



販売者ガイド

AWS Marketplace



AWS Marketplace: 販売者ガイド

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

Amazon の商標とトレードドレスは、Amazon 以外の製品またはサービスとの関連において、顧客に混乱を招いたり、Amazon の名誉または信用を毀損するような方法で使用することはできません。Amazon が所有しない他の商標はすべてそれぞれの所有者に帰属します。所有者は必ずしも Amazon との提携や関連があるわけではありません。また、Amazon の支援を受けているとはかぎりません。

Table of Contents

とは AWS Marketplace	1
販売 AWS Marketplace 者としての の使用	1
の契約構造 AWS Marketplace	3
での製品の料金 AWS Marketplace	4
販売者としての開始方法	6
無料のソフトウェア製品を公開するための販売者の要件	6
有料製品の販売者の追加要件	7
有料製品の対象となる管轄区域	7
AWS Marketplace 管理ポータル	8
販売者登録プロセス	9
公開プロファイルの作成	10
税に関する情報の入力	11
米国の銀行口座情報の入力	11
本人確認プロセスの完了	13
銀行口座検証プロセスの完了	15
(オプション) 本人確認手順にセカンダリユーザーを追加する	16
支払いと購入者への請求	16
すでに販売者ですか?	18
苦情処理ポリシー - Amazon Payments Europe	18
出品料金	20
パブリックオファーの出品料金	20
プライベートオファーの出品料金	20
チャンネルパートナーのプライベートオファー (CPPO) の出品料金	21
プロフェッショナルサービスの出品料金	21
販売者ツールキット	21
AWS Marketplace コマース分析サービス	21
AWS Marketplace フィールドデモンストレーションプログラム	39
その他のリソース AWS Marketplace 管理ポータル	40
製品の準備	41
製品デリバリー	41
製品の料金	45
料金モデル	45
料金モデルの変更	50
料金の変更	50

プライベートオファー	50
製品の返金	50
リージョンと国	53
AWS リージョン	54
Countries (国)	54
標準契約書	55
の標準契約 AWS Marketplace	55
のリセラー契約 AWS Marketplace	57
カテゴリとメタデータ	57
製品の名前付けと説明	58
カテゴリとキーワードの選択	59
AMI とコンテナ製品の使用手順	60
.....	58
CloudFormation 配信	61
アプリケーション機能のモニタリングと評価	62
プログラムによるシステム認証情報と暗号化キーのローテーション	62
製品の検索エンジン最適化	63
検索エンジンの最適化	63
AWS Marketplace 検索	64
プライベートオファーの準備	67
プライベートオファーの仕組み	67
プライベートオファーに関する考慮事項	68
購入者向けのプライベートオファーエクスペリエンス	69
プライベートオファーのレポート	70
サポートされる製品タイプ	71
AMI 製品のプライベートオファー	71
コンテナ製品のプライベートオファー	71
プロフェッショナルサービス製品のプライベートオファー	72
SaaS 製品のプライベートオファー	72
ML 製品のプライベートオファー	73
プライベートオファーの作成と管理	74
新しいプライベートオファーの開始	74
オファーのステータスを理解する	75
プライベートオファーのドラフトと公開	75
購入者へのプライベートオファーの送信	77
プライベートオファーの進行状況を保存する	78

プライベートオファターの有効期限の更新	78
プライベートオファターのキャンセル	79
チャンネルパートナーのプライベートオファター	79
追加情報	81
ISV としての再販オポチュニティの作成	81
分割プラン	83
支払いスケジュールの作成	84
分割プランのレポート	85
プライベートオファターのアップグレード、更新、修正	85
プライベートオファターの契約の管理	85
プライベートオファターの修正でサポートされる製品タイプ	86
アップグレードと更新の提示プロセス	86
アップグレード、更新、修正に関するレポート	88
将来の日付の契約	88
将来の日付の契約の作成	89
将来の日付の契約での分割プランの使用	90
将来の日付の契約の通知を受け取る	90
チャンネルパートナーのプライベートオファターの再販に関する将来の日付の契約を使用す る	90
AMI ベースの製品	92
AMI ベースの製品配信方法	92
AMI ベースの製品について	93
製品のライフサイクル	93
AMI 製品コード	95
変更リクエスト	96
製品ロードフォーム	97
年間契約の修正	98
単一 AMI 製品	99
前提条件	100
セルフサービスエクスペリエンスを理解する	100
単一 AMI 製品の作成	102
変更リクエストを作成する	107
変更リクエストのステータスを取得する	109
製品情報を更新する	110
許可リスト (プレビューアカウント) を更新する	111
製品の可視化の更新	111

を追加する AWS リージョン	112
の制限 AWS リージョン	113
将来の AWS リージョンのサポートの更新	114
インスタンスを追加する	115
インスタンスの制限	115
バージョン情報を更新する	116
新しいバージョンを追加する	117
バージョンを制限する	120
料金を更新する	122
国別の可用性の更新	123
EULA を更新する	123
返金ポリシーの更新	124
AMI AWS Marketplace へのアクセスを許可する	125
AWS Marketplaceから製品を削除する	126
変更リクエストを送信する際によくあるエラーのトラブルシューティング	127
を使用した AMI ベースの配信 CloudFormation	128
製品の出品の構築	129
CloudFormation テンプレートの準備	130
テンプレートインフラストラクチャの料金見積りの取得	132
アーキテクチャ図	133
送信要件の一致	134
製品リクエストの送信	136
サーバーレスアプリケーションコンポーネントの追加	136
AMI 構築のベストプラクティス	145
再販権の確保	145
AMI の構築	145
の AMI の準備と保護 AWS Marketplace	146
公開要件を確認するための AMI のスキャン	147
AWS Marketplace AMI でソフトウェアが実行されていることを確認する	148
AMI 製品の料金	149
AMI 料金モデル	149
AWS 料金とソフトウェア料金	153
AMI 製品のカスタム計量価格	154
AMI 製品の契約料金	164
AMI 製品の請求、計測、ライセンスの統合	168
AWS Marketplace Metering Serviceを使用した AMI 製品のカスタム計量	168

AWS License Managerとの AMI 製品の契約料金	173
AMI 製品に関する Amazon SNS 通知	187
Amazon SNS トピック: aws-mp-subscription-notification	188
Amazon SNS トピックへ Amazon SQS キューをサブスクライブする	188
AMI 製品チェックリスト	189
AMI ベースの製品要件	191
セキュリティポリシー	191
アクセスポリシー	192
カスタマー情報ポリシー	193
製品使用ポリシー	194
アーキテクチャポリシー	195
AMI 製品使用説明書	195
コンテナベースの製品	196
ヘルプの利用	197
コンテナ製品の開始方法	197
前提条件	198
コンテナ製品の作成	198
製品のライフサイクル	199
コンテナ製品 (レガシー) の更新	200
製品の可視化の更新	200
AWS アカウント ID の許可リストを更新します。	201
料金設定ディメンションの追加	201
ディメンション情報の更新	202
料金条件の更新	202
国別の可用性の更新	203
エンドユーザーライセンス契約の更新	204
製品の返金ポリシーの更新	204
コンテナ製品の製品 ID と製品コードを作成します。	205
初期リストの作成	205
コンテナ製品 (レガシー) の料金詳細の作成または更新	206
コンテナ製品 AWS Marketplace Metering Service との統合	208
コンテナ製品 AWS License Manager との統合	208
製品の新しいバージョンを追加する。	208
製品のテストとリリース	216
バージョン情報の更新	218
Amazon EKS アドオンのバージョンを制限する	220

コンテナ製品の製品情報の作成または更新	220
コンテナ製品の公開 (レガシー)	221
コンテナ製品はセキュリティ上の問題をスキャンします。	222
コンテナベースの製品要件	222
セキュリティ要件	223
アクセスの要件	223
顧客情報の要件	223
製品の使用要件	224
アーキテクチャの要件	225
コンテナ製品の使用手順	226
Amazon EKS アドオン製品の要件	226
コンテナ製品の料金	242
コンテナ料金モデル	243
コンテナ製品の契約料金	247
コンテナ製品の請求、計測、ライセンスの統合	250
AWS Marketplace Metering Serviceによる時間単位計測およびカスタム計測	251
との契約料金 AWS License Manager	253
AWS Marketplace Metering Service による時間単位の計測	255
コンテナ製品のカスタム計測	266
を使用したコンテナ製品の契約料金 AWS License Manager	279
コンテナ製品の Amazon SNS 通知	313
Amazon SNS トピック: aws-mp-subscription-notification	313
Amazon SNS トピックへ Amazon SQS キューをサブスクライブする	314
Machine Learning 製品	315
機械学習製品の使用を開始する	315
SageMaker モデルパッケージ	315
SageMaker アルゴリズム	316
推論モデルのデプロイ	316
セキュリティと知的財産	317
知的財産の保護	317
ネットワークアクセスなし	317
顧客データのセキュリティ	317
機械学習製品の料金	318
インフラストラクチャの料金	318
ソフトウェアの料金	318
で製品を準備する SageMaker	321

コードをイメージにパッケージ化する	321
イメージのアップロード	344
Amazon SageMaker リソースの作成	347
AWS Marketplaceでの製品の公開	354
公開プロセスの概要	354
必要なアクセス許可	354
製品リストの作成	355
製品のテスト	363
公開のサインオフ	364
製品の更新	364
機械学習製品を作成するための要件とベストプラクティス	366
必須アセット	366
ML 製品の一般的なベストプラクティス	367
使用状況情報の要件	367
入力と出力の要件	368
Jupyter Notebook の要件	369
ML 製品リストの要件と推奨事項の要約	370
サービスの制限とクォータ	374
ネットワークの隔離	374
イメージのサイズ	375
ストレージサイズ	375
インスタンスサイズ	375
推論のペイロードサイズ	375
推論の処理時間	376
Service Quotas	376
非同期推論	376
サーバーレス推論	376
マネージドスポットトレーニング	376
Docker イメージと AWS アカウント	376
組み込みアルゴリズムまたは からのモデルパッケージの公開 AWS Marketplace	377
公開 AWS リージョン でサポート	377
トラブルシューティング	378
レポート作成	379
日別ビジネスレポート	380
月別収益レポート	380
支払いレポート	380

その他のレポートと分析	380
SaaS ベースの製品	381
SaaS 製品を開始する	381
前提条件	382
SaaS 製品のライフサイクル	382
SaaS 製品の作成	383
初期 SaaS 製品ページを作成する	387
SaaS 製品設定	389
SaaS サブスクリプション製品の統合	400
SaaS 契約製品を統合する	403
SaaS 契約を pay-as-you-go 製品と統合する	407
サーバーレス SaaS 統合ソリューションのデプロイ	411
SaaS 製品の計画	412
料金の計画	413
請求の統合の計画	413
Amazon SNS インテグレーションを計画する	413
顧客が製品にアクセスする方法の計画	413
SaaS 製品ガイドライン	414
製品設定ガイドライン	415
顧客情報の要件	415
製品使用ガイドライン	415
アーキテクチャガイドライン	416
SaaS 製品の料金	417
SaaS サブスクリプションの料金	418
SaaS 契約の料金	420
SaaS 無料トライアル	424
SaaS 無料トライアルオファーの作成	425
SaaS 無料トライアルオファーのキャンセル	426
SaaS の顧客のオンボーディング	426
新しい購入者を受け入れるための SaaS 製品の設定	427
SaaS 製品に関する Amazon SNS 通知	429
Amazon SNS トピック: aws-mp-entitlement-notification	430
Amazon SNS トピック: aws-mp-subscription-notification	431
SNS トピックへの SQS キューのサブスクライブ	432
AWS Marketplace Metering Service API および使用権限管理サービス API へのアクセス	433
使用量の計測	434

使用権限の確認	440
SaaS 製品統合チェックリスト	441
レポート作成	445
SaaS のコード例	446
ResolveCustomer コード例	446
GetEntitlement コード例	447
BatchMeterUsage コード例	448
使用量割り当てタグ付きの BatchMeterUsage のコード例 (オプション)	450
PrivateLink での AWS の使用 AWS Marketplace	452
序章	452
製品の設定	454
製品を に送信する AWS Marketplace	455
VPC エンドポイントへの購入者のアクセス	455
付録: チェックリスト	456
プロフェッショナルサービス製品	458
ヘルプの利用	459
プロフェッショナルサービス製品の開始方法	459
前提条件	459
プロフェッショナルサービス製品の作成	460
プライベートオファターの作成	461
製品情報の編集	463
製品の料金の編集	464
製品の可視性の編集	464
プロフェッショナルサービス製品の削除	465
製品の詳細	466
製品の説明	467
追加リソース	468
サポート情報	468
料金ディメンション	469
製品の可視性	469
製品要件	469
製品設定ガイドライン	469
顧客情報の要件	470
製品使用ガイドライン	470
アーキテクチャガイドライン	471
プロフェッショナルサービス製品の料金	471

データ製品	473
製品の送信	474
[製品] タブの使用	475
会社と製品のロゴ要件	476
有料の再パッケージ版ソフトウェアを送信するための要件	477
ハードウェアコンポーネントを使用する製品の要件	478
AWS CloudFormationが発表した製品 (無料または有料) または使用量ベースの有料 AMI 製品	478
製品の送信	478
製品の更新	479
製品の変更と更新	479
タイミングと心構え	480
への AMIsの送信 AWS Marketplace	481
AMI セルフサービススキャン	481
AMI クローン作成と製品コード割り当て	482
最終チェックリスト	482
製品のマーケティング	484
180 日間の GTM アカデミー	484
製品が入手可能になったことの発表	484
AWS Marketplace メッセージング	485
のレビュー AWS Marketplace	486
へのリンク AWS Marketplace	487
AWS Marketplace ロゴの使用	487
で製品に直接リンクする AWS Marketplace	487
プレスリリース	488
AWS Marketplace 商標使用ガイドライン	489
通知	491
E メール通知	491
イベントタイプ	492
通知を管理する	492
Amazon EventBridge 通知	494
AWS Marketplace Catalog API Amazon EventBridge イベント	494
Amazon Simple Notification Service Notifications	501
出品者レポート、データフィード、ダッシュボード	503
販売者配送データフィードサービス	503
データフィードの保存と構造	504

データフィードへのアクセス	506
データフィードからのサブスクリプションの解除	509
データフィードの使用	509
データフィードテーブルの概要	510
データフィードクエリの例	518
データフィード	534
販売者レポート	572
レポートへのアクセス	572
日別ビジネスレポート	573
日次顧客サブスクライバーレポート	583
支払いレポート	586
月別請求済み収益レポート	594
販売補償レポート	603
米国の消費税と使用税レポート	606
補足レポート	610
契約詳細レポート	610
販売者ダッシュボード	612
ダッシュボードへのアクセス	572
財務業務用ダッシュボード	614
販売業務用ダッシュボード	642
AWS Marketplace Vendor Insights	660
AWS Marketplace Vendor Insights について	661
販売者として設定する	662
セキュリティプロファイルを作成する	662
証明書のアップロード	663
自己評価をアップロードする	665
AWS Audit Manager 自動評価を有効にする	666
プロファイルの表示	672
販売者としてのセキュリティプロファイルを確認する	673
スナップショットの管理	673
スナップショットを作成する	674
スナップショットを表示する	675
スナップショットをエクスポートする	675
最新リリースのスナップショットを表示する	676
スナップショットリリースを延期する	676
スナップショットリストの設定を変更する	677

アクセスの制御	677
AWS Marketplace Vendor Insights 販売者のアクセス許可	678
CreateDataSource	678
DeleteDataSource	679
GetDataSource	679
UpdateDataSource	679
ListDataSources	679
CreateSecurityProfile	679
ListSecurityProfiles	680
GetSecurityProfile	680
AssociateDataSource	680
DisassociateDataSource	681
UpdateSecurityProfile	681
ActivateSecurityProfile	681
DeactivateSecurityProfile	681
UpdateSecurityProfileSnapshotCreationConfiguration	682
UpdateSecurityProfileSnapshotReleaseConfiguration	682
ListSecurityProfileSnapshots	682
GetSecurityProfileSnapshot	682
TagResource	683
UntagResource	683
ListTagsForResource	683
追加リソース	683
セキュリティ	191
の IAM AWS Marketplace	686
ユーザーの作成	688
グループの作成または使用	689
ユーザーとしてのサインイン	691
AWS Marketplace 販売者のポリシーとアクセス許可	691
ポリシー	692
アクセス許可	693
AWS マネージドポリシー	700
AWSMarketplaceAmiIngestion	701
AWSMarketplaceFullAccess	701
AWSMarketplaceGetEntitlements	704
AWSMarketplaceMeteringFullAccess	705

AWSMarketplaceMeteringRegisterUsage	705
AWSMarketplaceSellerFullAccess	706
AWSMarketplaceSellerProductsFullAccess	709
AWSMarketplaceSellerProductsReadOnly	711
AWSVendorInsightsVendorFullAccess	712
AWSVendorInsightsVendorReadOnly	714
ポリシーの更新	715
AWS Marketplace Commerce Analytics Service アカウントのアクセス許可	719
Amazon SQS のアクセス許可	720
AWS Marketplace 計測と使用権限 API のアクセス許可	721
SaaS 製品の IAM ポリシー	721
AMI 製品の IAM ポリシー	722
コンテナ製品の IAM ポリシー	722
サービスリンクロールの使用	723
再販承認のロール	723
を使用した AWS Marketplace API コールのログ記録 AWS CloudTrail	728
AWS Marketplace 計測 API ログファイルエントリの例	729
ドキュメント履歴	735
AWS 用語集	756
.....	dcclvii

