必需：否

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新别名。使用此选项以避免修改自上次读取后已发生更改的别名。

类型：字符串

必需：否

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AliasArn": "string",
   "Description": "string",
   "FunctionVersion": "string",
   "Name": "string",
   "RevisionId": "string",
   "RoutingConfig": {
      "AdditionalVersionWeights": {
         "string" : number
      }
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AliasArn

是别名的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

Name

别名的名称。

类型：字符串

长度限制：长度下限为 1。长度上限为 128。

模式：(?!^[0-9]+$)([a-zA-Z0-9-_]+)

RevisionId

更新别名时更改的唯一标识符。

类型：字符串

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3

- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateCodeSigningConfig

更新代码签名配置。对代码签名配置的更改将在用户下次尝试将代码包部署到函数时生效。

请求语法

```
PUT /2020-04-22/code-signing-configs/CodeSigningConfigArn HTTP/1.1
Content-type: application/json

{
   "AllowedPublishers": {
      "SigningProfileVersionArns": [ "string" ]
   },
   "CodeSigningPolicies": {
      "UntrustedArtifactOnDeployment": "string"
   },
   "Description": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

AllowedPublishers

此代码签名配置的签名配置文件。

类型：AllowedPublishers 对象

必需：否

CodeSigningPolicies

代码签名策略。

类型：CodeSigningPolicies 对象

必需：否

Description

此代码签名配置的描述性名称。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "CodeSigningConfig": {
      "AllowedPublishers": {
         "SigningProfileVersionArns": [ "string" ]
      },
      "CodeSigningConfigArn": "string",
      "CodeSigningConfigId": "string",
      "CodeSigningPolicies": {
         "UntrustedArtifactOnDeployment": "string"
      },
      "Description": "string",
      "LastModified": "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

CodeSigningConfig

代码签名配置

类型：CodeSigningConfig 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateEventSourceMapping

更新事件源映射。您可以更改 AWS Lambda 调用的函数，或暂停调用并稍后从同一位置恢复。

有关如何配置不同事件源的详细信息，请参阅以下主题。

- Amazon DynamoDB Streams
- Amazon Kinesis
- Amazon SQS
- Amazon MQ 和 RabbitMQ
- Amazon MSK
- Apache Kafka
- Amazon DocumentDB

以下错误处理选项仅适用于流源（DynamoDB 和 Kinesis）：

- `BisectBatchOnFunctionError` – 如果函数返回错误，则将批次拆分为两批并重试。
- `DestinationConfig` – 将丢弃的记录发送到 Amazon SQS 队列或 Amazon SNS 主题。
- `MaximumRecordAgeInSeconds` – 丢弃超过指定期限的记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。
- `MaximumRetryAttempts` – 在指定重试次数后丢弃记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。
- `ParallelizationFactor` – 同时处理来自每个分片的多个批次。

有关适用于每个事件源的配置参数信息，请参阅以下主题。

- Amazon DynamoDB Streams
- Amazon Kinesis
- Amazon SQS
- Amazon MQ 和 RabbitMQ
- Amazon MSK
- Apache Kafka
- Amazon DocumentDB

请求语法

```
PUT /2015-03-31/event-source-mappings/UUID HTTP/1.1
Content-type: application/json

{
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "Enabled": boolean,
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
         }
      ]
   },
   "FunctionName": "string",
   "FunctionResponseTypes": [ "string" ],
   "MaximumBatchingWindowInSeconds": number,
   "MaximumRecordAgeInSeconds": number,
   "MaximumRetryAttempts": number,
   "ParallelizationFactor": number,
   "ScalingConfig": {
      "MaximumConcurrency": number
   },
   "SourceAccessConfigurations": [
      {
         "Type": "string",
         "URI": "string"
      }
```

```
    ],
    "TumblingWindowInSeconds": number
}
```

URI 请求参数

请求使用以下 URI 参数。

UUID

事件源映射的标识符。

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

- Amazon Kinesis – 默认值为 100。最大值为 10,000。
- Amazon DynamoDB Streams – 默认值为 100。最大值为 10,000。
- Amazon Simple Queue Service – 默认值为 10。对于标准队列，最大值为 10,000。对于 FIFO 队列，最大值为 10。
- Amazon Managed Streaming for Apache Kafka – 默认值为 100。最大值为 10,000。
- 自行管理的 Apache Kafka – 默认值为 100。最大值为 10,000。
- Amazon MQ（ActiveMQ 和 RabbitMQ）– 默认值为 100。最大值为 10,000。
- DocumentDB – 默认值 100。最大值为 10,000。

类型：整数

有效范围：最小值为 1。最大值为 10000。

必需：否

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。

类型：布尔值

必需：否

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Kafka）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

必需：否

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

必需：否

Enabled

当为 true 时，事件源映射处于活动状态。当为 false 时，Lambda 暂停轮询和调用。

默认值：True

类型：布尔值

必需：否

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

必需：否

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `MyFunction`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction`。

- 版本或别名 ARN – `arn:aws:lambda:us-west-2:123456789012:function:MyFunction:PROD`。
- 部分 ARN – `123456789012:function:MyFunction`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度将限制为 64 个字符。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

必需：否

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

必需：否

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。默认值为无限 (-1)。

类型：整数

有效范围：最小值为 -1。最大值为 604800。

必需：否

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。默认值为无限 (-1)。如果设置为无限 (-1)，则会一直重试失败的记录，直到记录过期。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

必需：否

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。

类型：整数

有效范围：最小值为 1。最大值为 10。

必需：否

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

必需：否

SourceAccessConfigurations

用于保护事件源的身份验证协议数组或 VPC 组件。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

必需：否

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

必需：否

响应语法

```
HTTP/1.1 202
Content-type: application/json

{
   "AmazonManagedKafkaEventSourceConfig": {
      "ConsumerGroupId": "string"
   },
   "BatchSize": number,
   "BisectBatchOnFunctionError": boolean,
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "DocumentDBEventSourceConfig": {
      "CollectionName": "string",
      "DatabaseName": "string",
      "FullDocument": "string"
   },
   "EventSourceArn": "string",
   "FilterCriteria": {
      "Filters": [
         {
            "Pattern": "string"
```

```
            }
         ]
      },
      "FunctionArn": "string",
      "FunctionResponseTypes": [ "string" ],
      "LastModified": number,
      "LastProcessingResult": "string",
      "MaximumBatchingWindowInSeconds": number,
      "MaximumRecordAgeInSeconds": number,
      "MaximumRetryAttempts": number,
      "ParallelizationFactor": number,
      "Queues": [ "string" ],
      "ScalingConfig": {
         "MaximumConcurrency": number
      },
      "SelfManagedEventSource": {
         "Endpoints": {
            "string" : [ "string" ]
         }
      },
      "SelfManagedKafkaEventSourceConfig": {
         "ConsumerGroupId": "string"
      },
      "SourceAccessConfigurations": [
         {
            "Type": "string",
            "URI": "string"
         }
      ],
      "StartingPosition": "string",
      "StartingPositionTimestamp": number,
      "State": "string",
      "StateTransitionReason": "string",
      "Topics": [ "string" ],
      "TumblingWindowInSeconds": number,
      "UUID": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 202 响应。

服务以 JSON 格式返回以下数据。

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

默认值：因服务而异。对于 Amazon SQS，默认值为 10。对于所有其他服务，默认值为 100。

相关设置：当将 `BatchSize` 设置为一个大于 10 的值时，必须至少将 `MaximumBatchingWindowInSeconds` 设置为 1。

类型：整数

有效范围：最小值为 1。最大值为 10000。

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。默认值为 False。

类型：布尔值

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Apache Kafka 事件源）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

FunctionArn

Lambda 函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：`ReportBatchItemFailures`

LastModified

上次更新事件源映射或其状态更改的日期（以 Unix 时间秒为单位）。

类型：时间戳

LastProcessingResult

函数的最后一次 Lambda 调用结果。

类型：字符串

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 `MaximumBatchingWindowInSeconds` 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 `MaximumBatchingWindowInSeconds`，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 `BatchSize` 设置为大于 10 的值时，必须将 `MaximumBatchingWindowInSeconds` 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。原定设置值为 -1，该值将最大年龄设置为无限。将该值设置为无限后，Lambda 永远不会删除旧记录。

> **Note**
>
> 最长记录期限的最小有效值为 60 秒。尽管小于 60 和大于 -1 的值都在参数的绝对范围内，但不允许使用它们

类型：整数

有效范围：最小值为 -1。最大值为 604800。

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。原定设置值为 -1，该值将最大重试次数设置为无限。将 MaximumRetryAttempts 设置为无限后，Lambda 会重试失败的记录，直到事件源中的记录过期为止。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。默认值是 1。

类型：整数

有效范围：最小值为 1。最大值为 10。

Queues

(Amazon MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：[\s\S]*

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

SourceAccessConfigurations

用于保护与定义事件源的身份验证协议数组 VPC 组件或虚拟化主机。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

State

事件源映射的状态。此状态可以是以下值之一：Creating、Enabling、Enabled、Disabling、Disabled、Updating 或 Deleting。