とは AWS Marketplace

AWS Marketplace は厳選されたデジタルカタログで、お客様はサードパーティーのソフトウェア、データ、サービスを検索、購入、デプロイ、管理してソリューションを構築し、ビジネスを運営できます。AWS Marketplace には、ヘルスケア、金融サービス、通信など、特定の業界におけるセキュリティ、ビジネスアプリケーション、機械学習、データ製品などの一般的なカテゴリからの数千のソフトウェアリストが含まれています。お客様は、数回クリックするだけで事前設定されたソフトウェアをすばやく起動し、ソフトウェアソリューションを Amazon マシンイメージ (AMI) 形式、Software as a Service (SaaS)、およびその他の形式で選択できます。また、サードパーティー製ソフトウェアの設定、デプロイ、管理を支援するプロフェッショナルサービスも利用できます。配送方法の詳細なリストについては、「[製品の配送](#)」を参照してください。

購入者 (サブスクライバー)、販売者 (プロバイダー)、またはその両方 AWS Marketplace として使用できます。を持つユーザーは誰でも購入者 AWS Marketplace としてを使用 AWS アカウントでき、販売者に登録できます。販売者は、独立系ソフトウェアベンダー (ISV)、チャンネルパートナー、マネージドサービスプロバイダー (MSP)、または AWS 製品やサービスと連携する何かを提供する個人です。

Note

データ製品プロバイダーは、AWS Data Exchange の資格要件を満たす必要があります。詳細については、「AWS Data Exchange ユーザーガイド」の「[AWS Data Exchange でのデータ製品の提供](#)」を参照してください。

対象となるパートナーは、の外部でプログラムで AWS Marketplace 製品を一覧表示できます AWS Marketplace。対象となるパートナーになる方法の詳細については、AWS Marketplace ビジネス開発パートナーにお問い合わせください。

次の動画では、での販売について詳しく説明します AWS Marketplace。

[AWS Marketplace の紹介](#)

販売 AWS Marketplace 者としての の使用

でソフトウェア製品を販売するプロセス AWS Marketplace には、次の 7 つのステップが含まれます。

販売者のプロセス

[ステップ]	[アクション]	説明
1	登録	販売者は、まずに登録します AWS Marketplace 管理ポータル。既存の AWS 組織と簡単にリンク AWS アカウント できる新しい専用 を実装することをお勧めします。AWS パートナーの税情報が管轄区域の資格基準を満たしていることを確認します。既存の米国の銀行口座を持たない AWS パートナーの場合、 Hyperwallet を使用して追加料金なしで作成できます。
2	製品タイプを決定する	販売する製品の種類を決定します。で製品タイプを作成する方法の詳細については AWS Marketplace、以下を参照してください。 <ul style="list-style-type: none">• AMI ベースの製品• コンテナベースの製品• Machine Learning 製品• SaaS ベースの製品• プロフェッショナルサービス製品• データ製品 (データ製品の詳細については、「AWS Data Exchange ユーザーガイド」の「AWS Data Exchange とは」を参照してください)。
3	製品を準備する	パッケージと料金スキームを設定し、製品が表示される関連するカテゴリを決定して、製品が関連のある検索に表示されるようにキーワードを追加します。調達プロセスを簡素化するには、公開出品とプライベートオファアの両方で 標準化されたライセンス条項 を使用できます。
4	製品を送信する	製品を AWS Marketplace で入手可能にするには、製品送信プロセスを使用します。製品はシンプルにすることができます。料金体系が 1 つの、単一の Amazon マシンイメージ (AMI) がその例です。または、複数の AMI、AWS CloudFormation テンプレート、複雑な価格オプションと支払いスケジュールを含めて、複雑にすることもできます。

[ステップ]	[アクション]	説明
5	製品を売る	への認識を高め、の製品ページに直接トラフィックを誘導 AWS Marketplace することで、製品の成功に貢献します AWS Marketplace。
6	レポートとデータフィードを表示する	販売者として登録したら、を使用して製品の使用状況レポート AWS Marketplace 管理ポータル にアクセスします。 AWS Marketplace は、製品の販売に関する情報を収集および分析するた めのツールを提供します。
7	製品を管理する	AWS Marketplace 管理ポータル を使用してアカウントと製品ペー ジを管理します。

販売者は、[AWS Marketplace 管理ポータル](#) に移動して登録します。製品の使用に対して請求する場合は、登録の一部として税金と銀行の情報も指定する必要があります。登録時に AWS Marketplace で検索可能な会社のプロファイルまたは自分自身のプロファイルを作成します。[AWS Marketplace 管理ポータル](#) を使用して製品用の製品ページを作成して管理することもできます。

の契約構造 AWS Marketplace

で販売されるソフトウェア、サービス、およびデータ製品の使用は、購入者と販売者の間の契約 AWS Marketplace に準拠します。AWS はこれらの契約の当事者ではありません。

販売者としての契約には以下が含まれます。

- 購入者とのエンドユーザーライセンス契約 (EULA)。この契約は、AWS Marketplace に出品されている公開ソフトウェアの製品リストページに記載されています。多くの出品者は、[AWS Marketplace 標準契約書 \(SCMP\)](#) をデフォルトの EULA として使用しています。SCMP をプライベートオファーの交渉のベースとして使用したり、修正テンプレートを使用して SCMP を変更したりすることもできます。プライベートオファーには、当事者間で交渉したカスタム契約条件を含めることもできます。

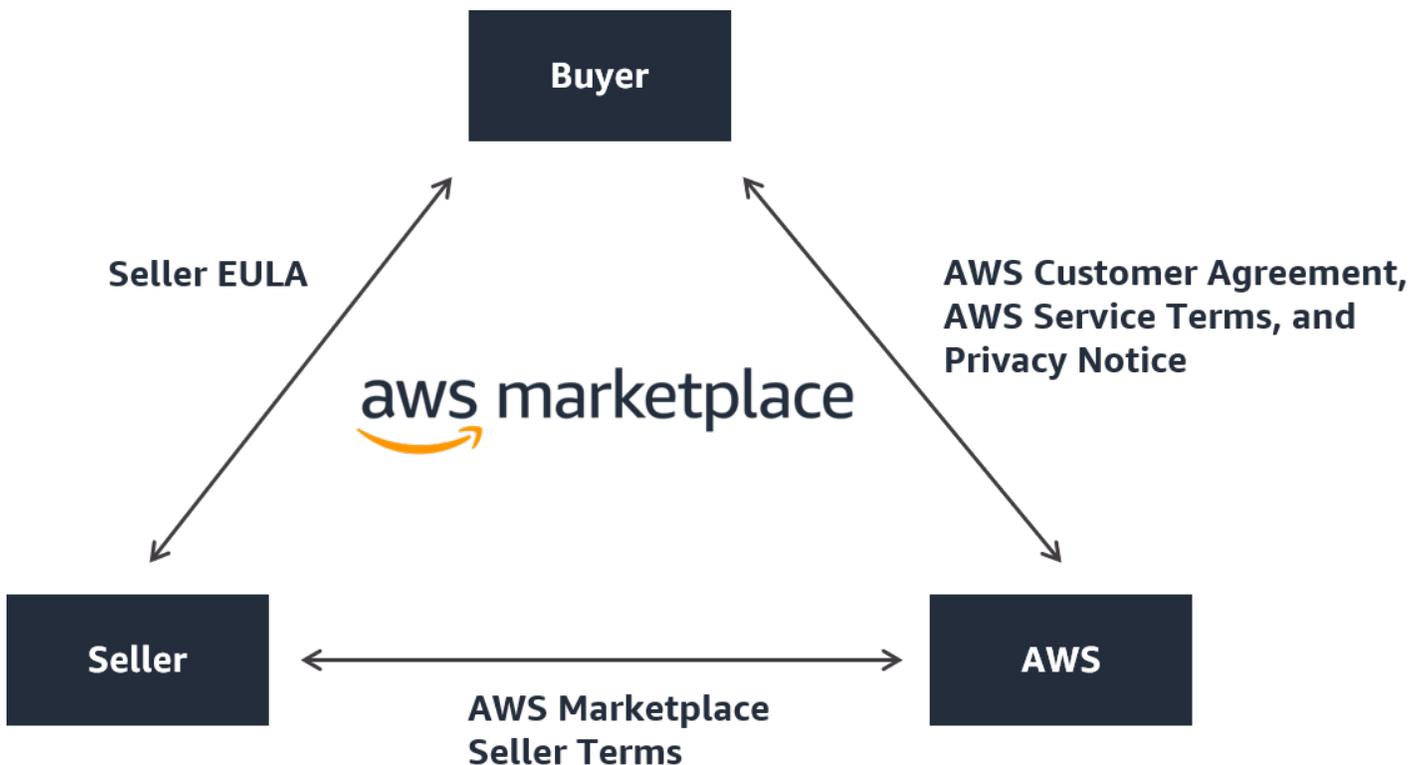
Note

オファータイプと料金モデルに基づいて EULA 更新が行われるタイミングについては、「AWS Marketplace 購入者ガイド」の「[EULA 更新](#)」を参照してください。

- [AWS Marketplace 販売者規約](#)。AWS Marketplaceでのアクティビティを規定しています。

購入者による の使用 AWS Marketplace には、[AWS サービス条件](#)、[AWS カスタマーアグリーメント](#)、および[プライバシー通知](#)が適用されます。

次の図は、 の契約構造を示しています AWS Marketplace。



での製品の料金 AWS Marketplace

では AWS Marketplace、製品は無料で使用でき、関連する料金が発生する可能性があります。料金は購入者の AWS 請求書の一部となり、購入者が支払うと、販売者に AWS 支払いが行われます。製品はたくさんの形式を持つことができます。例えば、製品を購入者の AWS アカウントを使用してインスタンス化された Amazon マシンイメージ (AMI) として提供することができます。購入者への配送に テンプレートを使用する CloudFormation のように製品を設定することもできます。製品は、ISV

からの SaaS サービス、ウェブアクセスコントロールリスト (ウェブ ACL)、ルールセット、または AWS WAF の条件の場合もあります。ISV、チャネルパートナー、MSP が提供するプロフェッショナルサービスもあります。

柔軟な料金オプションには、無料トライアル、時間単位、月単位、年単位、複数年単位、BYOL (Bring Your Own License Model)、1 つのソースから請求される料金が含まれます。請求と支払いは 1 つのソースから AWS 処理され、請求額はお客様の AWS 請求に表示されます。

ソフトウェア製品は、ISV の標準のエンドユーザーライセンス契約 (EULA) を利用して出品された価格で購入できます。さらに、ソフトウェア製品は、独自の料金や EULA を適用してプライベートオファーで提供することもできます。製品は、時間や使用範囲を指定した契約の下で購入することもできます。製品をサブスクライブした後、購入者は AWS Service Catalog を使用して製品をコピーし、購入者の組織で製品にアクセスして使用方法を管理できます。購入者のエクスペリエンスについて詳しくは、「<https://docs.aws.amazon.com/marketplace/latest/buyerguide/service-catalog.html>」を参照してください。料金の詳細については、「[the section called “製品の料金”](#)」を参照してください。

販売者としての開始方法

でソフトウェアを販売する場合は AWS Marketplace、要件を確認し、手順に従って販売者として登録します。お住まいの場所と、どのような製品を販売するかによって、登録要件が異なります。で販売者として登録するには AWS Marketplace、既存の を使用する AWS アカウント が、新しいアカウントを作成します。すべての AWS Marketplace インタラクションは、選択したアカウントに関連付けられます。

メモ

- AWS Marketplace 販売者として登録することは、AWS Data Exchange にデータ製品を一覧表示し、で利用できるようにするための前提条件です AWS Marketplace。これらの要件の詳細については、「AWS Data Exchange ユーザーガイド」の「[AWS Data Exchange でのデータ製品の提供](#)」を参照してください。
- AWS Marketplace 販売者が必要とするアクセス許可については、「」を参照してください [AWS Marketplace 販売者のポリシーとアクセス許可](#)。
- 商品出品手数料の詳細について、登録済み販売者は AWS Marketplace 管理ポータル「[AWS Marketplace 販売者規約](#)」を参照することができます。

無料のソフトウェア製品を公開するための販売者の要件

で製品を提供するときに製品に課金するかどうかにかかわらず AWS Marketplace、その製品を販売します。顧客への費用は 0.00 USD ですが、販売者と顧客は製品の使用に関する相互契約に同意することになります。無料の製品のみを提供する場合は、AWS Marketplaceに銀行情報を提供する必要はありません。で無料製品を作成して提供するには AWS Marketplace、以下を実行する必要があります。

- 全機能を使用できる一般公開版の本番環境用ソフトウェアを販売する。
- 明確なカスタマーサポートプロセスおよびサポート組織があること。
- ソフトウェアを定期的に更新し、脆弱性の影響を受けないようにする手段を提供している。
- AWS Marketplaceで製品を販売するときのベストプラクティスとガイドラインに従う。
- 優良な AWS 顧客であり、AWS Marketplace 販売者の利用規約の要件を満たしている。

有料製品の販売者の追加要件

製品の代金を請求する場合、または自分のライセンス使用モデル (BYOL) 製品を販売する場合は、以下の要件も満たし、以下の追加情報を提供する必要があります。

- [対象となる管轄区域](#)の永住者または市民であるか、これらの区域のいずれかで組織化または法人化された事業体である必要があります。
- 税金および銀行口座の情報を提供する必要があります。米国を拠点とする企業の場合は、W-9 フォームと、米国を拠点とする銀行からの銀行口座が必要です。
- 米国以外の販売者は、(i) W-8 フォーム、付加価値税 (VAT)、商品・サービス税 (GST) 登録番号、および (ii) 米国の銀行情報を提供する必要があります。米国銀行口座がない場合は、[Hyperwallet](#) から仮想米国銀行口座を登録できます。
- データ製品を提供するには、の[ケース作成](#)ウィザードを使用してオンボーディングをリクエストする必要もあります AWS Support。
- Amazon Web Services EMEA SARL を通じて、欧州、中東、アフリカ (EMEA) (トルコと南アフリカを除く) の国と地域に拠点 AWS アカウント を置く顧客に製品を販売するには、[顧客確認プロセス](#)を完了する必要があります。加えて:
 - 最大 2 つの支払いを受け取ります (AWS Inc. および Amazon Web Services EMEA SARL を介した取引の場合)。
 - 場所によっては、特定の取引について出品手数料に課税される場合があります。税金の詳細については、「[AWS Marketplace 出品者の税金](#)」のヘルプページを参照してください。出品料に付加価値税 (VAT) が課せられる場合、AWS Marketplace は税制に準拠した請求書を発行します。
 - Amazon Web Services EMEA SARL の詳細については、「[Amazon Web Services ヨーロッパ FAQ](#) ウェブサイト」の「AWS EMEA Marketplace - 出品者」を参照してください。

に販売するには AWS GovCloud (US) Region、販売者は[AWS GovCloud \(US\) アカウント](#)を持っている必要があります。ITAR 要件の詳細については、「[AWS GovCloud \(US\) ユーザーガイド](#)」を参照してください。

AWS Marketplace 販売者の要件や登録プロセスに関するご質問は、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。

有料製品の対象となる管轄区域

で有料ソフトウェアを販売するには AWS Marketplace、次のいずれかの国または SARs の永住者または市民であるか、その中に組織または法人化された事業体である必要があります。

- オーストラリア¹
- バーレーン^{1 2}
- 欧州連合 (EU) 加盟国¹
- 香港特別行政区
- イスラエル^{1 2}
- 日本^{1 2 3}
- ニュージーランド¹
- ノルウェー^{1 2}
- カタール
- スイス^{1 2}
- アラブ首長国連邦 (UAE)^{1 2}
- 英国 (UK)¹
- 米国 (US)

¹ これらの国の有料製品の販売者は、設立国の VAT 登録情報を提供する必要があります。

² 販売者が購入者と同じ国に居住している場合、販売者は税金の請求、回収、送金の責任を担う場合があります。税務顧問にお問い合わせください。

³ 日本を拠点とする販売者には、出品料金に対する日本の消費税 (JCT) を申告する義務があります。他の法域に拠点を置く販売者も同様の義務を負っている場合があります。税務顧問にお問い合わせください。

出品者としての VAT、請求、および納税義務に関する詳細については、「[Amazon Web Service 税務ヘルプ](#)」の「[AWS Marketplace 出品者](#)」を参照してください。

上記の国または SARs [AWS Marketplace 「管轄区域外の企業のリソース」](#) を参照してください。

AWS Marketplace 管理ポータル

[AWS Marketplace 管理ポータル](#) は、AWS Marketplace 販売者として登録するために使用するツールです。その後、ポータルを使用して、AWS Marketplace で販売する商品を管理できます。ポータルでは、以下のタスクを完了できます。

- AWS Marketplace 販売者として登録します。

- [製品] ページを使用して、新しいソフトウェア製品を送信したり、既存のソフトウェア製品を更新したりする。
- リクエストのステータスをモニタリングする。
- 新しいソフトウェア製品を作成して管理するために必要なファイルをアップロードする。
- go-to-market アクティビティを活用して、ソフトウェア製品を増分チャネル収益に管理します。
- ローンチから数時間以内に、使用状況およびキャンペーンを使用した収益を含め、マーケティング活動の結果を測定する。
- カスタマーサービス担当者がリアルタイムで顧客データを取得できるようにします。
- 自動 Amazon マシンイメージ (AMI) スキャンを開始して脆弱性を検出する。

Note

データ製品は AWS Data Exchange コンソールから公開および管理します。AWS Data Exchange プロバイダーは AWS Marketplace 管理ポータル、を使用して、販売者としての登録、AWS Data Exchange オンボーディングのリクエスト、販売者のレポートへのアクセス、返金リクエストの送信を行うことができます。

登録済みのすべての販売者は、製品の作成に使用したアカウントの AWS 認証情報 AWS Marketplace 管理ポータル を使用して にアクセスできます。使用するアカウントは、顧客が製品をサブスクライブするときの登録販売者として定義されます。製品の登録販売者である特定アカウントの決定に関してサポートが必要な場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

AWS Marketplace では、ルートアカウントの認証情報を使用するのではなく、AWS Identity and Access Management (IAM) ロールを使用して AWS Marketplace 管理ポータル にサインインすることを強くお勧めします。詳細については、「[AWS Marketplace セキュリティ](#)」を参照してください。