类型：字符串

StateTransitionReason

表明是用户还是 Lambda 对事件源映射进行了最后一次更改。

类型：字符串

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

UUID

事件源映射的标识符。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceInUseException

该操作与资源的可用性冲突。例如，您试图在 CREATING 状态中更新事件源映射，或尝试删除当前处于 UPDATING 状态的事件源映射。

HTTP 状态代码：400

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面

- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateFunctionCode

更新 Lambda 函数代码。如果为函数启用了代码签名，则代码包必须由受信任的发布者签名。有关更多信息，请参阅为 Lambda 配置代码签名。

如果函数的代码包类型为 `Image`，则您必须将 `ImageUri` 中的代码包指定为 Amazon ECR 注册表中容器映像的 URI。

如果函数的代码包类型为 `Zip`，则您必须将部署包指定为 .zip 文件归档。输入代码 .zip 文件位置的 Amazon S3 存储桶和密钥。此外，您还可以使用 `ZipFile` 字段提供函数代码内联。

部署包中的代码必须与函数的目标指令集架构兼容（`x86-64` 或 `arm64`）。

发布版本时，函数的代码将被锁定。您无法修改已发布版本的代码，只能修改未发布版本的代码。

> **Note**
>
> 对于定义为容器镜像的函数，Lambda 会将镜像标签解析为镜像摘要。在 Amazon ECR 中，如果您将镜像标签更新为新镜像，Lambda 不会自动更新该函数。

请求语法

```
PUT /2015-03-31/functions/FunctionName/code HTTP/1.1
Content-type: application/json

{
   "Architectures": [ "string" ],
   "DryRun": boolean,
   "ImageUri": "string",
   "Publish": boolean,
   "RevisionId": "string",
   "S3Bucket": "string",
   "S3Key": "string",
   "S3ObjectVersion": "string",
   "ZipFile": blob
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

Architectures

该函数支持的指令集架构。输入具有其中一个有效值（arm64 或 x86_64）的字符串数组。默认值为 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：`x86_64 | arm64`

必需：否

DryRun

设置为 true，以在不修改函数代码的情况下验证请求参数和访问权限。

类型：布尔值

必需：否

ImageUri

Amazon ECR 注册表中容器映像的 URI。请勿用于使用 .zip 文件归档定义的函数。

类型：字符串

必需：否

Publish

设置为 true，以在更新代码后发布新版本的函数。这与单独调用 PublishVersion 的效果相同。

类型：布尔值

必需：否

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新函数。使用此选项以避免修改自上次读取后发生更改的函数。

类型：字符串

必需：否

S3Bucket

与您的函数处于同一 AWS 区域 的 Amazon S3 桶。存储桶可能位于不同的 AWS 账户 中。仅与使用 .zip 文件归档部署包定义的函数搭配使用。

类型：字符串

长度限制：最小长度为 3。长度上限为 63。

模式：`^[0-9A-Za-z\.\-_]*(?<!\.)$`

必需：否

S3Key

部署程序包的 Amazon S3 密钥。仅与使用 .zip 文件归档部署包定义的函数搭配使用。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

S3ObjectVersion

对于版本控制的对象，指要使用的部署程序包对象的版本。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

ZipFile

部署程序包的 base64 编码内容。AWS开发工具包和 AWS CLI 客户端为您处理编码。仅与使用 .zip 文件归档部署包定义的函数搭配使用。

类型：Base64 编码的二进制数据对象

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Architectures": [ "string" ],
   "CodeSha256": "string",
   "CodeSize": number,
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
```

```
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionArn": "string",
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
```

```
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "RuntimeVersionArn": "string"
   },
   "SigningJobArn": "string",
   "SigningProfileVersionArn": "string",
   "SnapStart": {
      "ApplyOn": "string",
      "OptimizationStatus": "string"
   },
   "State": "string",
   "StateReason": "string",
   "StateReasonCode": "string",
   "Timeout": number,
   "TracingConfig": {
      "Mode": "string"
   },
   "Version": "string",
   "VpcConfig": {
      "Ipv6AllowedForDualStack": boolean,
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcId": "string"
   }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：x86_64 | arm64

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 Successful。

类型：字符串

有效值：Successful | Failed | InProgress

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Layers

函数的层。

类型：Layer 对象数组

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

PackageType

部署程序包的类型。将容器映像设置为 `Image`，然后将 .zip 文件存档设置为 `Zip`。

类型：字符串

有效值：`Zip | Image`

RevisionId

函数或别名的最新更新版本。

类型：字符串

Role

函数的执行角色。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

RuntimeVersionConfig

运行时的 ARN 和发生的任何错误。

类型：RuntimeVersionConfig 对象

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

SnapStart

发布函数版本时，将 ApplyOn 设置为 PublishedVersions 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

State

函数的当前状态。当状态为 Inactive，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：Pending | Active | Inactive | Failed

StateReason

函数当前状态的原因。

类型：字符串

StateReasonCode

函数当前状态的原因代码。当代码为 Creating 时，无法调用或修改该函数。

类型：字符串

有效值：Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：(\$LATEST|[0-9]+)

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeSigningConfigNotFoundException

指定的代码签名配置不存在。

HTTP 状态代码：404

CodeStorageExceededException

您的 AWS 账户 已超出最大总代码大小。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：400

CodeVerificationFailedException

代码签名失败，一个或多个签名验证检查不匹配或过期，并且代码签名策略设置为“ENFORCE”。Lambda 阻止部署。

HTTP 状态代码：400

InvalidCodeSignatureException

代码签名未通过完整性检查。如果完整性检查失败，Lambda 会阻止部署，即使代码签名策略设置为 WARN 也是如此。

HTTP 状态代码：400

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateFunctionConfiguration

修改 Lambda 函数的版本特定设置。

更新函数时，Lambda 会预置该函数及其支持资源的实例。如果您的函数连接到 VPC，则此过程可能需要一分钟时间。在此期间，您无法修改函数，但仍可以调用函数。来自 GetFunctionConfiguration 响应的 `LastUpdateStatus`、`LastUpdateStatusReason` 和 `LastUpdateStatusReasonCode` 字段指示更新何时完成，且函数正在处理具有新配置的事件。有关更多信息，请参阅 Lambda 函数状态。

这些设置可能因函数的版本而异，并且在发布版本时会被锁定。您无法修改已发布版本的配置，只能修改未发布版本的配置。

要配置函数并发，请使用 PutFunctionConcurrency。要向 AWS 账户 或 AWS 服务 授予调用权限，请使用 AddPermission。

请求语法

```
PUT /2015-03-31/functions/FunctionName/configuration HTTP/1.1
Content-type: application/json

{
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "Handler": "string",
   "ImageConfig": {
```

```
      "Command": [ "string" ],
      "EntryPoint": [ "string" ],
      "WorkingDirectory": "string"
   },
   "KMSKeyArn": "string",
   "Layers": [ "string" ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MemorySize": number,
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "SnapStart": {
      "ApplyOn": "string"
   },
   "Timeout": number,
   "TracingConfig": {
      "Mode": "string"
   },
   "VpcConfig": {
      "Ipv6AllowedForDualStack": boolean,
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ]
   }
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

请求体

请求接受采用 JSON 格式的以下数据。

DeadLetterConfig

死信队列配置，用于指定 Lambda 在处理失败时发送异步事件的队列或主题。有关更多信息，请参阅死信队列。

类型：DeadLetterConfig 对象

必需：否

Description

该函数的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

Environment

执行期间可从函数代码访问的环境变量。

类型：Environment 对象

必需：否

EphemeralStorage

函数的 `/tmp` 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

必需：否

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

必需：否

Handler

代码中方法的名称，Lambda 可调用该方法来运行您的函数。如果部署包是 .zip 文件归档，则需要使用处理程序。该格式包括文件名，也可以包括命名空间和其他限定符，具体取决于运行时。有关更多信息，请参阅 Lambda 编程模型。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

必需：否

ImageConfig

覆盖容器镜像 Docker 文件中的值的容器镜像配置值。

类型：ImageConfig 对象

必需：否

KMSKeyArn

AWS Key Management Service (AWS KMS) 客户托管密钥的 ARN，用于加密函数的环境变量。激活 Lambda SnapStart 后，Lambda 还使用此密钥对函数的快照进行加密。如果您使用容器映像部署函数，Lambda 还会在部署函数后使用此密钥对其进行加密。请注意，此密钥与在 Amazon Elastic Container Registry（Amazon ECR）中用于保护容器映像的密钥不同。如果您未提供客户托管密钥，Lambda 将使用默认服务密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

必需：否

Layers

要添加到函数执行环境中的函数层列表。通过 ARN 指定每一层，包括版本。

类型：字符串数组

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

必需：否

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

必需：否

MemorySize

在运行时 可用于函数的内存 量。提高该函数的内存也会增加它的 CPU 分配。默认值为 128 MB。该值可以是 1 MB 的任意倍数。

类型：整数

有效范围：最小值为 128。最大值为 10240。

必需：否

RevisionId

仅当修订版 ID 与指定的 ID 匹配时才更新函数。使用此选项以避免修改自上次读取后发生更改的函数。

类型：字符串

必需：否

Role

函数的执行角色的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

必需：否

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：`nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21`

必需：否

SnapStart

函数的 SnapStart 设置。

类型：SnapStart 对象

必需：否

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。默认值为 3 秒。允许的最大值为 900 秒。有关更多信息，请参阅 Lambda 执行环境。

类型：整数

有效范围：最小值为 1。

必需：否

TracingConfig

将 Mode 设置为 `Active`，以使用 X-Ray 对传入请求的子集进行采样和跟踪。

类型：TracingConfig 对象

必需：否

VpcConfig

对于与 VPC 中的AWS资源的网络连接，指定 VPC 中的安全组和子网列表。当您将函数连接到 VPC 时，它只能通过该 VPC 访问资源和互联网。有关更多信息，请参阅配置 Lambda 函数以访问 VPC 中的资源。