販売者登録プロセス

を の販売者として登録することで AWS Marketplace、製品やサービスを他の AWS Marketplace 顧客に販売できます。

販売者として登録するには、以下の手順が必要です。

1. パブリックプロフィールを作成する – 会社名やロゴなど、会社について知らせる に表示される情報を AWS Marketplace 購入者に提供します。このプロセスが完了すると、製品を無料で販売できます。有料製品を販売するには、ステップ 2 と 3 を完了させる必要があります。
2. 税に関する情報の提供 - 支払い済み売上に対する税金の評価、報告、源泉徴収 (該当する場合) を適切に実施するには、税金と付加価値税 (VAT) の情報を入力する必要があります。
3. 銀行情報の提供 – が売上の支払い AWS Marketplace を行えるように、米国の銀行情報を提供します。

これらの手順については、次のセクションで詳しく説明します。

アカウントを販売者として登録したら、 を通じて購入者に販売する製品を作成できます AWS Marketplace。詳細については、「[製品の準備](#)」を参照してください。

AWS Identity and Access Management (IAM) を使用して、さまざまなアクセス許可を持つ AWS アカウント 複数のユーザーが にアクセスできるようにプライマリを設定できます AWS Marketplace 管理ポータル。詳細については、「[the section called “の IAM AWS Marketplace”](#)」を参照してください。

公開プロフィールの作成

登録する最初のステップは、プライマリ AWS Marketplace アカウント AWS アカウント として使用する を選択し、 コンソールで AWS Marketplace 潜在的な購入者に表示される情報を提供することです。このアカウントは、 で製品の登録販売者 AWS Marketplace となり、 からお客様 AWS Marketplace への報告、支払い、通信に使用されます。

を使用して販売者として AWS アカウント 登録し、 に製品を一覧表示すると AWS Marketplace、その製品に関連付けられたアカウントを変更することはできません。AWS Marketplace 販売者として登録するには、新しいアカウントを使用することをお勧めします。ただし、2017 年 9 月 27 日以降に作成されたアカウントであれば、既存のアカウントを使用できます。

公開プロフィールを作成するには

1. [AWS Marketplace 管理ポータル](#) (AMMP) から [今すぐ登録] を選択し、選択した販売者 AWS アカウントにサインインします。
2. [公開プロフィールを追加] を選択して販売者情報を入力します。

公開プロフィールを作成すると、無料の製品を公開して販売できます。有料製品を販売するには、税および銀行に関する情報を提供する必要があります。

税に関する情報の入力

AWS Marketplace が製品の売上に対する税金を正確に報告し、源泉徴収できるように、税金と付加価値税 (VAT) (該当する場合) の情報を入力する必要があります。

税に関する情報を入力するには

1. [AWS Marketplace 管理ポータル](#) にサインインして [設定] を選択します。
2. [支払い情報] セクションの [税金ダッシュボードに移動] を選択します。
3. 米国税に関する質問に回答してください。
 - プロフェッショナルサービスを で販売するには AWS Marketplace、DAC7 の税務調査を完了する必要があります。
4. 税に関する情報を入力したら、[設定] ページに戻り、[VAT 情報の入力] を選択します (可能な場合)。この選択により、AWS Billing コンソールの税金設定ページにリダイレクトされます。

Note

VAT 情報セクションは、VAT をサポート AWS リージョンしている いる場合にのみ使用できます。

税務書類へのアクセス

1099 フォームなどの税務書類には、 からアクセスできます AWS Marketplace 管理ポータル。

税務書類にアクセスするには

1. [AWS Marketplace 管理ポータル](#) にサインインして [設定] を選択します。
2. 支払い情報セクションに移動します。
3. 関連する納税申告書 (1099K または DAC7) を選択します。
4. 納税申告書がある場合は、税務ダッシュボードページでダウンロードできます。

米国の銀行口座情報の入力

で有料製品を販売するすべての販売者には、米国の銀行口座が必要です AWS Marketplace。は米国の銀行口座 AWS Marketplace にのみ支払います。

Note

で有料製品を提供できる国のリストについては、AWS Marketplace「」を参照してください。[有料製品の対象となる管轄区域](#)。

米国の銀行情報を入力するには

1. [AWS Marketplace 管理ポータル](#) にサインインして [設定] を選択します。
2. [支払い情報] セクションで [銀行情報の入力] を選択します。
3. 米国の銀行口座に関する必要な情報を入力します。

Note

税に関する情報 (および該当する場合は付加価値税情報) をまだ入力していない場合、銀行情報を入力することはできません。

米国の銀行口座をまだお持ちでない場合は、[Hyperwallet](#) を通じて口座を取得できる場合があります。[Hyperwallet](#) は、AWS Marketplace 支払いのために提供できる米国の口座を提供できます。

[Hyperwallet](#) は、資金を他の銀行口座にサポートされている通貨で送金することを可能にする、独立したサービスプロバイダーです。期間限定で、AWS Marketplace の支払いに対する特定の [Hyperwallet](#) サービス料金を支払うことは要求されません。

- [Hyperwallet](#) アカウントの詳細を AWS Marketplace 販売者アカウントに追加することで、AWS Marketplace がお客様の名前、E メールアドレス、およびアカウント番号を と共有 [Hyperwallet](#) し、販売者としてのステータスを確認することに同意し、AWS Marketplace 承認します。
- [Hyperwallet](#) サービス (資金を現地の通貨に送金するために必要な送金手数料および為替手数料を含みます) の使用に加え、為替レートに追加料金が適用される場合があります。[Hyperwallet](#) サービス料金は、有料製品からアカウントへの収益 AWS Marketplace の支払いに関してのみ、期間限定で免除されます [Hyperwallet](#)。詳細については、[Hyperwallet](#) サイトの料金セクションを参照してください。または、[Hyperwallet](#) まで詳細と適用される料金をお問い合わせください。サービスの詳細については、[Hyperwallet サポートサイト](#) を参照してください。

Hyperwallet への登録を開始し、米国の銀行口座情報を取得するには

1. [AWS Marketplace 管理ポータル](#) にログインして [設定] を選択します。次に [支払い情報] セクションの [銀行情報の入力] を選択します。
2. Hyperwallet 口座を持っておらず、AWS Marketplace で使用する必要がある場合は、[米国の銀行口座をお持ちですか?] と [Hyperwallet に登録していますか?] に対して [いいえ] を選択してください。Hyperwallet に登録するための個人識別番号 (PIN) とリンクが提供されます。
3. Hyperwallet アカウントを有効にしたら、Hyperwallet 登録ポータルに記載されているステップに従って登録を完了し、預金口座情報を受け取ります。
4. からアカウントを取得したらHyperwallet、 にサインイン AWS アカウントして にHyperwallet アカウント情報を追加します[AWS Marketplace 管理ポータル](#)。次に [設定] を選択し、[支払い情報] セクションで [銀行情報の入力] を選択します。

本人確認プロセスの完了

本人確認 (KYC) は、金融機関やオンラインビジネスが顧客の身元を確認するために使用するコンプライアンス要件です。この要件は、改正された決済サービス指令 (PSD 2) と、銀行やその他の決済機関などの金融機関に適用される EU マネーロンダリング防止指令によるものです。

AWS Marketplace Amazon Web Services EMEA SARL 経由のトランザクションは、ルクセンブルクの認可された電子マネー機関である Amazon Payments Europe, S.C.A. (APE) を通じて処理されます。この金融機関は、支払いサービスを利用するためにお客様の ID の検証を必要とします。

販売者が Amazon Web Services EMEA SARL 経由で取引を行うには、KYC プロセスを完了させる必要があります。このプロセスには、会社に関する追加情報、主な連絡先、実質的所有者、補足書類の提供が含まれます。

KYC プロセスを完了させるには

1. AWS Marketplace 管理ポータルで、設定 を選択します。
2. [アカウントサマリ] セクションで、表示されている [国] が正しいことを確認します。

Note

[情報] リンクをクリックすると、国を変更する方法が表示されます。

3. [KYC 情報に移動] または [本人確認 (KYC)] タブを選択し、[KYC コンプライアンスを開始] を選択すると、KYC 登録ポータルにリダイレクトされます。

による情報の使用と共有方法の詳細については AWS Marketplace、[Amazon Payments Europe プライバシー通知](#)」を参照してください。

4. [KYC 概要に移動] を選択します。
5. [本人確認 (KYC) の概要] で、必要な情報と書類のリストに目を通し、必要な書類を集めてください (まだ行っていない場合)。次に、[KYC コンプライアンスを続行] を選択します。
6. 指示に従って [基本情報] を入力します。Amazon Payments Europe 利用規約を確認したら、[同意して続行] を選択します。

KYC プロセスの次のページまたは次のステップに進むと、Amazon Payments Europe 利用規約に同意したものとみなされます。

質問がある場合は、コンソールの右側にある [よくある質問 (FAQ)] を参照してください。

7. 指示に従って必要な [ビジネス情報] を入力し、[次へ] を選択します。

 Note

[次へ] を選択して次のステップに進むたびに、情報が保存されます。

8. 指示に従って必要な [連絡先情報] を入力し、[次へ] を選択します。
9. [実質的所有者] が連絡先と同じかどうかを選択し、必要に応じて実質的所有者 (最大 4 人) を追加します。追加内容を確認して [次へ] を選択します。
10. 法定代理人が連絡先または実質的所有者と同じかどうかを選択します。法定代理人が別のエンティティである場合は、必要な情報を入力し、エントリを保存して、次へ を選択します。
11. [その他の書類] については、事業許可証、身分証明書、委任状 (該当する場合) をアップロードしてください。
12. [確認と送信] で、入力したすべての情報を確認して確認します。

必要に応じて [編集] を選択して前のセクションに戻ることができます。

13. [検証のため送信] をクリックします。

KYC コンプライアンスの状況がレビューされます (通常は 24 時間以内)。レビューが完了すると、Eメールで通知されます。通常、KYC プロセス全体で約 2 週間かかります。

[設定] タブに戻ると、[アカウントサマリ] カードで KYC コンプライアンスの状況を確認できます。KYC ステータスの詳細については、[アカウントサマリ] カードで [本人確認 (KYC)] タブをクリックしてください。レビューが完了するまで [レビュー中] と表示されます。

KYC が確認された後で、APE を通じて支払いを受け取るには、[支払い情報] タブに銀行取引明細を入力する必要があります。

銀行口座検証プロセスの完了

Amazon Payments Europe (APE) からの支払いを受け取るには、追加情報を入力して、AWS Marketplace 管理ポータル の [支払い情報] タブに記載されている支払い銀行口座を検証する必要があります。

追加の銀行情報の入力

追加の銀行情報を入力するには

1. にサインインし AWS Marketplace 管理ポータル、設定 を選択します。
2. [支払い情報] セクションで [銀行情報の更新] を選択します。
3. 適切な支払い口座を選択します。

[検証ステータス] には [未検証] と表示されます。

4. 確認を選択します。
5. 銀行口座検証 登録ポータルにリダイレクトされるので、そこで銀行取引明細書をアップロードして送信できます。

Hyperwallet 仮想銀行口座ソリューションを使用している場合は、「[the section called “Hyperwallet からの銀行取引明細書のダウンロード”](#)」を参照してください。

6. ポータルで [銀行書類のアップロード] を選択し、[送信] を選択します。

Hyperwallet からの銀行取引明細書のダウンロード

[Hyperwallet仮想銀行口座ソリューション](#)を使用する販売者は、以下の手順で Hyperwallet 銀行取引明細書をダウンロードできます。その後、[銀行口座検証プロセスの完了](#) に記載の指示に従って銀行書類をアップロードできます。

Hyperwallet から銀行取引明細書をダウンロードするには

1. [Hyperwallet アカウント](#) にサインインします。
2. [\[預金口座情報\]](#) ページに移動します。
3. [\[銀行口座確認書\]](#) をダウンロードします。

(オプション) 本人確認手順にセカンダリユーザーを追加する

Note

支払い情報を更新するには、ユーザーは多要素認証 (MFA) を有効にする必要があります。MFA の詳細については、「[IAM の多要素認証 \(MFA\)](#)」を参照してください。

セカンダリユーザーとは、KYC 情報の修正、資金や返金の流れの管理、銀行口座情報などの財務情報の変更ができる個人のことです。

KYC 認証を受けたセカンダリユーザーのみが、前述の更新を行うことができます。こうしたセカンダリユーザーは、ルートアカウントの所有者と同じ継続的なスクリーニング管理の対象となります。

セカンダリユーザーが KYC 認証を受けるには、[本人確認プロセスの完了](#) に記載の手順を完了させる必要があります。

本人確認手順にセカンダリユーザーを追加するには

1. ユーザーに AWS Marketplace 管理ポータルにサインインするよう依頼します。
2. [設定] タブに移動します。
3. [本人確認 (KYC)] タブを選択し、[セカンダリユーザー情報] のセクションを確認します。
4. [セカンダリユーザー情報の入力] を選択します。

[セカンダリユーザー] 登録ポータルにリダイレクトされます。

5. [セカンダリユーザー] 登録ポータルで、必須フィールドを入力し、[次へ] を選択します。
6. [確認と送信] ページで、本人確認書類 ([パスポートのアップロード]) と住所証明 ([ドキュメントのアップロード]) のコピーをアップロードします。
7. [検証のため送信] をクリックします。

KYC コンプライアンスの状況がレビューされます (通常は 24 時間以内)。レビューが完了すると、Eメールで通知されます。通常、KYC プロセス全体で約 2 週間かかります。

支払いと購入者への請求

AWS Marketplace 独立系ソフトウェアベンダー (ISVs やチャネルパートナーなどの販売者は、未払い残高を受け取るための支払い設定を行うことができます。販売者は、日次または月次の支払いオプションを選択し、支払いを受ける日を選択できます。

支払いについて設定するには

1. にサインインし AWS Marketplace 管理ポータル、設定 を選択します。
2. [支払い情報] タブをクリックし [銀行情報の更新] を選択します。
3. [支払い設定] セクションで、現在の支払いオプションを表示します。支払いスケジュールを日次に変更するには、[毎日] を選択し、[送信] をクリックします。更新が完了するまでの銀行情報更新の進捗状況を示すパーセンテージバーが表示されます。
4. 支払いを日次から月次に変更するには、[毎月] を選択し、支払いを処理する日を 1~28 の範囲から選択します。[送信] を選択します。

Note

販売者は、支払い設定オプションにアクセスするため、きめ細かな IAM アクセス許可を受ける必要があります。きめ細かい IAM アクセス許可を受けるには、「[the section called “AWS Marketplace 販売者のポリシーとアクセス許可”](#)」を参照してください。

AWS は、ユーザーに代わって請求メカニズムとして機能します。購入者が利用できる最も一般的な支払い方法は、クレジットカードによるものと請求書によるものの 2 つです。

以下は、AWS Marketplace サブスクリプションの請求に関する情報です。

- 前払いによる購入は、サブスクリプション購入後すぐに請求されます。
- プライベートオファターの請求スケジュールは、購入者と販売者の間で合意されます。
- 請求書の支払い条件 (請求書の期日を含む) は、購入者と AWS の間で合意されます。条件はベンダーには開示されません。
- フレキシブルな支払いスケジュールを使用したプライベートオファターは、支払いオプションとして請求書に記載する必要があります。
- [月別請求済み収益レポート](#) を使用して請求書を検証することができます。お客様に代わって AWS が請求をこのレポートにまとめています。このレポートには、請求書作成日と請求書期日を照合して確認できるトランザクション参照キーが含まれています。

以下は、販売者としてどのように支払いを受けるかについての情報です。

- 支払いには、有効な [支払い方法](#)、[登録された米国の銀行口座](#)、および提出された W9 フォームが必要です。

- 有料製品の販売者は、W-8、付加価値税 (VAT) または商品サービス税 (GST) 登録番号、および米国の銀行口座を提供する必要があります。 [Hyperwallet](#) は、支払い AWS Marketplace のために提供できる米国の銀行口座を提供できます AWS Marketplace。
- AWS は、以下の方法で支払いを行います。
 - 毎日 - 支払いが可能になったときに日単位で支払いが行われます。支払いを受けるには、販売者の残高がプラスである必要があります。
 - 毎月 - 販売者は支払いを受ける日 (1~28 日) を選択します。支払い日は毎月同じです。 [支払いレポート](#) に支払い日が表示されます。
- AWS は、購入者が請求書の支払いを行った後、Automated Clearinghouse (ACH) 転送を使用して支払いを行います。
- 資金は、顧客から回収された後にのみ支払われます。
- 支払いが販売者の銀行に届くまでに支払い日から約 1~2 営業日かかります。正確なタイミングは、銀行とタイムゾーンによって異なります。
- 支払いレポートは、支払い後 AWS Marketplace 管理ポータル 3~5 日以内に更新されます。
- 支払われた資金と未回収の資金に関する詳細は、未払いの売掛金を含め、支払いレポートに記載されています。

すでに販売者ですか？

で利用できる go-to-market アクティビティを活用して、製品を増分チャネル収益に管理します [AWS Marketplace 管理ポータル](#)。活動は次のとおりです。

- キャンペーンでもたらされた使用量や収益の増加など、マーケティング活動の結果を数時間で測定します。
- カスタマーサービス担当者がリアルタイムで顧客データを取得できるようにします。
- 製品を作成して管理するために必要なファイルをアップロードし、処理の進行状況をモニタリングします。

苦情処理ポリシー - Amazon Payments Europe

Amazon Payments Europe (APE) が提供するサービスに問題がある場合は、お知らせください。お客様からのフィードバックは、お客様だけでなく、購入者と販売者全員にとってより良い体験を提供するのに役立ちます。

Note

に固有の苦情のみが、次の手順で対処 AWS Marketplace されます。Amazon Payments Europe S.C.A. が提供するサービスには、支払い取引の処理、手数料に発生する可能性のあるエラーの検証、資金の支払いなどが含まれます。