类型：VpcConfig 对象

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "Architectures": [ "string" ],
   "CodeSha256": "string",
   "CodeSize": number,
   "DeadLetterConfig": {
      "TargetArn": "string"
   },
   "Description": "string",
   "Environment": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "Variables": {
         "string" : "string"
      }
   },
   "EphemeralStorage": {
      "Size": number
   },
   "FileSystemConfigs": [
      {
         "Arn": "string",
         "LocalMountPath": "string"
      }
   ],
   "FunctionArn": "string",
```

```
   "FunctionName": "string",
   "Handler": "string",
   "ImageConfigResponse": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "ImageConfig": {
         "Command": [ "string" ],
         "EntryPoint": [ "string" ],
         "WorkingDirectory": "string"
      }
   },
   "KMSKeyArn": "string",
   "LastModified": "string",
   "LastUpdateStatus": "string",
   "LastUpdateStatusReason": "string",
   "LastUpdateStatusReasonCode": "string",
   "Layers": [
      {
         "Arn": "string",
         "CodeSize": number,
         "SigningJobArn": "string",
         "SigningProfileVersionArn": "string"
      }
   ],
   "LoggingConfig": {
      "ApplicationLogLevel": "string",
      "LogFormat": "string",
      "LogGroup": "string",
      "SystemLogLevel": "string"
   },
   "MasterArn": "string",
   "MemorySize": number,
   "PackageType": "string",
   "RevisionId": "string",
   "Role": "string",
   "Runtime": "string",
   "RuntimeVersionConfig": {
      "Error": {
         "ErrorCode": "string",
         "Message": "string"
      },
      "RuntimeVersionArn": "string"
```

```
    },
    "SigningJobArn": "string",
    "SigningProfileVersionArn": "string",
    "SnapStart": {
       "ApplyOn": "string",
       "OptimizationStatus": "string"
    },
    "State": "string",
    "StateReason": "string",
    "StateReasonCode": "string",
    "Timeout": number,
    "TracingConfig": {
       "Mode": "string"
    },
    "Version": "string",
    "VpcConfig": {
       "Ipv6AllowedForDualStack": boolean,
       "SecurityGroupIds": [ "string" ],
       "SubnetIds": [ "string" ],
       "VpcId": "string"
    }
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：`x86_64 | arm64`

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 `Successful`。

类型：字符串

有效值：Successful | Failed | InProgress

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

Layers

函数的层。

类型：Layer 对象数组

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

PackageType

部署程序包的类型。将容器映像设置为 Image，然后将 .zip 文件存档设置为 Zip。

类型：字符串

有效值：Zip | Image

RevisionId

函数或别名的最新更新版本。

类型：字符串

Role

函数的执行角色。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

RuntimeVersionConfig

运行时的 ARN 和发生的任何错误。

类型：RuntimeVersionConfig 对象

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

SnapStart

发布函数版本时，将 `ApplyOn` 设置为 `PublishedVersions` 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

State

函数的当前状态。当状态为 `Inactive`，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：`Pending | Active | Inactive | Failed`

StateReason

函数当前状态的原因。

类型：字符串

StateReasonCode

函数当前状态的原因代码。当代码为 `Creating` 时，无法调用或修改该函数。

类型：字符串

有效值：`Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError`

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

CodeSigningConfigNotFoundException

指定的代码签名配置不存在。

HTTP 状态代码：404

CodeVerificationFailedException

代码签名失败，一个或多个签名验证检查不匹配或过期，并且代码签名策略设置为“ENFORCE”。Lambda 阻止部署。

HTTP 状态代码：400

InvalidCodeSignatureException

代码签名未通过完整性检查。如果完整性检查失败，Lambda 会阻止部署，即使代码签名策略设置为 WARN 也是如此。

HTTP 状态代码：400

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

PreconditionFailedException

提供的 RevisionId 与 Lambda 函数或别名的最新 RevisionId 不匹配。调用 `GetFunction` 或 `GetAlias` API 操作来检索您的资源的最新 RevisionId。

HTTP 状态代码：412

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateFunctionEventInvokeConfig

更新函数、版本或别名的异步调用配置。

要配置异步调用的选项，请使用 PutFunctionEventInvokeConfig。

请求语法

```
POST /2019-09-25/functions/FunctionName/event-invoke-config?Qualifier=Qualifier
 HTTP/1.1
Content-type: application/json

{
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "MaximumEventAgeInSeconds": number,
   "MaximumRetryAttempts": number
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称、版本或别名。

名称格式

- 函数名称 – `my-function`（仅限名称）、`my-function:v1`（具有别名）。
- 函数 ARN - `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN - `123456789012:function:my-function`。

您可以将一个版本号或别名附加到任何格式。长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

版本号或别名。

长度限制：最小长度为 1。长度上限为 128。

模式：`(|[a-zA-Z0-9$_-]+)`

请求正文

请求接受采用 JSON 格式的以下数据。

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。
- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

必需：否

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

必需：否

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "DestinationConfig": {
      "OnFailure": {
         "Destination": "string"
      },
      "OnSuccess": {
         "Destination": "string"
      }
   },
   "FunctionArn": "string",
   "LastModified": number,
   "MaximumEventAgeInSeconds": number,
   "MaximumRetryAttempts": number
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。

- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

LastModified

上次更新配置的日期和时间（以 Unix 时间秒为单位）。

类型：时间戳

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

UpdateFunctionUrlConfig

更新 Lambda 函数 URL 的配置。

请求语法

```
PUT /2021-10-31/functions/FunctionName/url?Qualifier=Qualifier HTTP/1.1
Content-type: application/json

{
   "AuthType": "string",
   "Cors": {
      "AllowCredentials": boolean,
      "AllowHeaders": [ "string" ],
      "AllowMethods": [ "string" ],
      "AllowOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAge": number
   },
   "InvokeMode": "string"
}
```

URI 请求参数

请求使用以下 URI 参数。

FunctionName

Lambda 函数的名称。

名称格式

- 函数名称 – `my-function`。
- 函数 ARN – `arn:aws:lambda:us-west-2:123456789012:function:my-function`。
- 部分 ARN – `123456789012:function:my-function`。

长度约束仅适用于完整 ARN。如果您仅指定函数名称，它的长度限制为 64 个字符。

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：是

Qualifier

别名。

长度限制：最小长度为 1。长度上限为 128。

模式：(^\$LATEST$)|((?!^[0-9]+$)([a-zA-Z0-9-_]+))

请求正文

请求接受采用 JSON 格式的以下数据。

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

必需：否

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

必需：否

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：BUFFERED | RESPONSE_STREAM

必需：否

响应语法