苦情の申し立て

Amazon Payments Europe S.C.A. の AWS Marketplace アカウントをお持ちの場合、苦情は Amazon Payments Europe S.C.A. によって処理されます。

苦情を申し立てるには

1. [AWS Marketplace](#) 販売者アカウントにサインインします。
2. [お問い合わせ] にアクセスします。
3. [コマースマーケットプレイス]、[販売者アカウント]、[登録] をクリックします。
4. 苦情の詳細を入力し、[送信] をクリックします。

Amazon Payments Europe の苦情解決期間

Amazon Payments Europe S.C.A. (APE) は、苦情を受領した日から 15 営業日以内に回答します。APE の管理が及ばない例外的な状況の場合、苦情の解決に、APE が苦情を最初に受領した日から最大 35 営業日かかる場合があります。

苦情のエスカレーション

対応にご満足いただけない場合は、以下の連絡先に苦情を申し立てることができます。

- Amazon Payments Europe 上級管理職

苦情は、上級管理職 <ape-management@amazon.lu> 宛てにメールを送信してください。お客様のコメントを慎重に検討し、上級管理職が苦情を受領した日から 15 営業日以内に回答します。Amazon Payment Europe の管理が及ばない例外的な状況では、苦情の解決に、上級管理職が最初に苦情を受けた日から最大 35 営業日かかる場合があります。

- Commission de Surveillance du Secteur Financier (CSSF)

ルクセンブルク所在の CSSF は、金融部門の企業を慎重に監督する機関です。110 Route d'Arlon L-2991 ルクセンブルクにある CSSF に連絡するか、<https://www.cssf.lu/contacts/> の [連絡先ページ] をご利用ください。CSSF に関する詳細情報と連絡方法については、CSSF ウェブサイトの「[Customer complaints](#)」を参照してください。

- オンライン紛争解決

EU においてオンラインでアカウントを開設した場合、オンライン紛争解決プラットフォームを使用して CSSF に苦情を申し立てることもできます。このオプションが利用できるのは、Amazon Payments Europe S.C.A. が金融サービスを提供しており、CSSF がそのライセンスを担当する機関だからです。詳細については、欧州委員会ウェブサイトの [Online Dispute Resolution](#) プラットフォームを参照してください。

出品料金

AWS Marketplace では、製品の次の出品料金を提供しています。

Note

これらの出品料金は、2024 年 1 月 5 日午前 0 時 (UTC) から適用されます。

パブリックオファターの出品料金

ソフトウェアとデータのパブリックオファターの出品料金は、デプロイ方法によって決まります。

- Software-as-a-service (SaaS) – 3%
- サーバー (Amazon マシンイメージ (AMI)、コンテナ、機械学習) — 20%
- AWS Data Exchange – 3%

プライベートオファターの出品料金

プライベートオファターの出品料金は、契約総額と、プライベートオファターが以前のプライベートオファターからの更新、または AWS Marketplace 以外の契約からの更新であるかによって決まります。

- 100 万 USD 未満 — 3%
- 100 万 USD 以上 1,000 万 USD 未満 — 2%

- 1,000 万 USD 以上 — 1.5%
- すべての更新 — 1.5%

チャンネルパートナーのプライベートオファー (CPPO) の出品料金

CPPO 製品は、オファータイプやデプロイ方法にかかわらず、出品料金が 0.5% 上乗せされます。例えば、その製品が SaaS プライベートオファーで、契約総額が 100 万 USD 未満の場合、出品料金は 3.5% になります。

プロフェッショナルサービスの出品料金

すべてのプロフェッショナルサービスには、プライベートオファーの 2.5% の出品料金がかかります。

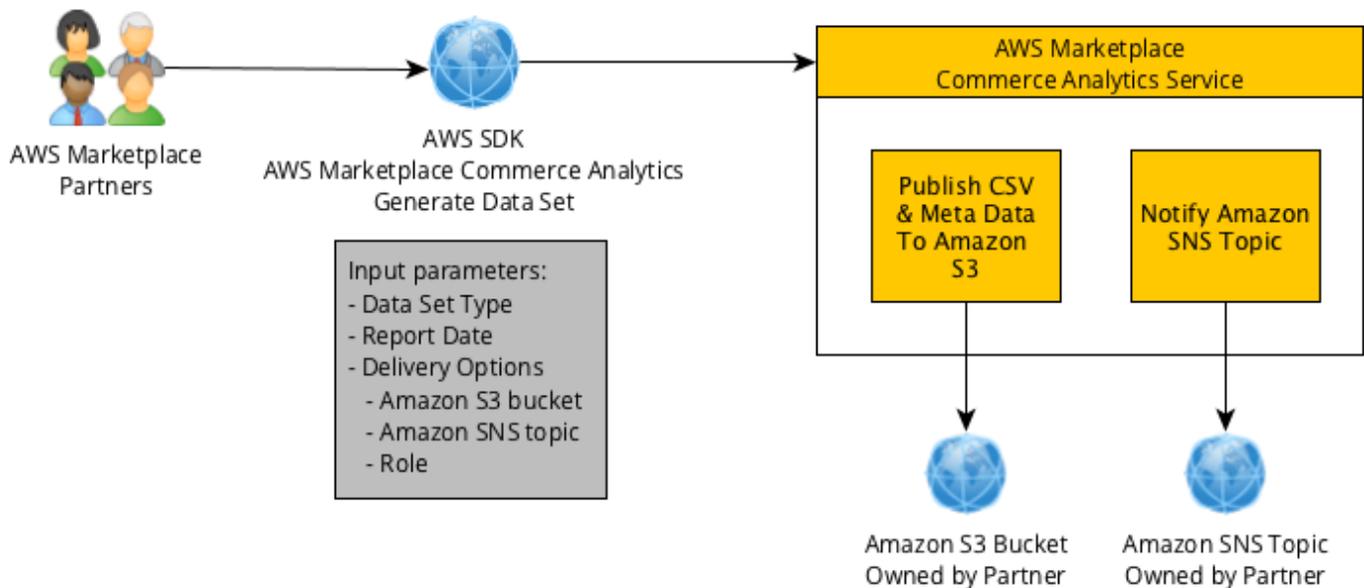
販売者ツールキット

[AWS Marketplace 管理ポータル](#) は、で製品を販売するための主要なツールです AWS Marketplace。以下の追加のツールは、顧客ベースのニーズをより詳しく分析し、販売の理解を深めるのに役立ちます。

- [AWS Marketplace コマース分析サービス](#)
- [AWS Marketplace フィールドデモンストレーションプログラム](#)
- [出品者レポート、データフィード、ダッシュボード](#)
- [のその他のリソース AWS Marketplace 管理ポータル](#)

AWS Marketplace コマース分析サービス

AWS Marketplace Commerce Analytics Service を使用すると、を通じて製品と顧客データにプログラムでアクセスできます AWS Marketplace。サービスに登録すると、AWS SDK から使用状況、サブスクリプション、請求レポートにアクセスできます。



SDK ツールを使用してリクエストしたデータは、データセット AWS アカウント として に配信されます。ほとんどのデータセットは、[AWS Marketplace 管理ポータル](#) で利用可能なテキストベースのレポートと同じデータに対応します。特定の日付のデータセットをリクエストできます。データは指定先の Amazon S3 バケットに配信されます。データ配信の通知は、Amazon Simple Notification Service (Amazon SNS) によって提供されます。

利用規約

これらの AWS Marketplace Commerce Analytics Service 利用規約 (以下「CAS 条件」) には、AWS Marketplace Commerce Analytics Service (「CA Service」) の使用とアクセスに固有の条件が含まれており、これらの CAS 条件に表示された「I Accept」ボタンまたはチェックボックスをクリックした日付、または以前に CA サービスを使用する日付に有効になります。本 CAS 条件は、お客様と Amazon Web Services, Inc の間の AWS Marketplace 販売者向け利用規約 (AWS Marketplace 「販売者条件」) の補足です。(「AWS」、「we」、「us」、または「our」)、その条件はここに組み込まれます。これらの CAS 規約と AWS Marketplace 販売者規約の間に競合が発生した場合は、当該の競合の範囲内で、およびお客様の CA サービスの使用に関してのみ、これらの CAS 規約が適用されます。大文字で表記された用語は、本書で使用されているが、本書で定義されていない用語は、AWS Marketplace 販売者条件に規定された意味を持つものとします。

1. CA サービスおよび CAS データ。CA サービスへのアクセス資格を得るには、既存の AWS Marketplace 販売者条件に拘束されている AWS Marketplace 販売者である必要があります。CA サービス (「CAS Data」) に関連して受け取った、またはアクセスできる情報およびデータは、サブスクライバー情報であり、AWS Marketplace 販売者条件に規定された制限および義務の

対象となります。(a) CAS データを第三者に開示しないことを条件として、CAS データを機密ベースで使用して、お客様の AWS Marketplace コンテンツに関連するマーケティングやその他のプロモーション活動を改善およびターゲットにすることができます。(b) CAS データを、適用されるプライバシーポリシーまたは法律と矛盾する形で使用すること。(c) サブスクライバーに連絡して、の外部で代替購入を行うよう影響を与える AWS Marketplace。(d) わたしたちを誹謗中傷する 関連会社 またはそれらもしくは当社のそれぞれの製品。または (e) 目的の受信者が AWS Marketplace サブスクライバーであることに基づいて、あらゆる種類のターゲット通信。

2. CA サービスの制限とセキュリティ。CA サービスドキュメントに記載されている方法によってのみ、CA サービスにアクセスします (またはアクセスを試みます)。CA サービスを使用中に自身のアイデンティティまたは顧客のアイデンティティを偽ったり、隠したりしてはなりません。いずれかの期間において CA サービスへのアクセスを許可された接続、通話、またはサーバーの数に関して、これを含みますが限定されず、CA サービスに使用を設定または制限する権利を単独裁量にて保有します。当該の制限に同意し、迂回することを回避しようとしません。これらの CAS 規約に違反している、または CA サービスを悪用していると確信した場合、CA サービスにアクセスする権利を制限、停止または終了する権利を保有します。
3. CA サービス認証情報の機密性とセキュリティ。CA サービス認証情報 (パスワード、キー、およびクライアント ID など) は、お客様が API クライアントを識別するために使用されることが目的です。認証情報を秘匿する全責任はお客様が負い、少なくとも同様の性質の機密情報を保護するために取る措置を含め、当該認証情報の開示、流布、または不正使用を避けるための合理的な措置を取ります。CA サービス認証情報をオープンソースプロジェクトに埋め込むことはできません。お客様の認証情報での CA サービスへのあらゆるアクセスの全責任はお客様が負います。
4. 変更。AWS は、AWS サイトに改訂されたバージョンを投稿するか、AWS Marketplace 販売者規約に従ってお客様に通知することで、いつでも本 CAS 規約を変更できます。変更された規約は、掲載時に、または、E メールによって通知した場合、E メールメッセージに記載されているとおりに、発行します。これらの CAS 規約への変更の発行日後に CA サービスの使用またはアクセスを継続することで、変更された規約に縛られることに同意したものと見なされます。
5. ターミネーション。これらの CAS 規約およびここに付与される CAS データを使用する権限は、あらゆる理由による AWS Marketplace 販売者規約の終了により、通知の有無に関わらず、終了します。さらに、あらゆるときにあらゆる理由で CA サービスの提供を停止するか、CA サービスへのアクセスを終了する場合があります。

オンボーディングガイド

AWS Marketplace Commerce Analytics Service AWS のサービスを使用するには、AWS アカウントとを設定する必要があります。

AWS Marketplace Commerce Analytics Service を使用するには

1. [アクセス許可 AWS アカウント を使用して をセットアップする](#).
2. [送信先の Amazon S3 バケットを作成する](#).
3. [レスポンス通知用の Amazon SNS トピックを設定する](#).
4. [Commerce Analytics Service プログラムに登録する](#).
5. [設定を確認する](#).

アクセス許可 AWS アカウント を使用して をセットアップする

では、ルートアカウントの認証情報を使用するの AWS Marketplace 管理ポータルではなく、AWS Identity and Access Management (IAM) ロールを使用して にサインインすることをAWS Marketplace 強くお勧めします。AWS Marketplace Commerce Analytics Service のアクセス許可の、特定の IAM のアクセス許可については、「[the section called “AWS Marketplace 販売者のポリシーとアクセス許可”](#)」を参照してください。お客様のアカウントにアクセスする人に対して個別ユーザーを作成することにより、ユーザーそれぞれに一意の認証情報を設定することができます。さらに、各ユーザーにそれぞれ異なるアクセス権限を付与することもできます。必要な場合には、いつでもユーザーのアクセス許可を変更、または無効にすることができます。

送信先の Amazon S3 バケットを作成する

Commerce Analytics Service により、リクエストしたデータが、指定した Amazon S3 バケットに配信されます。既に Amazon S3 バケットをお持ちで使用できる場合は、次のステップに進みます。

Amazon S3 バケットがない場合、このデータ専用の Amazon S3 バケットを作成する場合は、「[S3 バケットを作成する方法](#)」を参照してください。

レスポンス通知用の Amazon SNS トピックを設定する

Commerce Analytics Service は、Amazon SNS を使用してレスポンス通知を配信します。このサービスでは、データセットが利用可能になったり、エラーが発生したりしたときに、このトピックにメッセージを発行して通知します。既にこの目的で Amazon SNS トピックをお持ちの場合は、次のステップに進みます。

このサービスに対して Amazon SNS トピックを設定していない場合は、この時点で設定します。手順については、「[トピックの作成](#)」を参照してください。

作成したトピックの Amazon リソースネーム (ARN) をメモしておきます。サービスを呼び出すときに ARN が必要になります。

Commerce Analytics Service プログラムに登録する

トピックの ARN およびバケットの名前を使用してサービスを設定すると、Commerce Analytics Service は Amazon S3 バケットと Amazon SNS トピックにアクセスします。

そのアクセスを有効にするには

1. AWS Marketplace 製品の管理 AWS アカウント に使用する [AWS Marketplace 管理ポータル](#) を使用して にログインします。
2. AWS Marketplace Commerce Analytics Service に登録するために [必要な IAM 権限](#) があることを確認してください。
3. [Commerce Analytics Service 登録ページ](#) に移動します。
4. Amazon S3 バケット名および Amazon SNS トピック ARN を入力し、[登録] を選択します。
5. [permissions (権限)] ページの [Allow (許可)] を選択します。
6. で AWS Marketplace 管理ポータル、成功メッセージにロール名 ARN を記録します。サービスを呼び出すには ARN が必要です。

Note

Commerce Analytics Service へのオンボーディングにより、に IAM ロールが作成されます AWS アカウント。IAM ロールは、AWS Marketplace が Amazon S3 バケットに書き込み、Amazon SNS トピックに通知を発行できるようにします。AWS Marketplace はアカウント 452565589796 を使用して、この IAM ロールでこれらの関連アクションを実行します。

設定を確認する

最後のステップは、設定が正常に動作しているかどうかを確認することです。

設定をテストするには

1. [AWS Command Line Interface](#) (AWS CLI) をダウンロードして、インストールおよび設定します。
2. を使用して AWS CLI、このコマンドを実行します。

```
aws marketplacecommerceanalytics generate-data-set \
```

```
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \  
--data-set-publication-date "{TODAY'S-DATE}" \  
--role-name-arn "{YOUR-ROLE-NAME-ARN}" \  
--destination-s3-bucket-name "{YOUR-S3-BUCKET}" \  
--destination-s3-prefix "test-prefix" \  
--sns-topic-arn "{YOUR-SNS-TOPIC-ARN}"
```

- --data-set-publication-date で、{TODAY'S DATE} を ISO-8601 形式を使用して現在の日付 YYYY-MM-DDT00:00:00Z に置き換えます。YYYY は 4 桁の年、MM は 2 桁の月、DD は 2 桁の日を表します。
- --role-name-arn で、{YOUR-ROLE-NAME-ARN} を [Commerce Analytics Service プログラムに登録する](#) の登録プロセスで取得したロールの ARN に置き換えます。
- --destination-s3-bucket-name では、{YOUR-S3-BUCKET} を、[送信先の Amazon S3 バケットを作成する](#) で作成した Amazon S3 バケットに置き換えてください。
- --sns-topic-arn では、{YOUR-SNS-TOPIC-ARN} を、[レスポンス通知用の Amazon SNS トピックを設定する](#) で作成した Amazon SNS トピックに置き換えます。

サービスからデータ SetRequestId レスポンスを含むレスポンスを受け取った場合は、オンボーディングプロセスが完了しました。正常な応答は次のようになります。

```
{  
  "dataSetRequestId": "646dd4ed-6806-11e5-a6d8-fd5dbcaa74ab"  
}
```

技術的な実装ガイド

AWS Marketplace Commerce Analytics Service は [AWS SDK](#) を通じて提供されます。このガイドでは、[AWS CLI](#) および [AWS SDK for Java](#) を使用してサービス进行操作する方法を説明します。

Commerce Analytics Service の IAM ポリシー

ユーザーが Commerce Analytics Service を使用できるようにするには、以下の権限が必要です。

AWS Marketplace Commerce Analytics Service に登録するには、次の IAM アクセス許可ポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles",
        "iam:CreateRole",
        "iam:CreatePolicy",
        "iam:AttachRolePolicy",
        "aws-marketplace-management:viewReports"
      ],
      "Resource": "*"
    }
  ]
}
```

以下の IAM アクセス許可ポリシーを使用して、ユーザーが AWS Marketplace Commerce Analytics Service にリクエストを送信できるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

詳細については、「IAM ユーザーガイド」の「[IAM コンソールでのポリシーの作成](#)」を参照してください。

を使用したリクエストの実行 AWS CLI

開始するには、[AWS CLI](#)をダウンロードします。次の AWS CLI 例では、2017 年 10 月 1 日の時間単位/月単位のサブスクリプションデータセットをリクエストします。このデータセットはプレフィックス demo-prefix を使用して Amazon S3 バケット demo-bucket に発行され、通知メッセージは Amazon SNS トピック demo-topic に配信されます。

```
aws marketplacecommerceanalytics generate-data-set \  
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \  
--data-set-publication-date "2017-10-01T00:00:00Z" \  
--role-name-arn "arn:aws:iam::123412341234:role/MarketplaceCommerceAnalyticsRole" \  
--destination-s3-bucket-name "demo-bucket" \  
--destination-s3-prefix "demo-prefix" \  
--sns-topic-arn "arn:aws:sns:us-west-2:123412341234:demo-topic"
```

このリクエストは、リクエストごとに一意の識別子を返します。この識別子を使用して、リクエストを Amazon SNS トピックに発行された通知に関連付けることができます。以下は、この識別子の例です。

```
{  
  "dataSetRequestId": "646dd4ed-6806-11e5-a6d8-fd5dbcaa74ab"  
}
```

AWS SDK for Javaによるリクエストの作成

開始するには、[AWS Java SDK](#) をダウンロードします。次の AWS SDK for Java 例では、2015 年 10 月 1 日の時間単位/月単位のサブスクリプションデータセットをリクエストします。このデータセットはプレフィックス demo-prefix を使用して Amazon S3 バケット demo-bucket に発行され、通知メッセージは Amazon SNS トピック demo-topic に配信されます。

```
/*  
 * Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License").  
 * You may not use this file except in compliance with the License.  
 * A copy of the License is located at  
 *  
 * http://aws.amazon.com/apache2.0  
 *  
 * or in the "license" file accompanying this file. This file is distributed  
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
 * express or implied. See the License for the specific language governing  
 * permissions and limitations under the License.  
 */  
import java.text.DateFormat;
```

```
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import com.amazonaws.AmazonClientException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import
    com.amazonaws.services.marketplacecommerceanalytics.AWSMarketplaceCommerceAnalyticsClient;
import
    com.amazonaws.services.marketplacecommerceanalytics.model.GenerateDataSetRequest;
import com.amazonaws.services.marketplacecommerceanalytics.model.GenerateDataSetResult;
/**
 * This sample demonstrates how to make basic requests to the AWS Marketplace Commerce
 * Analytics service using the AWS SDK for Java.
 * <p>
 * <b>Prerequisites:</b> Follow the on-boarding guide: {URL OR SOMETHING}
 * <p>
 * Fill in your AWS access credentials in the provided credentials file
 * template, and be sure to move the file to the default location
 * (~/.aws/credentials) where the sample code will load the credentials from.
 * <p>
 * <b>WARNING:</b> To avoid accidental leakage of your credentials, DO NOT keep
 * the credentials file in your source directory.
 * <p>
 * http://aws.amazon.com/security-credentials
 */
public class MarketplaceCommerceAnalyticsSample {
    public static void main(String[] args) throws ParseException {
        /**
         * The ProfileCredentialsProvider will return your [default]
         * credential profile by reading from the credentials file located at
         * (~/.aws/credentials).
         */
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException("Cannot load the credentials from the credential
                profiles "
                + "file. Make sure that your credentials file is at the correct "
```

```
+ "location (~/.aws/credentials), and is in valid
format.", e);
}
AWSMarketplaceCommerceAnalyticsClient client = new
    AWSMarketplaceCommerceAnalyticsClient(credentials);
Region usEast1 = Region.getRegion(Regions.US_EAST_1);
client.setRegion(usEast1);
System.out.println("=====");
System.out.println("Getting Started with AWS Marketplace Commerce Analytics Service");
System.out.println("=====
\n");
// Create a data set request with the desired parameters
GenerateDataSetRequest request = new GenerateDataSetRequest();
request.setDataSetType("customer_subscriber_hourly_monthly_subscriptions");
request.setDataSetPublicationDate(convertIso8601StringToDateUtc("2014-06-09T00:00:00Z"));
request.setRoleNameArn("arn:aws:iam::864545609859:role/
MarketplaceCommerceAnalyticsRole");
request.setDestinationS3BucketName("awsmp-goldmine-seller");
request.setDestinationS3Prefix("java-sdk-test");
request.setSnsTopicArn("arn:aws:sns:us-west-2:864545609859:awsmp-goldmine-seller-
topic");
System.out.println(
String.format("Creating a request for data set %s for publication date %s.",
request.getDataSetType(), request.getDataSetPublicationDate()));
try {
// Make the request to the service
GenerateDataSetResult result = client.generateDataSet(request);
// The Data Set Request ID is a unique identifier that you can use to correlate the
// request with responses on your Amazon SNS topic
System.out.println("Request successful, unique ID: " + result.getDataSetRequestId());
} catch (AmazonServiceException ase) {
System.out.println("Caught an AmazonServiceException, which means your request made it
"
+ "to the AWS Marketplace Commerce Analytics service, but was rejected with an "
+ "error response for some reason.");
System.out.println("Error Message: " + ase.getMessage());
System.out.println("HTTP Status Code: " + ase.getStatusCode());
System.out.println("AWS Error Code: " + ase.getErrorCode());
System.out.println("Error Type: " + ase.getErrorType());
System.out.println("Request ID: " + ase.getRequestId());
} catch (AmazonClientException ace) {
System.out.println("Caught an AmazonClientException, which means the client encountered
"
+ "a serious internal problem while trying to communicate with the AWS Marketplace"
```

```

+ "Commerce Analytics service, such as not being able to access the "
+ "network.");
System.out.println("Error Message: " + ace.getMessage());
}
}
private static Date convertIso8601StringToDateUtc(String dateIso8601) throws
    ParseException {
    TimeZone utcTimeZone = TimeZone.getTimeZone("UTC");
    DateFormat utcDateFormat = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ssX");
    utcDateFormat.setTimeZone(utcTimeZone);
    return utcDateFormat.parse(dateIso8601);
}
}

```

正常な結果はこの例に似たものとなります。

```

=====
Getting Started with AWS Marketplace Commerce Analytics Service
=====
Creating a request for data set customer_subscriber_hourly_monthly_subscriptions for
publication
date Sun Jun 08 17:00:00 PDT 2014.
Request successful, unique ID: c59aff81-6875-11e5-a6d8-fd5dbcaa74ab

```

技術ドキュメント

このサービスでは、1つのメソッド `GenerateDataSet` が公開されます。このメソッドを使用すると、Amazon S3 バケットに発行するデータセットをリクエストできます。次の表に `GenerateDataSet` のパラメータの一覧を示します。

データセットのパラメータ

フィールド	説明
Data Set Type	このデータセットはリクエストの結果として返されます。
Data Set Publication Date	データセットの発行日。

フィールド	説明
	<p>日別データセットでは、該当日の日付が日単位の粒度で提供されます。</p> <p>月別データセットでは、該当月の日付が月単位の粒度で提供されます。日付の値は無視されます。</p>
Role Name ARN	<p>ロールの ARN の後に、リソースへのアクセス権を持つサービスを提供するアクセス権限ポリシーが続いています。</p>
Destination Amazon S3 Bucket Name	<p>送信先 Amazon S3 バケットの名前 (ARN ではなくフレンドリ名)。データセットはこの場所に発行されます。</p>
Destination Amazon S3 Prefix	<p>(オプション) データセットの発行先である Amazon S3 のプレフィックス。標準ファイルシステムのディレクトリパスに似ています。</p> <p>たとえば、バケットの名前が mybucket であり、プレフィックスが myprefix/mydatasets の場合、出力ファイルは s3://DOC-EXAMPLE-BUCKET/myprefix/mydatasets/outputfile に発行されます。</p> <p>プレフィックスのディレクトリ構造が存在しない場合は作成されます。</p> <p>プレフィックスを指定しないと、データセットの発行先は Amazon S3 バケットのルートになります。</p>
SNS トピックの ARN	<p>データセットの発行時またはエラーの発生時に通知先となる Amazon SNS トピックの ARN。</p>

レスポンス

AWS Marketplace Commerce Analytics Service は 2 つのレスポンスを返します。1 つ目は同期で即座に返され、2 つ目は非同期でこれは Amazon SNS を使用して返されます。同期のレスポンスは次の例のようになります。

データセットパラメータ

フィールド	説明
Data Set Request ID	サービスに対する特定のリクエストを表す一意の識別子。この識別子を使用して、リクエストを Amazon SNS トピックへの通知と関連付けることができます。

非同期レスポンスが JSON 形式のドキュメントとして Amazon SNS トピックに書き込まれ、次の例のようになります。

データセットのパラメータ

フィールド	説明
Data Set S3 Location	配信されたデータセットのバケット名とキー。
Data Set Meta Data S3 Location	配信されたデータセットのメタデータファイルのバケット名とキー。
Data Set Request ID	サービスに対する特定のリクエストを表す一意の識別子。この識別子を使用して、リクエストを Amazon SNS トピックへの通知と関連付けることができます。
成功	オペレーションが成功した場合は「true」、それ以外の場合は「false」。
メッセージ	(オプション) エラーが発生した場合 (「Success」が「false」の場合など)、このメッセージは失敗に関する情報を示します。

JSON 形式の非同期レスポンスの例

```
{
  "dataSetS3Location":{
    "bucketName":"demo-bucket",
    "key":"demo-prefix/
customer_subscriber_hourly_monthly_subscriptions_2014-06-09.csv"
  },
  "dataSetMetaDataS3Location":{
    "bucketName":"demo-bucket",
    "key":"demo-prefix/
customer_subscriber_hourly_monthly_subscriptions_2014-06-09.meta.json"
  },
  "dataSetRequestId":"f65b7244-6862-11e5-80e2-c5127e17c023",
  "success":true
}
```

出力

リクエストが成功すると、リクエストしたデータセットは .csv ファイルとして Amazon S3 バケツに配信されます。JSON 形式のメタデータファイルは、データセットファイルと同じ場所に発行されます。メタデータファイルは、データセットや元のリクエストのパラメータに関する有用な情報を提供します。メタデータファイル名は、データセットファイル名と同じですが、拡張子が .meta.json となります。次の表は、.csv ファイルのメタデータフィールドの一覧です。

メタデータフィールド

フィールド	説明
Data Set Request ID	サービスに対する特定のリクエストを表す一意の識別子。この識別子を使用して、リクエストを Amazon SNS トピックへの通知と関連付けることができます。
Data Set Coverage Range	データ範囲の開始日時と終了日時を定義します。これらの日時は ISO 8601 形式です。
Data Set Request Parameters	GenerateDataSet メソッドへの元のリクエストパラメータ。

フィールド	説明
Data Set S3 Location	配信されたデータセットのバケット名とキー。
Data Set Meta Data S3 Location	配信されたデータセットのメタデータファイルのバケット名とキー。

JSON 形式のメタデータコンテンツの例を次に示します。

```
{
  "dataSetRequestId": "43d7137b-8a94-4042-a09d-c41e87f371c1",
  "dataSetCoverageRange": {
    "startDateTime": "2014-06-08T00:00:00.000Z",
    "endDateTime": "2014-06-08T23:59:59.000Z"
  },
  "dataSetRequestParameters": {
    "sellerAccountId": "123412341234",
    "dataSetType": "customer_subscriber_hourly_monthly_subscriptions",
    "dataSetPublicationDate": "2014-06-09T00:00:00.000Z",
    "roleNameArn": "arn:aws:iam::123412341234:role/MarketplaceCommerceAnalyticsRole",
    "destinationS3BucketName": "demo-bucket",
    "destinationS3Prefix": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions",
    "snsTopicArn": "arn:aws:sns:us-west-2:123412341234:demo-topic"
  },
  "dataSetS3Location": {
    "bucketName": "demo-bucket",
    "key": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions_2014-06-09.csv"
  },
  "dataSetMetaDataSetS3Location": {
    "bucketName": "demo-bucket",
    "key": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions_2014-06-09.meta.json"
  }
}
```

利用可能な日付も含む利用可能なデータセットの詳細な一覧については、[AWS SDK ドキュメント](#)を参照してください。

トラブルシューティング

このセクションでは、AWS Marketplace Commerce Analytics Service の使用に伴って発生する可能性のある問題の解決策について説明します。

許可リストの問題のため、サービスにアクセスできません。

で販売者としてまだ登録されていない場合は AWS Marketplace、[AWS Marketplace 管理ポータル](#)にアクセスして登録してください。で販売者として登録済みの場合は AWS Marketplace、[AWS Marketplace 販売者オペレーション](#)チームにお問い合わせください。

過去の日付のデータセットをリクエストできません。ただし、SDK ドキュメントでは、この日付のデータが利用可能となっています。

過去の特定の日付に関するデータセットが利用可能であると記載されていても、利用できるのは AWS Marketplaceに参加した時点以降のデータのみです。エラーが発生したと思われる場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

このサービスを呼び出すと、「Could not connect to the endpoint URL: https://marketplacecommerceanalytics.eu-central-1.amazonaws.com/」というエラーメッセージが表示されます。

AWS Marketplace Commerce Analytics Service は、米国東部 (バージニア北部) リージョンでのみ利用できます。Commerce Analytics Service へのすべての呼び出しは us-east-1 エンドポイントに対して行う必要があります。

を使用している場合は AWS CLI、次の例に示すように、「」フラグを各呼び出しに追加--region us-east-1、を AWS リージョンとして指定します。