```
HTTP/1.1 200
Content-type: application/json

{
   "AuthType": "string",
   "Cors": {
      "AllowCredentials": boolean,
      "AllowHeaders": [ "string" ],
      "AllowMethods": [ "string" ],
      "AllowOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAge": number
   },
   "CreationTime": "string",
   "FunctionArn": "string",
   "FunctionUrl": "string",
   "InvokeMode": "string",
   "LastModifiedTime": "string"
}
```

响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

CreationTime

创建函数 URL 时，使用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?

FunctionUrl

函数的 HTTP URL 端点。

类型：字符串

长度限制：最小长度为 40。最大长度为 100。

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：BUFFERED | RESPONSE_STREAM

LastModifiedTime

上次更新函数 URL 时，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

错误

有关所有操作返回的常见错误的信息，请参阅 常见错误。

InvalidParameterValueException

请求中的参数之一无效。

HTTP 状态代码：400

ResourceConflictException

资源已存在，或者其他操作正在进行中。

HTTP 状态代码：409

ResourceNotFoundException

请求中指定的资源不存在。

HTTP 状态代码：404

ServiceException

AWS Lambda 服务遇到了内部错误。

HTTP 状态代码：500

TooManyRequestsException

超出了请求吞吐量限制。有关更多信息，请参阅 Lambda 限额。

HTTP 状态代码：429

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- AWS 命令行界面
- 适用于 .NET 的 AWS SDK
- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- AWS 适用于 JavaScript 的开发工具包 V3
- 适用于 PHP V3 的 AWS SDK
- 适用于 Python 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

数据类型

AWS Lambda API 包含各种操作使用的多种数据类型。本节详细描述每种数据类型。

> **Note**
>
> 不能保证数据类型结构中每个元素的顺序。应用程序不应假设特定的顺序。

支持以下数据类型：

- AccountLimit
- AccountUsage
- AliasConfiguration
- AliasRoutingConfiguration
- AllowedPublishers
- AmazonManagedKafkaEventSourceConfig
- CodeSigningConfig
- CodeSigningPolicies
- Concurrency
- Cors
- DeadLetterConfig
- DestinationConfig
- DocumentDBEventSourceConfig
- Environment
- EnvironmentError
- EnvironmentResponse
- EphemeralStorage
- EventSourceMappingConfiguration
- FileSystemConfig
- Filter
- FilterCriteria

- FunctionCode
- FunctionCodeLocation
- FunctionConfiguration
- FunctionEventInvokeConfig
- FunctionUrlConfig
- ImageConfig
- ImageConfigError
- ImageConfigResponse
- InvokeResponseStreamUpdate
- InvokeWithResponseStreamCompleteEvent
- InvokeWithResponseStreamResponseEvent
- Layer
- LayersListItem
- LayerVersionContentInput
- LayerVersionContentOutput
- LayerVersionsListItem
- LoggingConfig
- OnFailure
- OnSuccess
- ProvisionedConcurrencyConfigListItem
- RuntimeVersionConfig
- RuntimeVersionError
- ScalingConfig
- SelfManagedEventSource
- SelfManagedKafkaEventSourceConfig
- SnapStart
- SnapStartResponse
- SourceAccessConfiguration
- TracingConfig
- TracingConfigResponse

- VpcConfig
- VpcConfigResponse

AccountLimit

与并发和存储相关的限制。所有文件和存储大小均以字节为单位。

内容

CodeSizeUnzipped

提取函数部署包和图层时的最大值。

类型：长整型

必需：否

CodeSizeZipped

部署包直接上载到 Lambda 时的最大值。对于较大的文件，请使用 Amazon S3。

类型：长整型

必需：否

ConcurrentExecutions

同时执行函数的最大数量。

类型：整数

必需：否

TotalCodeSize

可用于所有部署程序包和层归档的存储空间量。

类型：长整型

必需：否

UnreservedConcurrentExecutions

同时执行函数的最大数量减去为具有 PutFunctionConcurrency 的单个函数保留的容量。

类型：整数

有效范围：最小值为 0。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AccountUsage

正在使用的函数数量和存储容量。

内容

FunctionCount

Lambda 函数数量。

类型：长整型

必需：否

TotalCodeSize

部署包和层归档使用的存储空间量（以字节为单位）。

类型：长整型

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AliasConfiguration

提供有关 Lambda 函数别名的配置信息。

内容

AliasArn

是别名的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

Description

别名的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

FunctionVersion

别名调用的函数版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

必需：否

Name

别名的名称。

类型：字符串

长度限制：长度下限为 1。长度上限为 128。

模式：`(?!^[0-9]+$)([a-zA-Z0-9-_]+)`

必需：否

RevisionId

更新别名时更改的唯一标识符。

类型：字符串

必需：否

RoutingConfig

别名的路由配置。

类型：AliasRoutingConfiguration 对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AliasRoutingConfiguration

Lambda 函数别名的流量转移配置。

内容

AdditionalVersionWeights

第二个版本，以及路由到第二个版本的流量百分比。

类型：字符串到双映射

密钥长度限制：最小长度为 1。长度上限为 1024。

密钥模式：`[0-9]+`

有效范围：最小值为 0.0。最大值为 1.0。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AllowedPublishers

可对代码包进行签名的签名配置文件列表。

内容

SigningProfileVersionArns

每个签名配置文件的 Amazon 资源名称（ARN）。签名配置文件定义了可以对代码包进行签名的可信用户。

类型：字符串数组

数组成员：最少 1 项。最多 20 项。

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：是

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

内容

ConsumerGroupId

要加入的 Kafka 使用者组的标识符。在所有 Kafka 事件源中，使用者组 ID 必须是唯一的。在使用指定的使用者组 ID 创建 Kafka 事件源映射后，无法更新此值。有关更多信息，请参阅可自定义的使用者组 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 200。

模式：`[a-zA-Z0-9-\/*:_+=.@-]*`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CodeSigningConfig

有关 代码签名配置 的详细信息。

内容

AllowedPublishers

允许的发布者列表。

类型：AllowedPublishers 对象

必需：是

CodeSigningConfigArn

代码签名配置的 Amazon Resource Name (ARN)。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:code-signing-config:csc-[a-z0-9]{17}`

必需：是

CodeSigningConfigId

代码签名配置的唯一标识符。

类型：字符串

模式：`csc-[a-zA-Z0-9-_\.]{17}`

必需：是

CodeSigningPolicies

代码签名策略控制针对签名不匹配或过期的验证失败操作。

类型：CodeSigningPolicies 对象

必需：是

LastModified

上次修改代码签名配置的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

必需：是

Description

代码签名配置说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

CodeSigningPolicies

代码签名配置策略指定签名不匹配或过期的验证失败操作。

内容

UntrustedArtifactOnDeployment

部署验证失败的代码签名配置策略。如果将策略设置为 `Enforce`，则当签名验证检查失败时，Lambda 将阻止部署请求。如果将策略设置为 `Warn`，Lambda 将允许该部署并创建 CloudWatch 日志。

默认值：`Warn`

类型：字符串

有效值：`Warn | Enforce`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Concurrency

内容

ReservedConcurrentExecutions

为此函数预留的并发执行数。有关更多信息，请参阅管理 Lambda 预留并发。

类型：整数

有效范围：最小值为 0。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Cors

适用于 Lambda 函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。使用 CORS 授予从任何来源访问函数 URL 的权限。还可以使用 CORS 控制对函数 URL 请求中对特定 HTTP 标头和方法的访问。