```
aws marketplacecommerceanalytics generate-data-set \  
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \  
--data-set-publication-date "2016-04-21T00:00:00Z" \  
--role-name-arn "arn:aws:iam::138136086619:role/MarketplaceCommerceAnalyticsRole" \  
--destination-s3-bucket-name "marketplace-analytics-service" \  
--destination-s3-prefix "test-prefix" \  
--sns-topic-arn "arn:aws:sns:eu-  
central-1:138136086619:Marketplace_Analytics_Service_Notice" \  
--region us-east-1
```

オンボーディングプロセスで選択したのとは別の Amazon S3 バケットまたは別の Amazon SNS トピックを使用するにはどうすればいいですか。

AWS Marketplace Commerce Analytics Service に登録するときに、Amazon S3 バケットと Amazon SNS トピックを指定しました。オンボーディングプロセスでは、サービスがこれらの特定のリソースにのみアクセスできるように IAM 権限を設定します。別のリソースを使用するには、IAM ポリシーを変更する必要があります。

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. IAM コンソールの左側にある [ロール] を選択します。
3. を選択します MarketplaceCommerceAnalyticsRole。
4. [Inline Roles (インラインロール)] セクションが展開されていない場合は展開します。
5. oneClick_MarketplaceCommerceAnalyticsRole で始まる名前のポリシーを探し、[Edit Policy] を選択します。
6. そのポリシードキュメントで、変更するサービスに関連するアクションが指定されているセクションを見つけます。例えば、Amazon S3 バケットを変更するには、s3: で始まるアクションが含まれているセクションを見つけ、各リソースの選択肢を新しい Amazon S3 バケットに変更します。

IAM ポリシーの詳細については、https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html のガイドを参照してください。

GenerateDataSet アクションを呼び出すと、**AccessDeniedException** エラーが表示される

これは、GenerateDataSet を呼び出すために必要なアクセス許可がユーザーにない場合に発生することがあります。以下の手順は、IAM コンソールを使用してこれらの権限を持つ IAM ポリシーを作成し、ユーザー、グループ、またはロールにアクセス許可を追加するために必要な手順の概要を示しています。

JSON ポリシーエディタでポリシーを作成するには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. 左側のナビゲーションペインで、[ポリシー] を選択します。

初めて [ポリシー] を選択する場合には、[管理ポリシーによるこそ] ページが表示されます。[今すぐ始める] を選択します。

3. ページの上部で、[ポリシーを作成] を選択します。
4. [ポリシーエディタ] セクションで、[JSON] オプションを選択します。
5. 次の JSON ポリシードキュメントを入力します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

6. [次へ] をクリックします。

Note

いつでも [Visual] と [JSON] エディタオプションを切り替えることができます。ただし、[Visual] エディタで [次] に変更または選択した場合、IAM はポリシーを再構成して visual エディタに合わせて最適化することがあります。詳細については、「IAM ユーザーガイド」の「[ポリシーの再構成](#)」を参照してください。

7. [確認と作成] ページで、作成するポリシーの [ポリシー名] と [説明] (オプション) を入力します。[このポリシーで定義されているアクセス許可] を確認して、ポリシーによって付与されたアクセス許可を確認します。
8. [ポリシーの作成] をクリックして、新しいポリシーを保存します。

アクセス権限を付与するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

- のユーザーとグループ AWS IAM Identity Center :

アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「[権限設定を作成する](#)」の手順に従ってください。

- IAM 内で、ID プロバイダーによって管理されているユーザー:

ID フェデレーションのロールを作成します。詳細については、「IAM ユーザーガイド」の「[サードパーティー ID プロバイダー \(フェデレーション\) 用のロールの作成](#)」を参照してください。

- IAM ユーザー:
 - ユーザーが担当できるロールを作成します。手順については、「IAM ユーザーガイド」の「[IAM ユーザー用ロールの作成](#)」を参照してください。
 - (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加する。詳細については、「IAM ユーザーガイド」の「[ユーザー \(コンソール\) へのアクセス権限の追加](#)」を参照してください。

ここに記載されていない問題。

[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

AWS Marketplace フィールドデモンストレーションプログラム

AWS Marketplace フィールドデモンストレーションプログラム (FDP) では、AWS フィールドチーム (社内で承認された AWS 従業員) は、を通じて一部の製品やソリューション AWS Marketplace を無料で使用できます。

承認された AWS 従業員の例には、ソリューションアーキテクト、営業およびマーケティングの専門家などがあります。FDP により、これらの従業員は、教育のための製品機能と、顧客のワークロードに組み込む可能性を実証することができます。

次の製品タイプがサポートされています。

- [Amazon マシンイメージ \(AMI\)](#)
- [コンテナ](#)
- [機械学習アルゴリズムとモデルパッケージ \(SageMaker \)](#)
- [データセット \(AWS Data Exchange\)](#)

Note

AWS Data Exchange 製品の場合、FDP は公開オファーが 0 USD (無料) の製品にのみ適用されます。

サブスクリプション検証が有効になっている AWS Data Exchange 製品の場合、プロバイダーはサブスクリプションリクエストを承認する必要があります。サブスクリプション検

証の詳細については、「AWS Data Exchange ユーザーガイド」の「[サブスクライバーのサブスクリプション検証](#)」を参照してください。

AWS Marketplace 販売者としてサインアップすると、FDP プログラムに自動的に登録されます。オプトアウトするには、[AWS Marketplace Seller Operations](#) チームにサポートリクエストを送信します。

このプログラムでの製品使用状況に関する情報は、[月別請求済み収益レポート](#)の「[AWS フィールド デモ使用量](#)」セクションを参照してください。

のその他のリソース AWS Marketplace 管理ポータル

AWS Marketplace 管理ポータルにはさらに多くのリソースがあります。[AWS Marketplace 管理ポータル](#)を開いてサインインすると、ホームページの [Marketplace リソース] セクションに追加のリソースへのリンクが表示されます。

例えば、発売前 90 日間および発売後 90 日間に製品をマーケティングするためのサポートを受けるには、Marketplace リソースの AWS Marketplace 管理ポータル ホームページからリンク [180 日間の GTM アカデミー](#)されているを確認できます。

製品の準備

で製品を公開する準備 AWS Marketplace には、パッケージの設定、料金設定スキームの設定、製品を一覧表示する関連カテゴリの決定、関連検索で製品が表示されるようにキーワードの追加が含まれます。

トピック

- [製品デリバリー](#)
- [製品の料金](#)
- [AWS Marketplace 製品のリージョンと国](#)
- [での標準化された契約 AWS Marketplace](#)
- [カテゴリとメタデータ](#)
- [AMI とコンテナ製品の使用手順](#)
- [製品の検索エンジン最適化](#)

製品デリバリー

各配信方法には、パッケージング、料金、および配信に関するオプションが複数あります。一部のメソッドは、それをサポートするプログラムに登録 AWS Marketplace するまで、で販売者として使用することはできません。

製品の作成には、標準の表示価格とエンドユーザーライセンス契約 (EULA) を使用できます。またカスタムの料金と EULA を使用して、個別の顧客向けのプライベートオファーを作成できます。契約条件に追加の変更が必要な場合は、AWS Marketplace チームと連携してプライベートオファーをカスタマイズできます。調達プロセスを簡素化するには、公開出品とプライベートオファーの両方で[標準化されたライセンス条項](#)を使用できます。

Note

AWS では、特定の販売者に [でのガイド付きデモンストレーション](#)を提供するオプションを提供しています AWS Marketplace。APN カスタマーエンゲージメント (ACE) の対象となる AWS パートナーネットワーク (APN) パートナーで、このオプションを購入者に提供する場合は、AWS 担当者に連絡して、さらなる適格性を判断してください。 <https://aws.amazon.com/partners/programs/ace/>

次の表に、ソフトウェア製品の配信に使用できる方法と、AWS Marketplace 購入者が AWS Marketplace コンソールで各タイプの配信物を見つける方法を示します。

製品配信方法

製品配信方法	コンソールの配送方法フィルター	説明
単一 AMI	[Amazon マシンイメージ (AMI)]	<p>製品用のカスタム Amazon マシンイメージ (AMI) を配信します。AMI は Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを起動するために必要な情報を提供します。</p> <p>購入者は単一 AMI を使用して、製品がインストール済みですぐに使用できる Amazon EC2 インスタンスを作成できます。</p> <p>詳細については、「AMI ベースの製品」を参照してください。</p>
AWS CloudFormation テンプレートを使用して配信される AMI	CloudFormation テンプレート	<p>CloudFormation テンプレートを使用して、AWS Marketplace 購入者に配信される AMI ベースの製品を一覧表示できます。</p> <p>購入者は、1つのソリューションを購入すると、その製品に含まれているすべての AMI に対する権利を付与されます。</p>

製品配信方法	コンソールの配送方法フィルター	説明
		<p>AMIs 「を使用した AMI ベースの配信 AWS CloudFormation」 を参照してください。 CloudFormation</p> <p>CloudFormation テンプレートの詳細については、「ユーザーガイド」の「の AWS CloudFormation 概念」を参照してください。AWS CloudFormation</p>
<p>コンテナベースの製品またはアプリケーション</p>	<p>[コンテナ]</p>	<p>コンテナイメージにパッケージされた製品を配信します。コンテナ製品は、コンテナイメージとデプロイテンプレートのセットであるオプションで構成されます。</p> <p>詳細については、「コンテナベースの製品」を参照してください。</p>
<p>データ製品</p>	<p>AWS Data Exchange</p>	<p>既存の PLF を使用するか、AWS Data Exchange を使用してデータ製品を作成します。</p> <p>AWS Data Exchange を通じてデータ製品などを公開および管理する方法については、「AWS Data Exchange ユーザーガイド」の「AWS Data Exchange でのデータ製品の提供」を参照してください。</p>

製品配信方法	コンソールの配送方法フィルター	説明
機械学習アルゴリズムとモデルパッケージ	SageMaker モデル	<p>Amazon を使用してアルゴリズムまたはモデルパッケージ SageMaker を作成し、に公開します AWS Marketplace。</p> <p>機械学習アルゴリズムとモデルパッケージの提供の詳細については、「Machine Learning 製品」を参照してください。</p> <p>の詳細については SageMaker、「Amazon SageMaker デベロッパーガイド」の SageMaker 「とは」を参照してください。</p>
Software as a service (SaaS)	SaaS	<p>サブスクリプションベース、契約ベース、または従量制の契約料金モデルで SaaS 製品を提供できます。</p> <p>詳細については、「SaaS ベースの製品」を参照してください。</p>
プロフェッショナルサービス	[プロフェッショナルサービス]	<p>他の製品をサポートしたり、他の AWS Marketplace 製品と連携したりするプロフェッショナルサービスを提供できます。</p>

製品の料金

このトピックでは、AWS Marketplaceのソフトウェア製品に関する一般的な料金情報を提供します。すべての料金は、米ドル (USD) に基づいています。

有料製品の場合、は顧客からソフトウェア料金を AWS Marketplace 収集します。

無料のソフトウェア、または無償で提供されるオープンソースのソフトウェアにはサービス料はかかりません。

返金については、「[での製品の返金 AWS Marketplace](#)」を参照してください。

トピック

- [料金モデル](#)
- [料金モデルの変更](#)
- [料金の変更](#)
- [プライベートオファー](#)
- [での製品の返金 AWS Marketplace](#)

料金モデル

次のトピックでは、AWS Marketplaceで利用可能な料金モデルに関する一般的な情報を示します。

トピック

- [年間料金](#)
- [使用料金](#)
- [契約料金](#)
- [Bring Your Own License の料金](#)

特定の製品配信方法の料金モデルについては、以下を参照してください。

- [AMI 製品の料金](#)
- [コンテナ製品の料金](#)
- [機械学習製品の料金](#)
- [SaaS 製品の料金](#)
- [プロフェッショナルサービス製品の料金](#)

年間料金

年間価格モデルでは、12 か月のサブスクリプションを購入するお客様に製品を提供できます。例えば、サブスクリプション料金は、同じ製品を 1 時間ごとに長期間実行する場合に比べて、最大 40% 節約できます。サブスクリプション契約時に全額請求されます。年間サブスクリプションがお客様に提示される方法の詳細については、「[AMI サブスクリプション](#)」または「[有料コンテナ製品の料金モデル](#)」を参照してください。

年間サブスクリプションを使用する場合の考慮事項は以下のとおりです。

- 年間料金は、インスタンスタイプごとに定義されます。すべての Amazon Elastic Compute Cloud (Amazon EC2) インスタンスタイプで同じ場合や、インスタンスタイプごとに異なる場合があります。
- すべての年間インスタンスタイプには、時間単位のインスタンスタイプも定義されている必要があります。AWS Marketplace では、同じ製品に対して年間みの料金または年間なしの時間単位は提供されません。年間料金を提供する製品では、時間単位の料金も指定する必要があります。
- 時間料金が 0 USD で、0 USD 以外の年間インスタンスタイプが定義されている場合は、特定のインスタンスタイプで年間料金を 0 USD に設定することができます。
- 年間サブスクリプション期間の終了時、時間単位の請求が開始されます。
- お客様が年間サブスクリプション X を購入したが Y インスタンスで Y ソフトウェアを実行している場合、年間サブスクリプションの対象外である (Y-X) インスタンスの時間単位のソフトウェア料金で請求されます。そのため、年間料金のインスタンスタイプには必ず、時間レートを含める必要があります。
- 販売者のプライベートオファーを使用すると、複数年 (最大 3 年) またはカスタム期間 AMI を前払い料金で提供したり、柔軟な支払いスケジュールを提供したりできます。複数年契約とカスタム期間契約の詳細については、[プライベートオファーの準備](#)および [the section called “分割プラン”](#) を参照してください。

で年間製品を提供する場合 AWS Marketplace、の「ファイルアップロードドキュメント」セクションにある年間製品の特定の返金ポリシーに同意したことになります [AWS Marketplace 管理ポータル](#)。

料金の変更

年間料金 (年額 1,000 USD など、年額 1,200 USD) は 90 日ごとに変更できます。ただし、既存のお客様には 90 日前に年間料金を通知する必要があります。新しい価格は新しいサブスクリプションに適用されますが、既存のサブスクリプションには影響しません。

Note

90 日間の通知期間中は、サポートされているインスタンスタイプを更新することはできません。

価格の変更は、自動更新日より少なくとも 90 日前に変更された場合にのみ有効です。自動更新前に、新しい価格が記載された E メールメッセージがお客様に送信されます。

エンドユーザーライセンス契約

年間サブスクリプションに基づく 12 か月間の AWS お客様のソフトウェアの使用は、 の製品の詳細ページでお客様が提供した EULA の対象となります AWS Marketplace。

使用料金

従量制料金とも呼ばれる使用量料金モデルでは、使用した分のみを支払うお客様に製品を提供できません。

販売者は、以下の使用カテゴリのいずれかを選択できます。

- [ユーザー]
- [ホスト]
- [帯域幅]
- [データ]
- [階層]
- [単位] (カスタムカテゴリ用)

また、製品に対して最大 24 の分析を定義できます。料金は、ソフトウェアが API を呼び出した際に測定され、報告されます。ユースケースにもよりますが、ベストプラクティスとして API が 1 時間に 1 回呼び出されるよう設定することをお勧めします。すべての使用量は毎月計算され、既存の AWS Marketplace ソフトウェアと同じメカニズムを使用して毎月請求されます。

AWS Marketplace Metering Service では、新しい料金シナリオをいくつか使用することができます。

Example ホスト別料金

ソフトウェアがホストをモニタリングする場合は、モニタリングされるホストごとに請求し、ホストサイズに基づき異なる料金を設定することができます。

Example ユーザー別料金

お客様のソフトウェアで、組織における複数のユーザーの使用を許可している場合は、ユーザー単位で請求することができます。1時間ごとに、プロビジョニングされたユーザーの合計数が請求されます。

Note

製品積載フォーム (PLF) では、関連する列に「FCP」(Flexible Consumption Pricing) という名前が付けられます。例: [FCP カテゴリ (カスタム料金カテゴリ)]。

AWS Marketplace Metering Service 製品の場合は、次の点に注意してください。

- ソフトウェアがすでににある場合は AWS Marketplace、代替使用ディメンションを有効にする製品を作成する必要があります。標準製品を AWS Marketplace 計測サービスを使用するように変更することはできません。新製品が公開されたら、古い製品を削除するか、いずれもウェブサイトにも維持することができます。
- AWS Marketplace Metering Service では、ソフトウェアが 1 時間ごとに使用状況を報告し、1 時間の顧客使用状況を記録する必要があります。計測サービスレコードの送信または受信に障害が発生した場合、AWS はそのような使用に対して課金できません。計測レコードを正常に受け取ったことを確認する必要があります。
- AWS Marketplace Metering Service を使用する製品では 1-Click はサポートされていません。購入者は、特定のアクセス許可とインターネットゲートウェイを持つ AWS Identity and Access Management (IAM) ロールを使用してソフトウェアを起動する必要があります。
- 無料トライアルと年間料金は、AWS Marketplace Metering Service と互換性がありません。
- ディメンション (ユーザー、ホスト、帯域幅、およびデータ) またはディメンション名の変更はサポートされていません。新しい製品を作成する必要はありません。

契約料金

契約料金モデルを使用すると、1 か月、12 か月、24 か月、または 36 か月間のライセンスを購入できるような前払い料金をお客様に提供できます。

契約料金は、以下の製品で利用できます。

- 単一の AMI ベースの製品と AWS CloudFormation テンプレートベースの製品を含む AMI。詳細については、「[AMI 製品の契約料金](#)」を参照してください。
- コンテナベースの製品 詳細については、「[コンテナ製品の契約料金](#)」を参照してください。
- Software as a service (SaaS) ベースの製品 詳細については、「[SaaS 契約の料金](#)」を参照してください。

Note

AMI およびコンテナベースの製品の契約料金は、新製品のみを対象としています。

既存の AMI またはコンテナベースの製品があり、契約料金を使用する場合は、新しいリストを作成してから、Product Load Form (PLF) を使用して契約料金モデルを適用し、AMI またはコンテナベースの製品を と統合してから AWS License Manager、AMI またはコンテナベースの製品を公開します。

顧客が契約料金を購入すると、ライセンスマネージャー API を使用してソフトウェアがチェックできるライセンス AWS アカウント が顧客 AWS Marketplace 内で によって作成されます。お客様が AMI またはコンテナベースの製品のインスタンスを起動するには IAM ロールが必要です。