内容

AllowCredentials

是否允许在函数 URL 的请求中使用 Cookie 或其他凭证。默认为 `false`。

类型：布尔值

必需：否

AllowHeaders

源可以将其包含在函数 URL 请求中的 HTTP 标头。例如：`Date`、`Keep-Alive`、`X-Custom-Header`。

类型：字符串数组

数组成员：最多 100 项。

长度约束：最大长度为 1024。

模式：`.*`

必需：否

AllowMethods

调用函数 URL 时允许的 HTTP 方法。例如：`GET`、`POST`、`DELETE`，或通配符 (*)。

类型：字符串数组

数组成员：最多 6 项。

长度约束：最大长度为 6。

模式：`.*`

必需：否

AllowOrigins

可以访问函数 URL 的来源。可以列出任意数量的特定来源，用逗号分隔。例如，`https://www.example.com`、`http://localhost:60905`。

或者，可以使用通配符 (*) 授予对所有源的访问权限。

类型：字符串数组

数组成员：最多 100 项。

长度限制：最小长度为 1。最大长度为 253。

模式：`.*`

必需：否

ExposeHeaders

函数响应中要向调用函数 URL 的源公开的 HTTP 标头。例如：`Date`、`Keep-Alive`、`X-Custom-Header`。

类型：字符串数组

数组成员：最多 100 项。

长度约束：最大长度为 1024。

模式：`.*`

必需：否

MaxAge

web 浏览器可以缓存预检请求结果的最长时间（秒）。原定设置下，该选项设置为 `0`，这意味着浏览器不会缓存结果。

类型：整数

有效范围：最小值为 0。最大值为 86400。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DeadLetterConfig

失败异步调用的死信队列。

内容

TargetArn

Amazon SQS 队列或 Amazon SNS 主题的 Amazon Resource Name (ARN)。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DestinationConfig

一个配置对象，用于在 Lambda 处理事件后指定事件目标。

内容

OnFailure

失败调用的目标配置。

类型：OnFailure 对象

必需：否

OnSuccess

成功调用的目标配置。

类型：OnSuccess 对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

内容

CollectionName

将使用的数据库中集合的名称。如果您未指定集合，Lambda 会使用所有集合。

类型：字符串

长度限制：最小长度为 1。最大长度为 57。

模式：`(^(?!(system\x2e)))(^[_a-zA-Z0-9])([^$]*)`

必需：否

DatabaseName

将使用的 DocumentDB 集群中数据库的名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 63。

模式：`[^ /\.$\x22]*`

必需：否

FullDocument

确定 DocumentDB 在文档更新操作期间将向您的事件流发送的内容。如果设置为 UpdateLookup，DocumentDB 将发送一个描述所发生更改的增量以及完整文档的副本。否则，DocumentDB 将仅发送包含更改的部分文档。

类型：字符串

有效值：`UpdateLookup | Default`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Environment

函数的环境变量设置。您可以使用环境变量来调整函数的行为，而无需更新代码。环境变量是存储在函数的版本特定配置中的一对字符串。

内容

Variables

环境变量键值对。有关更多信息，请参阅使用 Lambda 环境变量。

类型：字符串到字符串映射

密钥模式：`[a-zA-Z]([a-zA-Z0-9_])+`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

EnvironmentError

无法应用的环境变量的错误消息。

内容

ErrorCode

错误代码。

类型：字符串

必需：否

Message

错误消息。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

EnvironmentResponse

更新或读取环境变量的操作结果。如果操作成功，则响应将包含环境变量。如果失败，则响应将包含有关错误的详细信息。

内容

Error

无法应用的环境变量的错误消息。

类型：EnvironmentError 对象

必需：否

Variables

环境变量键值对。在 AWS CloudTrail 日志中被忽略。

类型：字符串到字符串映射

密钥模式：`[a-zA-Z]([a-zA-Z0-9_])+`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

内容

Size

函数的 /tmp 目录大小。

类型：整数

有效范围：最小值为 512。最大值为 10240。

必需：是

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

EventSourceMappingConfiguration

AWS资源和 Lambda 函数之间的映射。有关详细信息，请参阅 CreateEventSourceMapping。

内容

AmazonManagedKafkaEventSourceConfig

Amazon Managed Streaming for Apache Kafka (Amazon MSK) 事件源的特定配置设置。

类型：AmazonManagedKafkaEventSourceConfig 对象

必需：否

BatchSize

Lambda 从流或队列中提取并发送到函数的每个批处理中的最大记录数。Lambda 在单次调用中将批处理中的所有记录传递给函数，最高可传递同步调用的负载上限 (6 MB)。

默认值：因服务而异。对于 Amazon SQS，默认值为 10。对于所有其他服务，默认值为 100。

相关设置：当将 `BatchSize` 设置为一个大于 10 的值时，必须至少将 `MaximumBatchingWindowInSeconds` 设置为 1。

类型：整数

有效范围：最小值为 1。最大值为 10000。

必需：否

BisectBatchOnFunctionError

（仅限 Kinesis 和 DynamoDB Streams）如果函数返回错误，则将分处理拆分为两个并重试。默认值为 False。

类型：布尔值

必需：否

DestinationConfig

（仅限 Kinesis、DynamoDB Streams、Amazon MSK 和自行管理的 Apache Kafka 事件源）用于在 Lambda 处理事件后指定事件目标的配置对象。

类型：DestinationConfig 对象

必需：否

DocumentDBEventSourceConfig

DocumentB 事件源的特定配置设置。

类型：DocumentDBEventSourceConfig 对象

必需：否

EventSourceArn

事件源的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

FilterCriteria

定义用于确定 Lambda 是否应处理事件的筛选条件的对象。有关更多信息，请参阅 Lambda 事件筛选。

类型：FilterCriteria 对象

必需：否

FunctionArn

Lambda 函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

FunctionResponseTypes

（Kinesis、DynamoDB Streams 和 Amazon SQS）应用于事件源映射的当前响应类型枚举的列表。

类型：字符串数组

数组成员：最少 0 项。最多 1 项。

有效值：ReportBatchItemFailures

必需：否

LastModified

上次更新事件源映射或其状态更改的日期（以 Unix 时间秒为单位）。

类型：Timestamp

必需：否

LastProcessingResult

函数的最后一次 Lambda 调用结果。

类型：字符串

必需：否

MaximumBatchingWindowInSeconds

Lambda 在调用函数之前收集记录的最长时间，以秒为单位。您可以将 MaximumBatchingWindowInSeconds 配置为介于 0 秒到 300 秒之间的任意值，以秒的整数倍调整。

对于流和 Amazon SQS 事件源，默认批处理时段为 0 秒。对于 Amazon MSK、自行管理的 Apache Kafka、Amazon MQ 和 DocumentDB 事件源，默认批处理时间窗口为 500 毫秒。请注意，由于您只能以秒为增量更改 MaximumBatchingWindowInSeconds，所以在更改后无法恢复到 500 毫秒的默认批处理时段。要恢复原定设置的批处理时段，必须创建新的事件源映射。

相关设置：对于流和 Amazon SQS 事件源，当将 BatchSize 设置为大于 10 的值时，必须将 MaximumBatchingWindowInSeconds 至少设置为 1。

类型：整数

有效范围：最小值为 0。最大值为 300。

必需：否

MaximumRecordAgeInSeconds

（仅限 Kinesis 和 DynamoDB Streams）丢弃超过指定龄期的记录。原定设置值为 -1，该值将最大年龄设置为无限。将该值设置为无限后，Lambda 永远不会删除旧记录。

> **Note**
>
> 最长记录期限的最小有效值为 60 秒。尽管小于 60 和大于 -1 的值都在参数的绝对范围内，但不允许使用它们

类型：整数

有效范围：最小值为 -1。最大值为 604800。

必需：否

MaximumRetryAttempts

（仅限 Kinesis 和 DynamoDB Streams）在指定重试次数后丢弃记录。原定设置值为 -1，该值将最大重试次数设置为无限。将 MaximumRetryAttempts 设置为无限后，Lambda 会重试失败的记录，直到事件源中的记录过期为止。

类型：整数

有效范围：最小值为 -1。最大值为 10000。

必需：否

ParallelizationFactor

（仅限 Kinesis 和 DynamoDB Streams）每个分片中同时处理的批处理数。默认值是 1。

类型：整数

有效范围：最小值为 1。最大值为 10。

必需：否

Queues

(Amazon MQ) 要使用的 Amazon MQ 代理目标队列的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。最大长度为 1000。

模式：`[\s\S]*`

必需：否

ScalingConfig

（仅限 Amazon SQS）事件源的扩缩配置。有关更多信息，请参阅为 Amazon SQS 事件源配置最大并发。

类型：ScalingConfig 对象

必需：否

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

类型：SelfManagedEventSource 对象

必需：否

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

类型：SelfManagedKafkaEventSourceConfig 对象

必需：否

SourceAccessConfigurations

用于保护与定义事件源的身份验证协议数组 VPC 组件或虚拟化主机。

类型：SourceAccessConfiguration 对象数组

数组成员：最少 0 项。最多 22 项。

必需：否

StartingPosition

在流中开始读取数据的位置。对于 Amazon Kinesis 和 Amazon DynamoDB Streams 事件源，为必需项。仅 Amazon Kinesis 流、Amazon DocumentDB、Amazon MSK 和自行管理的 Apache Kafka 支持 AT_TIMESTAMP。

类型：字符串

有效值：TRIM_HORIZON | LATEST | AT_TIMESTAMP

必需：否

StartingPositionTimestamp

将 StartingPosition 设置为 AT_TIMESTAMP 时，即开始读取的时间（以 Unix 时间秒为单位）。StartingPositionTimestamp 不能是未来时间。

类型：时间戳

必需：否

State

事件源映射的状态。此状态可以是以下值之一：Creating、Enabling、Enabled、Disabling、Disabled、Updating 或 Deleting。

类型：字符串

必需：否

StateTransitionReason

表明是用户还是 Lambda 对事件源映射进行了最后一次更改。

类型：字符串

必需：否

Topics

Kafka 主题的名称。

类型：字符串数组

数组成员：固定数量为 1 项。

长度限制：最小长度为 1。长度上限为 249。

模式：^[^.]([a-zA-Z0-9\-_.]+)

必需：否

TumblingWindowInSeconds

（仅限 Kinesis 和 DynamoDB Streams）DynamoDB 和 Kinesis Streams 事件源的处理时间窗口（以秒为单位）。值为 0 秒表示无滑动时间窗口。

类型：整数

有效范围：最小值为 0。最大值为 900。

必需：否

UUID

事件源映射的标识符。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FileSystemConfig

有关 Lambda 函数与 Amazon EFS 文件系统之间连接的详细信息。

内容

Arn

提供对文件系统访问的 Amazon EFS 访问点的 Amazon Resource Name (ARN)。

类型：字符串

长度约束：最大长度为 200。

模式：`arn:aws[a-zA-Z-]*:elasticfilesystem:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}:\d{12}:access-point/fsap-[a-f0-9]{17}`

必需：是

LocalMountPath

函数用来访问文件系统的路径，以 `/mnt/` 开头。

类型：字符串

长度限制：最大长度为 160。

模式：`^/mnt/[a-zA-Z0-9-_.]+$`

必需：是

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Filter

`FilterCriteria` 对象中用于定义事件筛选模式的一种结构。

内容

Pattern

筛选条件模式。有关筛选条件模式语法的更多信息，请参阅筛选条件规则语法。

类型：字符串

长度限制：长度下限为 0。最大长度为 4096。

模式：`.*`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FilterCriteria

包含事件源的筛选条件的对象。

内容

Filters

筛选条件列表。

类型：Filter 对象数组

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FunctionCode

Lambda 函数的代码。您可以指定 Amazon S3 中的对象，直接上载 .zip 文件存档部署包，或指定容器映像的 URI。

内容

ImageUri

Amazon ECR 注册表中容器映像的 URI。

类型：字符串

必需：否

S3Bucket

与您的函数处于同一 AWS 区域 的 Amazon S3 桶。存储桶可能位于不同的 AWS 账户 中。

类型：字符串

长度限制：最小长度为 3。长度上限为 63。

模式：`^[0-9A-Za-z\.\-_]*(?<!\.)$`

必需：否

S3Key

部署程序包的 Amazon S3 密钥。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

S3ObjectVersion

对于版本控制的对象，指要使用的部署程序包对象的版本。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

ZipFile

部署程序包的 base64 编码内容。AWS开发工具包和 AWS CLI 客户端为您处理编码。

类型：Base64 编码的二进制数据对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FunctionCodeLocation

有关函数部署程序包的详细信息。

内容

ImageUri

Amazon ECR 注册表中容器映像的 URI。

类型：字符串

必需：否

Location

可用于下载部署程序包的预签名 URL。

类型：字符串

必需：否

RepositoryType

托管文件的服务。

类型：字符串

必需：否

ResolvedImageUri

镜像的 URI 已解析。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK

- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FunctionConfiguration

有关函数配置的详细信息。

内容

Architectures

该函数支持的指令集架构。架构是具有有效值之一的字符串数组。默认架构值是 x86_64。

类型：字符串数组

数组成员：固定数量为 1 项。

有效值：x86_64 | arm64

必需：否

CodeSha256

函数部署包的 SHA256 哈希。

类型：字符串

必需：否

CodeSize

函数部署包的大小（以字节为单位）。

类型：长整型

必需：否

DeadLetterConfig

函数的死信队列。

类型：DeadLetterConfig 对象

必需：否

Description

函数说明。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

Environment

函数环境变量。在 AWS CloudTrail 日志中被忽略。

类型：EnvironmentResponse 对象

必需：否

EphemeralStorage

函数的 /tmp 目录大小（以 MB 为单位）。默认值为 512，但可以是介于 512 到 10240 MB 之间的任意整数。有关更多信息，请参阅配置短暂存储（控制台）。

类型：EphemeralStorage 对象

必需：否

FileSystemConfigs

Amazon EFS 文件系统的连接设置。

类型：FileSystemConfig 对象数组

数组成员：最多 1 个物品。

必需：否

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_\.]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

FunctionName

函数的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 170。

模式：`(arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]{2}(-gov)?-[a-z]+-\d{1}:)?(\d{12}:)?(function:)?([a-zA-Z0-9-_\.]+)(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

Handler

Lambda 为开始运行您的函数而调用的函数。

类型：字符串

长度限制：最大长度为 128。

模式：`[^\s]+`

必需：否

ImageConfigResponse

函数的镜像配置值。

类型：ImageConfigResponse 对象

必需：否

KMSKeyArn

用于加密函数的环境变量的 AWS KMS key。激活 Lambda SnapStart 时，此密钥还用于加密您的函数的快照。仅当您配置了客户托管密钥时，才会返回此密钥。

类型：字符串

模式：`(arn:(aws[a-zA-Z-]*)?:[a-z0-9-.]+:.*)|()`

必需：否

LastModified

该函数上次更新的日期和时间，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

必需：否

LastUpdateStatus

上次对函数执行的更新的状态。函数创建完成后，首先设置为 Successful。

类型：字符串

有效值：Successful | Failed | InProgress

必需：否

LastUpdateStatusReason

上次对函数执行更新的原因。

类型：字符串

必需：否

LastUpdateStatusReasonCode

函数上次更新的原因代码。

类型：字符串

有效值：EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError | EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout | InvalidRuntime | InvalidZipFileException | FunctionError

必需：否

Layers

函数的层。

类型：Layer 对象数组

必需：否

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

类型：LoggingConfig 对象

必需：否

MasterArn

对于 Lambda@Edge 函数，为主函数的 ARN。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

MemorySize

在运行时 可用于函数的内存 量。

类型：整数

有效范围：最小值为 128。最大值为 10240。

必需：否

PackageType

部署程序包的类型。将容器映像设置为 `Image`，然后将 .zip 文件存档设置为 `Zip`。

类型：字符串

有效值：`Zip | Image`

必需：否

RevisionId

函数或别名的最新更新版本。

类型：字符串

必需：否

Role

函数的执行角色。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+

必需：否

Runtime

函数的运行时的标识符。如果部署包是 .zip 文件归档，则需要使用运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

必需：否

RuntimeVersionConfig

运行时的 ARN和发生的任何错误。

类型：RuntimeVersionConfig 对象

必需：否

SigningJobArn

签名任务的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

必需：否

SigningProfileVersionArn

签名配置文件版本的 ARN。

类型：字符串

模式：arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)

必需：否

SnapStart

发布函数版本时，将 ApplyOn 设置为 PublishedVersions 以创建初始化执行环境的快照。有关更多信息，请参阅 Improving startup performance with Lambda SnapStart（使用 Lambda SnapStart 提高启动性能）。

类型：SnapStartResponse 对象

必需：否

State

函数的当前状态。当状态为 Inactive，您可以通过调用函数来重新激活该函数。

类型：字符串

有效值：Pending | Active | Inactive | Failed

必需：否

StateReason

函数当前状态的原因。

类型：字符串

必需：否

StateReasonCode

函数当前状态的原因代码。当代码为 Creating 时，无法调用或修改该函数。

类型：字符串

有效值：Idle | Creating | Restoring | EniLimitExceeded | InsufficientRolePermissions | InvalidConfiguration | InternalError | SubnetOutOfIPAddresses | InvalidSubnet | InvalidSecurityGroup | ImageDeleted | ImageAccessDenied | InvalidImage | KMSKeyAccessDenied | KMSKeyNotFound | InvalidStateKMSKey | DisabledKMSKey | EFSIOError