Bring Your Own License の料金

AWS Marketplaceでは、自分のライセンス使用 (BYOL) 製品のサービス料はかかりません。

お客様の選択を保証するために、すべての BYOL 製品に対し有料オプションが必要です。これにより、既存のライセンスがないお客様が製品を購入して使用することもできるようになります。

BYOL 製品の場合、ソフトウェアをオンラインで購入できるようにすることは、事業内容から逸脱する企業もあると当社では認識しています。したがって、当社は、ソフトウェアには AWS Marketplaceでの購入が可能なバージョンがあるべきとする要件を、起動後 90 日間は緩和します。この間、AWS Marketplace アカウント管理チームはお客様と協力して課題に対処します。チームがソフトウェアを AWS Marketplaceで購入可能にするかどうか、またどのように購入できるかを判断するお手伝いをします。

料金モデルの変更

料金モデルの変更は、ポジティブなカスタマーエクスペリエンスとすべての関係者へのリスクを軽減するために、AWS Marketplace にとってレビューおよび承認される必要があります。料金モデルの変更を行う場合は、[AWS Marketplace Seller Operations](#) チームに連絡して相談してください。

料金モデルの変更に関するリクエストは、処理とレビューに 30〜90 日ほどかかります。

料金の変更

AWS Marketplace 管理ポータルで価格とメタデータを更新できます。

価格を変更するには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [製品] タブに、作成した現在の製品のリストが表示されます。ここで製品リストを編集したり、変更をリクエストできます。

Note

新規サブスクライバーの場合、料金変更はすぐに有効になります。既存のサブスクライバーの場合、料金変更は、料金変更通知が送信された日から始まる 90 日間の期間の翌月の初日に有効になります。例えば、3 月 16 日に料金変更通知を送信したとします。6 月 16 日は 3 月 16 日から約 90 日後です。料金変更は 90 日間の期間の翌月の初日に行われるため、変更の発効日は 7 月 1 日です。

プライベートオファー

Seller AWS Marketplace Private Offer プログラムでは、販売 AWS Marketplace 者は個々の AWS Marketplace 顧客 (購入者) とカスタム料金と EULAs を交渉できます。詳細については、「[プライベートオファーの準備](#)」を参照してください。

での製品の返金 AWS Marketplace

のすべての有料製品には AWS Marketplace、料金モデルに関係なく、ソフトウェア料金の返金ポリシーが規定されている必要があります。この返金ポリシーには、返金の条件だけでなく、販売者に連絡して返金をリクエストする方法を盛り込む必要があります。返金ポリシーの詳細を決定するのは販売者です。ただし、製品の使用に対して何らかの方法で購入者に返金することをお勧めします。お客

様は、掲載した返金ポリシーを遵守する必要があります。このトピックでは、AWS Marketplace 製品の返金リクエストのタイプ、関連するポリシーと承認プロセス、および顧客に対して返金リクエストを送信する方法について説明します。

AWS Marketplace 製品の返金リクエストタイプ

お客様は AWS Marketplace、製品のさまざまなタイプの返金をリクエストできます。で販売されている AWS Marketplace 製品については AWS、返金ポリシーページを参照し、を使用して問い合わせサポートフォームを送信してください AWS Support Center Console。お客様が から直接ソフトウェアの返金をリクエストする場合 AWS、問題の製品の投稿されたサポート連絡先情報を使用して販売者に連絡するように指示します。AWS インフラストラクチャ料金の返金は、ソフトウェアの返金とは無関係に処理 AWS され、の裁量に委ねられます。

サードパーティーによって製品が販売されている場合は、購入者は製品の詳細ページの返金ポリシーを確認するよう指示されます。AWS Marketplace サブスクリプションのソフトウェア料金は製品の販売者に支払い、返金は販売者から直接リクエストする必要があります。各 AWS Marketplace 販売者は、AWS Marketplace ページに返金ポリシーを含める必要があります。

AWS Marketplace 製品の返金ポリシーと承認

次のリストは、AWS Marketplace 返金ポリシーと、承認が必要かどうかの概要を示しています。

- 無料トライアル

ソフトウェアを無料トライアル製品として一覧表示した場合、は無料トライアルから有料サブスクリプションへの変換から 7 日以内に発生するソフトウェア料金の返金をユーザーに代わって発行 AWS できます。返金が無料トライアルの変換に関連して行われる場合、お客様によるアクションは不要です。製品の無料トライアルを有効にすると、このポリシーに同意したものとみなされます。

- プライベートオファー

プライベートオファーの返金はすべて、が AWS 処理する前にお客様の承認を受ける必要があります。

- ソフトウェア計測の返金

を使用してソフトウェアの使用量を計測する場合 AWS Marketplace Metering Service、はソフトウェア計測エラーによるソフトウェア料金の返金をユーザーに代わって発行 AWS できます。これらのエラーが複数のお客様に共通している場合、は各お客様に適切な返金を決定し、各お客様に直接適用する権利 AWS を予約します。に関連して発行された返金は、販売者に 1 回確認 AWS

Marketplace Metering Service する必要がありますが、販売者が個々の返金を確認する必要はありません。を製品 AWS Marketplace Metering Service で使用すると、このポリシーに同意したことになります。

- 購入後 48 時間以内のサブスクリプションキャンセル

プライベートオファー以外の購入から 48 時間以内に購入者がサブスクリプションをキャンセルした場合、AWS から全額返金されます (100% 返金されるキャンセル)。購入から 48 時間以内のキャンセルに対する返金には、お客様側での対応は必要ありません。48 時間経過後にこのような購入者のリクエストがあった場合は、お客様の裁量に委ねられます。AWS Marketplace に製品を出品することにより、本ポリシーに同意したものとみなされます。

- サブスクリプションのアップグレード

購入者が既存の非プライベートオファーサブスクリプションをより高価なサブスクリプションまたは同等の価値のサブスクリプションに置き換えた場合、はユーザーに代わって低層サブスクリプションの返金を発行 AWS できます。購入者にとっては 2 段階のプロセスになります。新しいサブスクリプションを購入し、次に古いサブスクリプションの解約と返金をリクエストします。

- サブスクリプションのダウングレード

ダウングレードサブスクリプションの返金リクエストはすべて、が AWS 処理する前にお客様の承認を受ける必要があります。

AWS 承認された返金はすべて自動的に処理され、お客様側で何もする必要はありません。

AWS Marketplace 製品の返金プロセス

[返金申請フォーム](#)を送信すると、製品ソフトウェアの使用に対する返金を開始できます。AWS Marketplace 購入者サポートチームが受け取ると、関連するサポートケースが[AWS Support センターコンソール](#)に作成され、返金ステータスが件名に表示されます。返金関連のサポートは、これらのケースを通じて直接行われます。詳細については、「[AWS Support へのアクセス](#)」を参照してください。

以下の手順は、外部の顧客または内部のテスト用アカウントへの返金をリクエストする方法の概要です。

顧客にソフトウェアの返金を開始するには

1. 顧客についての以下の情報を収集します。

- に関連付けられているお客様の E メールアドレス AWS アカウント。

- 製品のサブスクリプションに使用されるアカウントの顧客 AWS アカウント 番号。顧客が組織の支払者である場合は、製品にサブスクリプションしているリンクされたアカウントの AWS アカウント ID を提供する必要がありますを顧客に伝えます。
 - 顧客が返金を求める請求期間。
2. にサインインし AWS アカウント、[返金リクエストフォーム](#) に移動します。
 3. フォームに顧客の情報を入力します。
 4. 顧客が返金をリクエストしている製品の製品 ID を入力します。製品 ID は、[日次カスタマーサブスクリプションレポート](#) で確認できます。
 5. 顧客が年間製品に対して返金、アップグレード、ダウングレードをリクエストしている場合は、次のタスクを完了する必要があります。
 - a. サブスクリプションレポート (反映が 24 時間後になる場合がある) を使用して、顧客が年間サブスクリプションを購入していることを確認します。
 - b. コメントフィールドにサブスクリプションのキャンセルの日付を入力します。
 - c. コメントフィールドに、承認中 (返金、アップグレード、ダウングレード) の変更の説明を入力します。
 6. フォームを送信します。当社に通知され、返金を処理した後、顧客に返金します。
 7. 返金リクエストのステータス情報を含むアウトバウンドケースが [AWS Support Center コンソール](#) で作成されます。件名には、次のいずれかが含まれます。
 - 完了 - 返金処理が完了したため、これ以上の操作は必要ありません。
 - 保留中 - 現在の請求サイクルが終了すると、返金が処理されます。
 - アクションが必要 - リクエストを処理できませんでした。追加情報が必要です。サポートケースに直接返信できますが、新しい返金リクエストフォームも送信する必要があります。
 8. 返金が正常に処理されると、24 ~ 48 時間以内に顧客のアカウントに反映されます。ただし、顧客の金融口座に資金が表示されるまでに最大 5 営業日かかります。

AWS Marketplace 製品のリージョンと国

で製品を作成するときは AWS Marketplace、その製品が利用可能な AWS リージョンを選択します。購入者が製品を購入できる国も選択します。これら 2 つのプロパティは似ていますが、同じではありません。例として、購入者が米国に居住し、米国から購入していても、製品を欧州 (フランクフルト) リージョンにインストールしている場合があります。この購入者が製品を購入するには、国のリストに米国を、リージョンのリストに欧州 (フランクフルト) を含める必要があります。

AWS リージョン

サーバーまたは機械学習製品情報を作成または編集するときは、ユーザーが製品をインストールして使用できる特定の AWS リージョンに製品を制限できます。

Amazon マシンイメージ (AMI)、コンテナベース、AWS CloudFormationベースの製品を含むサーバー製品の場合、製品が利用可能な特定のリージョンを選択できます。また、製品が利用可能になりしだい、新しい米国リージョン、米国以外のリージョン、またはすべてのリージョンで製品を自動的に利用可能になるように選択することもできます。

機械学習製品の場合は、特定のリージョンを選択することも、今後利用可能になるリージョンを含むすべてのリージョンを選択することもできます。

AWS リージョンの詳細については、AWS 全般のリファレンスの「[AWS サービスエンドポイント](#)」を参照してください。

Countries (国)

デフォルトでは、が利用可能なすべての国の購入者 AWS Marketplace が製品を利用できます。新規および既存のサーバーと Software as a Service (SaaS) 製品については、税務、コンプライアンス、サポート、またはマーケティングを目的として、特定の国での製品の使用可否を管理できます。

この機能には例外があります。

- 以前の購入 - 新しい国のリストと共に製品を更新した後も、既に製品をサブスクライブしている購入者は、サブスクリプションが有効である間、引き続きアクセスできます。
- プライベートオファー - 製品を特定の国の購入者に限定しても、プライベートオファーは制限されません。特定の購入者に対してプライベートオファーを作成すると、指定した国に含めなかった国にその購入者がいる場合でも利用できます。

Note

顧客の適格性は、AWS リンクされたアカウントレベルで決定されます。詳細については、「[How does AWS determine the Location of your account?](#)」を参照してください。

使用権限を共有する顧客は、許可したリージョンでのみその使用権限を有効化できます。使用権限の管理について詳しくは、「AWS Marketplace 購入者ガイド」の「[組織内でのサブスクリプションの共有](#)」を参照してください。

での標準化された契約 AWS Marketplace

[製品を準備する](#)際、製品の使用に対してどのエンドユーザーライセンス契約書 (EULA) を適用するか決定する必要があります。独自の EULA を適用するか、標準契約 AWS Marketplace (SCMP) を使用できます。SCMP は、調達ワークフローを合理化し、トランザクションを高速化するために AWS Marketplace が提供する契約テンプレートです。また、ISV が使用できる標準化されたリセラー契約テンプレートであるリセラー契約 AWS Marketplace (RCMP) も利用できます。ISVs

このセクションでは、AWS Marketplace で使用できる 2 つの標準契約書の概要を説明します。

トピック

- [の標準契約 AWS Marketplace](#)
- [のリセラー契約 AWS Marketplace](#)

の標準契約 AWS Marketplace

AWS Marketplace は、購入者および販売者のコミュニティと協力して、[標準契約 AWS Marketplace \(SCMP\)](#) を作成しました。SCMP は、デジタルソリューションの使用方法を規定し、購入者と販売者の義務を定めています。デジタルソリューションの例としては、サーバーソフトウェア、Software as a Service (SaaS)、人工知能と機械学習 (AI/ML) アルゴリズムなどがあります。

SCMP では、使用、保証、補償、準拠法などの主要な契約条項に共通する点が既に定義されています。販売者は、購入者がソリューションを検索、購入、迅速にデプロイできるセルフサービス取引の EULA として SCMP 条件を提示できます。[プライベートオファー](#)の場合、購入者は販売者に SCMP テンプレートをリクエストでき、両当事者の合意に従ってカスタム取引要件に対応するように条件を修正できます。

セルフサービスまたはプライベートオファーについては、以下の補遺を SCMP にオプションで付けることもできます。

- [セキュリティ強化補遺](#) - データセキュリティ要件が厳しい取引をサポートします。
- [HIPAA ビジネスアソシエイト補遺](#) - HIPAA (Health Insurance Portability and Accountability Act of 1996: 医療保険の携行性と責任に関する法律) に準拠している取引をサポートします。

免責事項

EULA はお客様と購入者の間で締結されます。SCMP を EULA として使用するかどうかは、お客様の裁量に委ねられます。SCMP を製品リストに適用することにより、SCMP プログラムに参加したことになります。このプログラムでは、SCMP テンプレートを定期的に AWS 更新し、条件を含む製品リストを最新バージョンで更新する場合があります。SCMP テンプレートを独自の EULA に置き換えることにより、いつでも SCMP プログラムから脱退することができます。

SCMP の利用方法

EULA を SCMP に更新して、新規および既存の Amazon マシンインスタンス (AMI) および Software as a Service (SaaS) の単一製品の購入者に提示することができます。更新をリクエストする手順は、製品がセルフサービスリスト (SSL) を通じて出品されているかどうかによって異なります。

ご質問がある場合は、の AWS Marketplace 標準契約チームに E メールメッセージを送信してください aws-mp-standardcontract@amazon.com。

SSL から出品されている AMI 製品および SaaS 製品の EULA を SCMP に更新するには

1. [AWS Marketplace 標準契約書](#) の条項を確認します。
2. [AWS Marketplace 管理ポータル](#) にサインインします。
3. 製品については [製品タイプ] を選択し、契約書を更新する [製品リスト] を選択します。
4. [変更をリクエスト] を選択し、[リージョンと料金の更新] を選択します。
5. EULA に [AWS Marketplace 標準契約書] を選択します (選択されていない場合)。
6. [メモと通知] に **Please update this product to SCMP V2** と入力します。
7. 変更内容を確認し、[レビューのために送信] をクリックします。

SSL から出品されていない AMI 製品および SaaS 製品の EULA を SCMP に更新するには

1. AWS Marketplace 管理ポータルの [お問い合わせ](#) ページを開きます。
2. AWS Marketplace 販売者アカウントにサインインします。
3. 以下の手順に従ってフォームに入力します。
 - 質問の件名として、[Commercial Marketplace] を選択します。

- カテゴリとして、[Product Listing] を選択します。
- サブカテゴリとして、[Standard Contract Request] を選択します。
- 説明に、**Please update these products to SCMP V2** と入力し、SCMP に更新する製品のタイトルと ID を入力します。

のリセラー契約 AWS Marketplace

リセラー契約 AWS Marketplace (RCMP) は、ISV がチャンネルパートナーに ISVs 製品を AWS Marketplace 購入者に再販することを許可する際に使用できる標準化されたリセラー契約テンプレートです。この契約書は、ISV とチャンネルパートナーがリセラー関係を締結したり、[チャンネルパートナープライベートオファー \(CPPO\)](#) のリセラー条件を使用したりする際に、法的契約レビューの重複を減らし、市場投入までの時間を短縮するのに役立ちます。CPPO を作成すると、ISVs は契約をオポチュニティにアップロードし (再販承認)、チャンネルパートナーは契約を表示して承諾できます。AWS Marketplace 購入者は RCMP を表示できません。

免責事項

RCMP は ISV 向けのオプション契約です。ISV がチャンネルパートナーを通じて製品を再販することを決定した場合、オポチュニティを作成する際に RCMP を添付することも、既存または事前に交渉した独自の契約条件を添付することもできます。

RCMP の利用方法

このセクションでは、条件を確認して RCMP を利用する方法について説明します。

AWS Marketplace オポチュニティの作成中に RCMP を使用するには

1. [AWS Marketplace リセラー契約書](#) の条項を確認します。
2. [RCMP ガイド](#) を参照して、[チャンネルパートナーの再販オポチュニティ](#) を作成します。

カテゴリとメタデータ

製品メタデータを提供するためのベストプラクティスと情報は次のとおりです。AWS Marketplace は、品質保証とエラー修正のみを目的として製品メタデータを改訂します。

製品の名前付けと説明

販売者が製品について提供する情報は、購入者に表示されます。潜在的な購入者が製品に関して十分な情報を得たうえで購入の決定ができるようにしてください。

製品名の決定

製品名を作成する際は、次のガイドラインに注意してください。

- タイトルケースを使用する (重要語それぞれの最初の文字が大文字になっている)
- 購入者が製品を名前のみで識別できるようにする
- ブランドまたは製造元の名前を使用する
- 説明的なデータや誇張を避ける

製品名の例: Smart Solution ロードバランサー - プレミアムエディション。

製品説明を書く

製品の説明では、製品の機能、利点、使用方法を示します。その他、製品固有の関連情報も提供できます。説明は最大 350 文字とすることができます。

製品説明を記述する際は、次のガイドラインに注意してください。

- 不必要な大文字を避ける
- 不要な句読点を避ける
- リダイレクト情報を含めない
- スペルと文法を確認する
- 重要で有益な情報のみを含める

製品ソリューションの例: Smart Solution は、アプリケーションの着信トラフィックを複数の Amazon EC2 インスタンスに自動的に分散します。これにより、アプリケーションの着信トラフィックに応じて必要な量の負荷分散能力を提供し、アプリケーションの耐障害性をさらに高めることができます。Smart Solution はプール内の異常なインスタンスを検出すると、この異常なインスタンスが回復するまで、自動的にトラフィックを正常なインスタンスに振り分けます。Smart Solution は、単一または複数の AWS アベイラビリティーゾーンで有効にすると、安定したアプリケーションパフォーマンスを実現できます。

製品ハイライトを書く

製品情報ページは、最大 3 つの製品ハイライトの箇条書きを表示します。これらの箇条書きを使用して、製品の主なセールスポイントを簡単に説明します。

製品ハイライトの例: コストの見積り - Smart Solution では、使用した分のみ支払います。Smart Solution が実行されている 1 時間 (または 1 時間未満) ごとに請求されます。

カテゴリとキーワードの選択

製品を出品するとき、製品に最大で 3 つのソフトウェアカテゴリと、対応するサブカテゴリを選択できます。これにより、購入者は製品を参照または検索するときに製品を検出できます AWS Marketplace。製品に関連するカテゴリのみを選択してください。ほとんどの場合、該当するカテゴリは 1 つのみです。製品ロードフォームおよび [製品] タブに、カテゴリの完全なリストが含まれます。

カテゴリはキーワードと同一ではありません。使用可能なカテゴリとサブカテゴリは用に事前定義されており AWS Marketplace、製品リクエストプロセス中にリストから選択して、製品に適用するカテゴリを決定します。キーワードは事前定義されていません。プロセス中に作成されます。カテゴリをキーワードとして追加する必要はありません。

検索キーワードの作成

製品リクエストプロセス中に、購入者がサイトを検索して製品を見つけやすくするために、最大 3 つのキーワード (単語またはフレーズ) を入力できます。キーワードのフィールドには最大 250 文字含めることができます。