```
| EFSMountConnectivityError | EFSMountFailure | EFSMountTimeout |
InvalidRuntime | InvalidZipFileException | FunctionError
```

必需：否

Timeout

Lambda 在停止函数前允许其运行的时长（以秒为单位）。

类型：整数

有效范围：最小值为 1。

必需：否

TracingConfig

函数的 AWS X-Ray 跟踪配置。

类型：TracingConfigResponse 对象

必需：否

Version

Lambda 函数的版本。

类型：字符串

长度限制：最小长度为 1。长度上限为 1024。

模式：`(\$LATEST|[0-9]+)`

必需：否

VpcConfig

函数的网络配置。

类型：VpcConfigResponse 对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FunctionEventInvokeConfig

内容

DestinationConfig

在事件发送到函数以进行处理后的目标。

目标

- Function - Lambda 函数的 Amazon 资源名称（ARN）。
- Queue - 标准 SQS 队列的 ARN。
- Topic - 标准 SNS 主题的 ARN。
- Event Bus - Amazon EventBridge 事件总线的 ARN。

类型：DestinationConfig 对象

必需：否

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

LastModified

上次更新配置的日期和时间（以 Unix 时间秒为单位）。

类型：Timestamp

必需：否

MaximumEventAgeInSeconds

Lambda 发送到函数以进行处理的请求的最长期限。

类型：整数

有效范围：最小值为 60。最大值为 21600。

必需：否

MaximumRetryAttempts

在函数返回错误时重试的最大次数。

类型：整数

有效范围：最小值为 0。最大值为 2。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

FunctionUrlConfig

有关 Lambda 函数 URL 的详细信息。

内容

AuthType

函数 URL 使用的身份验证类型。如果想将访问权限限制为仅经过身份验证的用户，请设置为 AWS_IAM。如果想绕过 IAM 身份验证以创建公有端点，请设置为 NONE。有关更多信息，请参阅 Lambda 函数 URL 的安全性和身份验证模型。

类型：字符串

有效值：NONE | AWS_IAM

必需：是

CreationTime

创建函数 URL 时，使用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

必需：是

FunctionArn

函数的 Amazon Resource Name (ARN)。

类型：字符串

模式：arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?

必需：是

FunctionUrl

函数的 HTTP URL 端点。

类型：字符串

长度限制：最小长度为 40。最大长度为 100。

必需：是

LastModifiedTime

上次更新函数 URL 时，采用 ISO-8601 格式 (YYYY-MM-DDThh:mm:ss.sTZD)。

类型：字符串

必需：是

Cors

适用于函数 URL 的 cross-origin resource sharing (CORS)（跨源资源共享）设置。

类型：Cors 对象

必需：否

InvokeMode

使用以下选项之一：

- BUFFERED – 这是默认选项。Lambda 通过 Invoke API 操作调用函数。负载完成后，调用结果可用。最大负载大小为 6MB。
- RESPONSE_STREAM – 函数能够在负载结果可用时对其进行流式处理。Lambda 通过 InvokeWithResponseStream API 操作调用函数。最大响应负载大小为 20MB，但是，您可以请求提高限额。

类型：字符串

有效值：BUFFERED | RESPONSE_STREAM

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ImageConfig

覆盖容器映像 Dockerfile 设置的配置值。有关更多信息，请参阅容器映像设置。

内容

Command

指定要通过 ENTRYPOINT 传入的参数。

类型：字符串数组

数组成员：最多 1500 项。

必需：否

EntryPoint

指定应用程序的入口点，通常是运行时可执行文件的位置。

类型：字符串数组

数组成员：最多 1500 项。

必需：否

WorkingDirectory

指定工作目录。

类型：字符串

长度限制：最大长度为 1000。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK

- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ImageConfigError

对 `GetFunctionConfiguration` 的错误响应。

内容

ErrorCode

错误代码。

类型：字符串

必需：否

Message

错误消息。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ImageConfigResponse

对 `GetFunctionConfiguration` 请求的响应 。

内容

Error

对 `GetFunctionConfiguration` 的错误响应。

类型：ImageConfigError 对象

必需：否

ImageConfig

覆盖容器镜像 Dockerfile的配置值。

类型：ImageConfig 对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

InvokeResponseStreamUpdate

流式处理响应负载的区块。

内容

Payload

Lambda 函数返回的数据。

类型：Base64 编码的二进制数据对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

InvokeWithResponseStreamCompleteEvent

用于确认事件流完成的响应。

内容

ErrorCode

错误代码。

类型：字符串

必需：否

ErrorDetails

任何返回错误的详细信息。

类型：字符串

必需：否

LogResult

Base64 编码执行日志的最后 4 KB。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

InvokeWithResponseStreamResponseEvent

包含响应负载区块的对象。流结束后，Lambda 会包含 `InvokeComplete` 对象。

内容

InvokeComplete

当流结束并且所有负载区块均已返回后所返回的对象。

类型：InvokeWithResponseStreamCompleteEvent 对象

必需：否

PayloadChunk

流式处理响应负载的区块。

类型：InvokeResponseStreamUpdate 对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

Layer

一个 AWS Lambda 层。

内容

Arn

函数层的 Amazon Resource Name (ARN)。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

必需：否

CodeSize

层存档的大小（以字节为单位）。

类型：长整型

必需：否

SigningJobArn

签名作业的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

SigningProfileVersionArn

签名配置文件的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

LayersListItem

有关某个 AWS Lambda 层的详细信息。

内容

LatestMatchingVersion

层的最新版本。

类型：LayerVersionsListItem 对象

必需：否

LayerArn

函数层的 Amazon Resource Name (ARN)。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+`

必需：否

LayerName

层的名称。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`(arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+)|[a-zA-Z0-9-_]+`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

LayerVersionContentInput

一个包含某个 AWS Lambda 层内容的 ZIP 归档文件。您可以指定 Amazon S3 位置，或者直接上载层归档。

内容

S3Bucket

层存档的 Amazon S3 存储桶。

类型：字符串

长度限制：最小长度为 3。长度上限为 63。

模式：`^[0-9A-Za-z\.\-_]*(?<!\.)$`

必需：否

S3Key

层存档的 Amazon S3 密钥。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

S3ObjectVersion

对于进行版本控制的对象，则为要使用的层存档对象版本。

类型：字符串

长度限制：最小长度为 1。最大长度为 1024。

必需：否

ZipFile

层归档的 base64 编码内容。AWS开发工具包和 AWS CLI 客户端为您处理编码。

类型：Base64 编码的二进制数据对象

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

LayerVersionContentOutput

有关某个 AWS Lambda 层的某个版本的详细信息。

内容

CodeSha256

层归档的 SHA-256 哈希值。

类型：字符串

必需：否

CodeSize

层存档的大小（以字节为单位）。

类型：长整型

必需：否

Location

指向 Amazon S3 中层归档的链接，有效期为 10 分钟。

类型：字符串

必需：否

SigningJobArn

签名作业的 Amazon Resource Name (ARN)。

类型：字符串

必需：否

SigningProfileVersionArn

签名配置文件的 Amazon Resource Name (ARN)。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

LayerVersionsListItem

有关某个 AWS Lambda 层的某个版本的详细信息。

内容

CompatibleArchitectures

兼容指令集架构列表。

类型：字符串数组

数组成员：最多 15 项。

有效值：x86_64 | arm64

必需：否

CompatibleRuntimes

层的兼容运行时。

以下列表包括已弃用的运行时。有关更多信息，请参阅运行时弃用策略。

类型：字符串数组

数组成员：最多 15 项。

有效值：nodejs | nodejs4.3 | nodejs6.10 | nodejs8.10 | nodejs10.x | nodejs12.x | nodejs14.x | nodejs16.x | java8 | java8.al2 | java11 | python2.7 | python3.6 | python3.7 | python3.8 | python3.9 | dotnetcore1.0 | dotnetcore2.0 | dotnetcore2.1 | dotnetcore3.1 | dotnet6 | nodejs4.3-edge | go1.x | ruby2.5 | ruby2.7 | provided | provided.al2 | nodejs18.x | python3.10 | java17 | ruby3.2 | python3.11 | nodejs20.x | provided.al2023 | python3.12 | java21

必需：否

CreatedDate

版本的创建日期，采用 ISO 8601 格式。例如，2018-11-27T15:10:45.123+0000。

类型：字符串

必需：否

Description

关于版本的描述。

类型：字符串

长度约束：最小长度为 0。长度上限为 256。

必需：否

LayerVersionArn

层版本的 ARN。

类型：字符串

长度限制：最小长度为 1。长度上限为 140。

模式：`arn:[a-zA-Z0-9-]+:lambda:[a-zA-Z0-9-]+:\d{12}:layer:[a-zA-Z0-9-_]+:[0-9]+`

必需：否

LicenseInfo

层的开源许可证。

类型：字符串

长度约束：最大长度为 512。

必需：否

Version

版本号。

类型：长整型

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

LoggingConfig

该函数的 Amazon CloudWatch Logs 配置设置。

内容

ApplicationLogLevel

设置此属性可筛选 Lambda 发送到 CloudWatch 的函数的应用程序日志。Lambda 仅在选择的详细信息级别及以下发送应用程序日志，其中 TRACE 是最高级别，FATAL 是最低级别。

类型：字符串

有效值：`TRACE | DEBUG | INFO | WARN | ERROR | FATAL`

必需：否

LogFormat

Lambda 将函数的应用程序和系统日志发送到 CloudWatch 的格式。在纯文本和结构化的 JSON 之间进行选择。

类型：字符串

有效值：`JSON | Text`

必需：否

LogGroup

由函数向其发送日志的 Amazon CloudWatch 日志组的名称。默认情况下，Lambda 函数将日志发送到名为 `/aws/lambda/<function name>` 的默认日志组。要使用其他日志组，请输入现有的日志组或输入新日志组名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 512。

模式：`[\.\-_/#A-Za-z0-9]+`

必需：否

SystemLogLevel

设置此属性可筛选 Lambda 发送到 CloudWatch 的函数的系统日志。Lambda 仅在选择的详细信息级别及以下发送系统日志，其中 DEBUG 是最高级别，WARN 是最低级别。

类型：字符串

有效值：DEBUG | INFO | WARN

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

OnFailure

处理失败的事件的目标。

内容

Destination

目标资源的 Amazon 资源名称（ARN）。

要保留异步调用的记录，您可以将 Amazon SNS 主题、Amazon SQS 队列、Lambda 函数或 Amazon EventBridge 事件总线配置为目标。

要保留来自 Kinesis 和 DynamoDB 事件源的失败调用记录，您可以将 Amazon SNS 主题或 Amazon SQS 队列配置为目标。

要保留来自自行管理的 Kafka 或 Amazon MSK 的失败调用记录，您可以将 Amazon SNS 主题、Amazon SQS 队列或 Amazon S3 存储桶配置为目标。

类型：字符串

长度限制：最小长度为 0。最大长度为 350。

模式：`^$|arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

OnSuccess

已成功处理的事件的目标。

内容

Destination

目标资源的 Amazon Resource Name (ARN)。

类型：字符串

长度限制：最小长度为 0。最大长度为 350。

模式：`^$|arn:(aws[a-zA-Z0-9-]*):([a-zA-Z0-9\-])+:([a-z]{2}(-gov)?-[a-z]+-\d{1})?:(\d{12})?:(.*)`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ProvisionedConcurrencyConfigListItem

有关函数别名或版本的预配并发配置的详细信息。

内容

AllocatedProvisionedConcurrentExecutions

输入要分配的预置并发数量。在线性部署和金丝雀部署期间使用加权别名时，该值会根据为函数版本预配置的并发量而波动。

类型：整数

有效范围：最小值为 0。

必需：否

AvailableProvisionedConcurrentExecutions

可用的预置并发数量。

类型：整数

有效范围：最小值为 0。

必需：否

FunctionArn

别名或版本的 Amazon Resource Name (ARN)。

类型：字符串

模式：`arn:(aws[a-zA-Z-]*)?:lambda:[a-z]{2}(-gov)?-[a-z]+-\d{1}:\d{12}:function:[a-zA-Z0-9-_]+(:(\$LATEST|[a-zA-Z0-9-_]+))?`

必需：否

LastModified

用户上次更新配置的日期和时间，格式为 ISO 8601。

类型：字符串

必需：否

RequestedProvisionedConcurrentExecutions

请求的预置并发数量。

类型：整数

有效范围：最小值为 1。

必需：否

Status

分配过程的状态。

类型：字符串

有效值：`IN_PROGRESS | READY | FAILED`

必需：否

StatusReason

分配失败的原因，即无法分配预置并发的原因。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

RuntimeVersionConfig

运行时的 ARN和发生的任何错误。

内容

Error

Lambda 无法检索函数的运行时版本时的错误响应。

类型：RuntimeVersionError 对象

必需：否

RuntimeVersionArn

您希望函数使用的运行时版本的 ARN。

类型：字符串

长度限制：最小长度为 26。最大长度为 2048。

模式：`^arn:(aws[a-zA-Z-]*):lambda:[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\d{1}::runtime:.+$`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

RuntimeVersionError

无法检索函数的运行时版本信息时返回的任何错误。

内容

ErrorCode

错误代码。

类型：字符串

必需：否

Message

错误消息。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

ScalingConfig

（仅限 Amazon SQS）事件源的扩展配置。若要删除配置，请传递空值。

内容

MaximumConcurrency

限制 Amazon SQS 事件源可以调用的并发实例数。

类型：整数

有效范围：最小值为 2。最大值为 1000。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

SelfManagedEventSource

适用于事件源的自行管理 Apache Kafka 集群。

内容

Endpoints

适用于以下格式的 Kafka 代理的 Bootstrap 服务器列表：`"KAFKA_BOOTSTRAP_SERVERS": ["abc.xyz.com:xxxx","abc2.xyz.com:xxxx"]`。

类型：字符串到字符串数组映射

映射条目：最多 2 项。

有效密钥：`KAFKA_BOOTSTRAP_SERVERS`

数组成员：最少 1 项。最多 10 项。

长度限制：最小长度为 1。最大长度为 300。

模式：`^(([a-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\-]*[A-Za-z0-9]):[0-9]{1,5}`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

SelfManagedKafkaEventSourceConfig

自行管理的 Apache Kafka 事件源的特定配置设置。

内容

ConsumerGroupId

要加入的 Kafka 使用者组的标识符。在所有 Kafka 事件源中，使用者组 ID 必须是唯一的。在使用指定的使用者组 ID 创建 Kafka 事件源映射后，无法更新此值。有关更多信息，请参阅可自定义的使用者组 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 200。

模式：`[a-zA-Z0-9-\/*:_+=.@-]*`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

SnapStart

函数的 Lambda SnapStart 设置。发布函数版本时，将 ApplyOn 设置为 PublishedVersions 以创建初始化执行环境的快照。

内容

ApplyOn

发布函数版本时，设置为 PublishedVersions 以创建初始化执行环境的快照。

类型：字符串

有效值：PublishedVersions | None

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

SnapStartResponse

函数的 SnapStart 设置。

内容

ApplyOn

设置为 PublishedVersions 时，Lambda 会在您发布函数版本时创建执行环境的快照。

类型：字符串

有效值：PublishedVersions | None

必需：否

OptimizationStatus

当您提供限定的 Amazon 资源名称（ARN）时，此响应元素会指示是否已针对指定的函数版本激活 SnapStart。

类型：字符串

有效值：On | Off

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

SourceAccessConfiguration

要保护和定义对事件源的访问，您可以指定身份验证协议、VPC 组件或虚拟主机。

内容

Type

适用于事件源的身份验证协议、VPC 组件或虚拟主机的类型。例如："Type":"SASL_SCRAM_512_AUTH"。

- BASIC_AUTH – （Amazon MQ）存储代理凭证的 AWS Secrets Manager 密钥。
- BASIC_AUTH – （自行管理的 Apache Kafka）用于 Apache Kafka 代理 SASL/PLAIN 身份验证密钥的 Secrets Manager ARN。
- VPC_SUBNET - （自行管理的 Apache Kafka）与您的 VPC 关联的子网。Lambda 会连接到这些子网，从您自行管理的 Apache Kafka 集群中获取数据。
- VPC_SECURITY_GROUP - （自行管理的 Apache Kafka）用于管理对自行管理的 Apache Kafka 代理访问的 VPC 安全组。
- SASL_SCRAM_256_AUTH - （自行管理的 Apache Kafka）用于自行管理的 Apache Kafka 代理 SASL SCRAM-256 身份验证私有密钥的 Secrets Manager ARN。
- SASL_SCRAM_512_AUTH - （Amazon MSK、自行管理的 Apache Kafka）用于自行管理的 Apache Kafka 代理 SASL SCRAM-512 身份验证私有密钥的 Secrets Manager ARN。
- VIRTUAL_HOST - （RabbitMQ）您的 RabbitMQ 代理中虚拟主机的名称。Lambda 将使用此 RabbitMQ 主机作为事件源。无法在 updateEventSourceMapping API 调用中指定此属性。
- CLIENT_CERTIFICATE_TLS_AUTH – （Amazon MSK，自行管理的 Apache Kafka）私有密钥的 Secrets Manager ARN，包含用于 MSK/Apache Kafka 代理的双向 TLS 身份验证的证书链（X.509 PEM）、私有密钥（PKCS#8 PEM）和私有密钥密码（可选）。
- SERVER_ROOT_CA_CERTIFICATE – （自行管理的 Apache Kafka）私有密钥的 Secrets Manager ARN，包含用于 Apache Kafka 代理的 TLS 加密的根 CA 证书（X.509 PEM）。

类型：字符串

有效值：BASIC_AUTH | VPC_SUBNET | VPC_SECURITY_GROUP | SASL_SCRAM_512_AUTH | SASL_SCRAM_256_AUTH | VIRTUAL_HOST | CLIENT_CERTIFICATE_TLS_AUTH | SERVER_ROOT_CA_CERTIFICATE

必需：否

URI

Type 中所选配置的值。例如："URI": "arn:aws:secretsmanager:us-east-1:01234567890:secret:MyBrokerSecretName"。

类型：字符串

长度限制：最小长度为 1。最大长度为 200。

模式：[a-zA-Z0-9-\/*:_+=.@-]*

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

TracingConfig

函数的 AWS X-Ray 跟踪配置。要对传入请求进行采样和记录，请将 Mode 设置为 Active。

内容

Mode

跟踪模式。

类型：字符串

有效值：Active | PassThrough

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

TracingConfigResponse

函数的 AWS X-Ray 跟踪配置。

内容

Mode

跟踪模式。

类型：字符串

有效值：`Active | PassThrough`

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

VpcConfig

附加到 Lambda 函数 VPC 安全组和子网。有关更多信息，请参阅配置 Lambda 函数以访问 VPC 中的资源。

内容

Ipv6AllowedForDualStack

允许连接到双堆栈子网的 VPC 函数上的出站 IPv6 流量。

类型：布尔值

必需：否

SecurityGroupIds

VPC 的安全组 ID 列表。

类型：字符串数组

数组成员：最多 5 项。

必需：否

SubnetIds

VPC 子网 ID 的列表。

类型：字符串数组

数组成员：最多 16 项。

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK

- 适用于 Ruby V3 的 AWS SDK

VpcConfigResponse

附加到 Lambda 函数 VPC 安全组和子网。

内容

Ipv6AllowedForDualStack

允许连接到双堆栈子网的 VPC 函数上的出站 IPv6 流量。

类型：布尔值

必需：否

SecurityGroupIds

VPC 的安全组 ID 列表。

类型：字符串数组

数组成员：最多 5 项。

必需：否

SubnetIds

VPC 子网 ID 的列表。

类型：字符串数组

数组成员：最多 16 项。

必需：否

VpcId

VPC 的 ID。

类型：字符串

必需：否

另请参阅

有关在特定语言的 AWS SDK 中使用此 API 的更多信息，请参阅以下内容：

- 适用于 C++ 的 AWS SDK
- 适用于 Go 的 AWS SDK
- 适用于 Java V2 的 AWS SDK
- 适用于 Ruby V3 的 AWS SDK

常见参数

以下列表包含所有操作用于使用查询字符串对 Signature Version 4 请求进行签名的参数。任何特定于操作的参数都列在该操作的主题中。有关 Signature Version 4 的更多信息，请参阅《IAM 用户指南》中的签署 AWS API 请求。

Action

要执行的操作。

类型：字符串

必需：是

Version

编写请求所针对的 API 版本，格式为 YYYY-MM-DD。

类型：字符串

必需：是

X-Amz-Algorithm

您用于创建请求签名的哈希算法。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

有效值：`AWS4-HMAC-SHA256`

必需：条件

X-Amz-Credential

凭证范围值，该值是一个字符串，其中包含您的访问密钥、日期、您要定位的区域、您请求的服务以及终止字符串（“aws4_request”）。值采用以下格式表示：access_key/YYYYMMDD/region/service/aws4_request。

有关更多信息，请参阅《IAM 用户指南》中的创建已签名的 AWS API 请求。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

X-Amz-Date

用于创建签名的日期。格式必须为 ISO 8601 基本格式 (YYYYMMDD'T'HHMMSS'Z')。例如，以下日期时间是有效的 X-Amz-Date 值：`20120325T120000Z`。

条件：X-Amz-Date 对于所有请求都是可选的；它可以用于覆盖对请求签名所使用的日期。如果以 ISO 8601 基本格式指定 Date 标头，则不需要 X-Amz-Date。使用 X-Amz-Date 时，它始终会覆盖 Date 标头的值。有关更多信息，请参阅《IAM 用户指南》中的 AWS API 请求签名的元素。

类型：字符串

必需：条件

X-Amz-Security-Token

通过调用 AWS Security Token Service（AWS STS）获得的临时安全令牌。有关支持来自 AWS STS 的临时安全凭证的服务列表，请参阅《IAM 用户指南》中的使用 IAM 的 AWS 服务。

条件：如果您使用来自 AWS STS 的临时安全凭证，则必须包含安全令牌。

类型：字符串

必需：条件

X-Amz-Signature

指定从要签名的字符串和派生的签名密钥计算的十六进制编码签名。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

X-Amz-SignedHeaders

指定作为规范请求的一部分包含的所有 HTTP 标头。有关指定已签名标头的更多信息，请参阅《IAM 用户指南》中的创建已签名的 AWS API 请求。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

常见错误

本部分列出了所有 AWS 服务的常见 API 操作错误。对于特定于此服务的 API 操作的错误，请参阅该 API 操作的主题。

AccessDeniedException

您没有足够的访问权限，无法执行该操作。

HTTP 状态代码：403

ExpiredTokenException

请求中包含的安全令牌已到期。

HTTP 状态代码：403

IncompleteSignature

请求签名不符合 AWS 标准。

HTTP 状态代码：403

InternalFailure

由于未知错误、异常或故障，请求处理失败。

HTTP 状态代码：500

MalformedHttpRequestException

HTTP 级别的请求问题，例如，我们无法根据内容编码指定的解压缩算法对正文进行解压缩。

HTTP 状态代码：400

NotAuthorized

您无权执行此操作。

HTTP 状态代码：401

OptInRequired

AWS 访问密钥 ID 需要订阅服务。

HTTP 状态代码：403

RequestAbortedException

当请求在回复发送前中止请求时（例如客户端关闭连接）可以使用的便捷异常。

HTTP 状态代码：400

RequestEntityTooLargeException

HTTP 级别的请求问题。请求实体太大。

HTTP 状态代码：413

RequestExpired

请求到达服务的时间超过请求上的日期戳 15 分钟或超过请求到期日期 15 分钟（例如，对于预签名 URL），或者请求上的日期戳比当前时间晚了 15 分钟以上。

HTTP 状态代码：400

RequestTimeoutException

HTTP 级别的请求问题。读取请求超时。

HTTP 状态代码：408

ServiceUnavailable

由于服务器发生临时故障而导致请求失败。

HTTP 状态代码：503

ThrottlingException

由于请求限制而导致请求被拒绝。

HTTP 状态代码：400

UnrecognizedClientException

在我们的记录中没有所提供的 X.509 证书或 AWS 访问密钥 ID。

HTTP 状态代码：403

UnknownOperationException

所请求的操作无效。确认正确键入了操作。

HTTP 状态代码：404

ValidationError

输入未能满足 AWS 服务指定的约束。

HTTP 状态代码：400