次のヒントを活用して、関連する一連の検索キーワードを作成します。

- 関連する用語を使用します。
- 他の販売者が公開している製品名を使用したり、他の販売者の名前を使用したりしないでください。
- 購入者のボキャブラリーからキーワードを選択します。つまり、この種の製品について購入者が考えるときに使用する可能性が高い単語やフレーズを選択します。
- 製品の具体的な特徴にもとづいてキーワードを作成する。
- 製品タイトルをキーワードとして使用しない。製品タイトルは検索のインデックスに登録済みです。

注: キーワードはソフトウェアカテゴリと同一ではありません。キーワードは、製品に関連する、より具体的な用語です。

AMI とコンテナ製品の使用手順

製品の使用手順を作成する際には、以下の情報を含める必要があります。

- 顧客が保存したすべての機密情報の保存場所
- すべてのデータ暗号化設定の説明
- プログラムによるシステム認証情報と暗号化キーをローテーションするための S tep-by-step 手順。[the section called “AMI ベースの製品要件”](#) は、認証情報と暗号化キーを使用するリストの基本要件について説明します。
- アプリケーションが何らかの暗号化技術を使用している場合、ユーザーがアプリケーションをどのように操作して必要なデータを復号化するかについて詳細に説明してください。
- アプリケーションの正常性と適切な機能を評価してモニタリングする方法に関する S tep-by-step の手順。例:
 - [Amazon EC2 コンソール](#) に移動し、正しいリージョンにいることを確認します。
 - [インスタンス] を選択し、起動しているインスタンスを選択します。
 - サーバーを選択してメタデータページを表示し、ページ下部の [ステータスチェック] タブを選択してステータスチェックに合格したか失敗したかを確認します。

リリースノートを書く

製品を更新するたびに、変更の内容をリリースノートで説明する必要があります。リリースノートには、ユーザーが更新をインストールするかどうかを判断できる具体的な情報を記載する必要があります。セキュリティ更新には「重大」、または他の種類の更新には「重要」や「オプション」といった、明確なラベルを使用します。

使用手順を書く

購入者がソフトウェアを正常に設定して実行するために役立つ使用手順を提供します。提供した使用手順は、設定プロセス中に表示されます。

効果的な使用手順を記述するには、次のガイドラインに従ってください。

- 新規のユーザーまたは技術的な知識が少しあるユーザーを対象に記述します。

- 製品、コンピュータのオペレーティングシステム、エンジニアリング、IT オペレーションについて、ユーザーに使用経験や幅広い知識があるとは想定しないでください。
- アプリケーションを実行するための設定や特別なステップも含め、製品の起動から使用までの手順を示します。

使用手順の例:

1. 1-Click を使用して製品を起動します。
2. ウェブブラウザを使用して `https://<EC2_Instance_Public_DNS>/index.html` からアプリケーションにアクセスします。
3. 次の認証情報を使用してサインインします。
 - ユーザー名: user
 - パスワード: インスタンスのインスタンス ID

アップグレード手順を書く

製品を旧バージョンからアップグレードする方法について購入者に詳しく説明します。別のインスタンスの作成時にデータと設定を保存する方法についての情報を含めます。アップグレードパスがない場合は、このフィールドを編集して具体的に説明してください。

アップグレード手順の例:

1. **** を実行してから、**** します。
2. *** を実行することによって、プロジェクトで使用されるすべてのプラグインが、バージョン *.* と互換性があることを確認します。互換性がない場合は *** を実行します。
3. *** を実行して、データのバックアップを作成します。

CloudFormation 配信

CloudFormation 配信を使用する場合は、以下も含める必要があります。

- テンプレートによって AWS CloudFormation 作成された各 AWS Identity and Access Management (IAM) ロールと IAM ポリシーの目的
- AWS CloudFormation テンプレートによって作成された各キーの目的と場所
- 複数の要素を含むデプロイにおけるネットワーク設定の詳細

- アプリケーションの起動方法と、デプロイに複数の AWS リソースが含まれている場合に通信するように設定する方法に関する詳細なガイド
- 標準制限を超えて追加された AWS リソースの実行コストを含む料金内訳。AWS サービス制限の管理に関する規範的なガイダンスを提供します。
- すべてのデータ暗号化設定。例: Amazon S3 サーバー側の暗号化、Amazon Elastic Block Store (Amazon EBS) 暗号化、Linux Unified Key Setup (LUKS) など

アプリケーション機能のモニタリングと評価

アプリケーション機能をモニタリングおよび評価するには

1. [Amazon EC2 コンソール](#)に移動し、正しいリージョンにいることを確認します。
2. [インスタンス] を選択し、起動しているインスタンスを選択します。
3. サーバーを選択してメタデータページを表示し、ページ下部の [ステータスチェック] タブを選択してステータスチェックに合格したか失敗したかを確認します。

Note

データストアのいずれかが専有データストアである場合は、設定、バックアップ、復旧 step-by-step の手順を提供します。

プログラムによるシステム認証情報と暗号化キーのローテーション

[the section called “AMI ベースの製品要件”](#) は、認証情報と暗号化キーを使用するリストの基本要件について説明します。

プログラムによるシステム認証情報と暗号化キーのローテーションには、以下を含めてください。

- AWS サービスクォータの管理に関する規範的なガイダンス。詳細については、「[AWS 全般のリファレンスガイド](#)」を参照してください。
- 標準クォータを超えて追加された AWS リソースの実行コストを含む料金内訳。この情報は、製品の使用手順に記載することも、Service Quotas (階層) の管理と増額のリクエストに関する詳細情報を含む[ドキュメント](#)リンクすることもできます。

製品の検索エンジン最適化

検索は、購入者のジャーニーにおいて重要なツールであり、顧客は独自のニーズを満たす最適な製品を見つけることができます。AWS Marketplace 顧客の場合、検索は主に 2 つの場所で行われます。検索エンジン (Google や Bing など) との検索関数です AWS Marketplace。このページでは、両方のロケーションのリストを最適化する方法について説明します。

検索エンジンの最適化

関連するキーワードの上位にランク付けするように製品詳細ページを最適化することは、検索エンジンを通じて製品詳細ページへのよりユニークな訪問者を誘導するために重要です。

製品の詳細ページには、キーワード、タイトルタグ、H1 見出しタグの 3 つの主要なページ要素があり、これらは、ウェブページへの器質的な検索を改善するために基礎的かつ影響があります。

キーワード

キーワードは、検索エンジンの最適化の中心的な要素であり、トピックを検索エンジンの結果を促進するフォーカスされたクエリに分割します。ページに最も関連性の高いキーワードを特定するプロセスには、キーワード調査が必要です。検索エンジン最適化ツールは、キーワード検索ボリューム (Google でキーワードが検索される 1 か月あたりの回数)、現在のランキング、検索傾向、キーワードの競合性、関連するキーワードなどの貴重な情報を提供します。この調査では、プライマリキーワードとセカンダリキーワードを識別できます。

主な検索エンジン最適化キーワードは、ページの主要トピックを表す一意の単一の単語またはフレーズである必要があります。この主なキーワードは、製品タイトル、簡単な説明、ハイライトセクションのコピーに自然に反映する必要があります。セカンダリキーワードは、残りのページの内容に含まれる関連性の高い用語である必要があります。

タイトルタグ

検索エンジンの結果ページとブラウザウィンドウまたはタブ内のページのタイトルとして表示されるタイトルタグは、リーダーと検索エンジンの両方のウェブクロウラーにページのコンテンツを通知します。AWS Marketplace 製品詳細ページでは、製品タイトルがタイトルタグとして機能するため、ランキングの可能性を高めるために、検索エンジン最適化キーワードを使用して製品タイトルを最適化することが重要です。検索エンジンの結果ページで高ランクを達成する可能性を高めるには、ブランド名、製品名、関連するキーワードをタイトルタグに組み込みます。

H1 見出しタグ

H1 見出しタグには 3 つのルールがあります。

- 訪問者がページの内容をスキャンして必要な情報を確認するのに役立ちます。
- これにより、スクリーンリーダーを使用してページの内容を理解する視覚障害を持つ訪問者のアクセシビリティが向上します。
- ページ見出しにあるキーワードを提供します。これにより、以下のページコンテンツでサポートされている場合は、検索エンジンの最適化の関連性の重みが付加されます。

AWS Marketplace 検索

この AWS Marketplace ウェブサイトでは、業界全体で使用されている検索最適化手法と同様の検索クエリの結果をランク付けしています。が検索結果を AWS Marketplace ランク付けして返す方法を理解することで、AWS Marketplace 検索エンジンに最適化された製品の詳細を作成できます。製品詳細ページを作成するときは、このガイダンスを考慮することをお勧めします。

キーワード

製品作成プロセス中に、顧客がサイト検索を通じてお客様の製品を見つける手助けをするため、最大 3 つのキーワード (単語またはフレーズ) を送信できます。キーワードテキストボックスに最大 250 文字を含めることができます。

検索キーワードを作成するために以下のヒントを使用します。

- 顧客が簡単にお客様の製品を見つけられるよう、関連する用語を使用する。
- 顧客のポキャブラリーにあるキーワード、つまり、製品タイプのことを考えるときに使用する可能性が高い単語やフレーズを選択する。
- 製品の具体的な特徴にもとづいてキーワードを作成する。
- 送信する用語に製品タイトルを含めない。製品タイトルは検索用にすでにインデックス化されています。

Note

キーワードはソフトウェアカテゴリと同一ではありません。キーワードは、製品に関連する、より具体的な用語です。

製品を作成した後に、製品のメタデータを編集してキーワードを編集できます。AWS Marketplace 管理ポータル の [製品] タブを使用して作成した製品の場合は、[製品] タブを使用して変更を加えます。詳細については、「[製品の変更と更新](#)」を参照してください。

AWS Marketplace Seller Operations チームは、同様の意味の単語や類似する意味の単語を含むクエリのリダイレクトを支援します。例えば、顧客が自動車を検索すると予想していたときに、車を検索した場合などです。

ソフトウェアカテゴリ

製品を出品するとき、製品に最大で 3 つのソフトウェアカテゴリと、対応するサブカテゴリを選択できます。これにより、顧客は製品を参照または検索するときに製品を検出できます AWS Marketplace。製品に関連するカテゴリのみを選択します。多くの場合、該当するのは 1 つのカテゴリのみです。Product Load Form および [製品] ページの両方に、カテゴリの完全なリストが含まれます。

Note

カテゴリはキーワードと同一ではありません。使用可能なカテゴリとサブカテゴリは、用に事前定義されています AWS Marketplace。リストからカテゴリを選択して、製品に適用するカテゴリの種類を決定します。キーワードは事前定義されていません。プロセス中に作成されます。

ハイライトセクション

製品詳細ページでは、最大 3 つの製品ハイライトを箇条書きで表示します。顧客はハイライトで製品を検索できるため、製品を作成するときにハイライトを含めます。ハイライトは、製品の主なセールスポイントを簡潔にわかりやすい言葉で記述する必要があります。

Example [ハイライト]

- コストの予測: AnyCompany の製品では、使用した分に対してのみ料金が発生します。AnyCompany が実行されている 1 時間 (または 1 時間未満) ごとに請求されます。

簡単な説明

製品説明は、製品の特徴、利点、使用方法を一覧表示し、他の関連する特定の製品情報を提供します。製品説明を作成する際は、次のガイドラインに注意してください。

- 必要以上に大文字や句読点を使わない
- リダイレクト情報を含めない
- スペルと文法を確認する
- 重要で有益な情報のみを含める

Example 簡単な説明

AnyCompanyの製品は、受信アプリケーショントラフィックを複数の Amazon EC2 インスタンスに自動的に分散します。これにより、受信アプリケーショントラフィックに対応するために必要な負荷分散容量をシームレスに提供することで、アプリケーションの耐障害性を向上させることができます。AnyCompanyの製品は、プール内の異常なインスタンスを検出し、異常なインスタンスが復元されるまでトラフィックを正常なインスタンスに自動的に再ルーティングします。お客様は、単一の AWS アベイラビリティゾーンで、または複数のアベイラビリティゾーンで有効にして、より一貫したアプリケーションパフォーマンスを実現できます。

プライベートオファーの準備

プライベートオファーは、 から製品を購入するために使用されるネゴシエートされた用語です AWS Marketplace。これには、カスタム料金プラン、エンドユーザーライセンス契約 (EULA)、またはカスタムソリューションが含まれる場合があります。販売者と購入者は、パブリックオファーとは異なるプライベートオファーにコミットする前に交渉します。

複数のプライベートオファーを作成して、1人の購入者に提供することができます。プライベートオファーが適用された購入者は、プライベートオファーとパブリックオファーのどちらかを選択できます。購入者が一度に購読できるオファーは1つのみです。プライベートオファーとパブリックオファーの両方を同時に購読することはできません。

Note

AWS は、購入者に のユニークまたはエンタープライズのユースケースを提供 AWS Marketplace して、製品の詳細ページから直接製品のプライベートオファーをリクエストします。APN カスタマーエンゲージメント (ACE) の対象となる AWS パートナーネットワーク (APN) パートナーで、このオプションを購入者に提供する場合は、AWS 担当者に連絡して、さらなる適格性を判断してください。 <https://aws.amazon.com/partners/programs/ace/>

トピック

- [プライベートオファーの仕組み](#)
- [プライベートオファーでサポートされている製品タイプ](#)
- [プライベートオファーの作成と管理](#)
- [AWS Marketplace チャンネルパートナーのプライベートオファー](#)
- [分割プラン](#)
- [プライベートオファーのアップグレード、更新、修正](#)
- [将来の日付の契約とプライベートオファー](#)

プライベートオファーの仕組み

のオファーページからプライベートオファーを作成および管理できます [AWS Marketplace 管理ポータル](#)。オファーの製品を指定して、一意の ID と URL を生成します。プライベートオファーの料金

プランを作成し、法的条件と販売文書を追加し、オファーを特定の購入者に拡張します AWS アカウント。オファーは、オファーを作成したアカウントのみに表示されます。

プライベートオファーを作成し、潜在的な購入者に通知すると、購入者はオファーを表示して受け入れることができます。オファーを表示するには、購入者がオファー AWS アカウント を受け取ったにサインインする必要があります。

Note

購入者は、リンクされたアカウントまたは管理アカウントに拡張しない限り、オファーを表示できません。オファーでサービス制限を指定することはできません。そのため、製品に制限がない限り、購入者は交渉済みの価格で製品を使用できます。

プライベートオファーの作成については、[「プライベートオファーの作成と管理」](#)を参照してください。

プライベートオファーは販売者レポートで追跡されます。詳細については、[「プライベートオファーのレポート」](#)および[「販売者レポートガイド」](#)を参照してください。

プライベートオファーに関する考慮事項

プライベートオファーを使用する場合は、次の点を考慮してください。

- 新しいインスタンスタイプまたは のサポートを追加すると AWS リージョン、製品のプライベートオファーに既にサブスクライブしているお客様は、新しく追加されたインスタンスまたはリージョンに自動的にアクセスできなくなります。お客様がアクセスするインスタンスとリージョンを使用して、別のプライベートオファーを作成する必要があります。新しいオファーを承諾すると、お客様は新しく追加されたインスタンスとリージョンにアクセスできます。将来の日付に製品をサブスクライブするお客様は、プライベートオファーに含まれている限り、製品にアクセスすることもできます。新しいプライベートオファーを作成する方法の詳細については、[「プライベートオファーのアップグレード、更新、および修正」](#)を参照してください。
- AWS CloudFormation 製品を使用したサードパーティー、Amazon マシンイメージ (AMI) の毎月の配信、またはマルチ AMI ベースの配信のプライベートオファーを作成したり、顧客の使用を制限したりすることはできません。
- 分割払いプランのプライベートオファーでは、前払いを時間の経過とともに複数の支払いに分割できます。詳細については、[「Installment plans」](#)を参照してください。

- プライベートオファーの購入者アカウントがプライベートマーケットプレイ스에서管理されている場合は、購入者のアカウントと、プライベートマーケットプレイス管理者が含まれるアカウントの両方をオファーに含める必要があります。
- プライベートオファーは、Bring Your Own License (BYOL) モデルをサポートしていません。
- プライベートオファーで一意のネゴシエートされた契約条件を持つプライベートオファーを作成するときは、カスタム EULA オプションを使用します。最大 5 つのドキュメントをアタッチできます。
- Software as a Service (SaaS) 契約製品および従量課金制の SaaS 契約製品については、購入者がプライベートオファーを承諾して締結した契約に対してアップグレードや更新を提供できます。これにより、新しい権利の付与、料金割引の提供、支払いスケジュールの調整、またはエンドユーザーライセンス契約 (EULA) の変更を行い、標準化されたライセンス条項を使用できます。詳細については、[「プライベートオファーのアップグレード、更新、修正」](#)を参照してください。

購入者向けのプライベートオファーエクスペリエンス

顧客が購入者のサブスクリプションページに移動すると、プライベートオファーが利用可能であることを示すバナーが表示されます。購入者がオファーを承諾すると、すべての AWS Marketplace トランザクションで使用されているのと同じポータルツールを使用して、購入の請求が行われます。承諾されたオファーは契約になります。購入者はの「サブスクリプションの管理」セクションで契約の詳細を検索でき AWS Management Console、販売者はの「契約」タブで詳細を確認できます AWS Marketplace 管理ポータル。

AWS Marketplace 購入者は、プライベートオファーのためにサードパーティーの資金調達にアクセスできます。詳細については、[「カスタマーローンがで利用可能になりました AWS Marketplace」](#)を参照してください。

Note

オファーは有効期限が切れる前にのみ受け入れることができます。オファーの有効期限が切れると、承諾済みおよび期限切れのオファータブに移動します。

<p>プライベートオファーを表示して受け入れるには</p>	<p>購入者は次のことができます。</p>
<p>AWS Marketplace コンソールから</p>	<p>AWS Marketplace コンソールでプライベートオファーに移動し、利用可能なオファータブからオファー ID を選択します。</p> <p>プライベートオファーの購入者体験の詳細については、「AWS Marketplace 購入者ガイド」の「プライベートオファー」を参照してください。</p>
<p>販売者が提供したリンクを使用する</p>	<p>販売者から送られたリンクに従って、プライベートオファーに直接アクセスします。</p> <p>詳細については、「購入者へのプライベートオファーの送信」を参照してください。</p>
<p>製品ページから</p>	<p>製品の製品ページに移動して、バナーのリンクを選択しプライベートオファーを表示します。</p> <p>プライベートオファーの購入者体験の詳細については、「AWS Marketplace 購入者ガイド」の「プライベートオファー」を参照してください。</p>

プライベートオファーのレポート

プライベートオファーは、既存の販売者レポートとオファーに関連するレポートに表示されます。[月別請求済み収益レポート](#)が毎月生成され、オファーの可視性とオファー ID 情報が記載されます。購入者の請求書が生成されたときに、該当する請求期間をカバーするレポートに表示されます。詳細については、「[販売者ダッシュボード](#)」を参照してください。

[Offer ID (オファー ID)] フィールドには、プライベートオファー用に生成された一意のオファー ID が記載されています。このレポートのエントリがプライベートオファー用でない場合、このフィールドは空白になります。[Offer Visibility (オファーの可視性)] フィールドには、レポートのエントリがパブリックオファーかプライベートオファーかが表示されます。プライベートオファーの場合はすべて、エントリがプライベートと記載されます。

プライベートオファーでサポートされている製品タイプ

Amazon マシンイメージ (AMIs)、コンテナ、プロフェッショナルサービス、Software as a Service (SaaS) 製品は、プライベートオファーでサポートされています。

AMI 製品のプライベートオファー

AMI 製品のプライベートオファーの料金を提供できます。

オファーは、以下の期間であれば自由にカスタマイズできます。

- AMI (時間単位) または年間プライベートオファー付きの AMI (時間単位): 最長 3 年間 (1,095 日)。柔軟な支払いスケジュールをサポートしているのは、年間プライベートオファー付きの AMI (時間単位)、または AMI 契約のみです。
- AMI 契約のプライベートオファー: 最長 5 年 (60 か月)

AMI 契約の場合、プライベートオファーによって使用状況が監視されることはありません。

購入者はいつでも手動で新しい契約レベルにアップグレードできます。ただし、契約階層を定義して、サービス制限を適用し、購入者に手動で契約をアップグレードしユニット数を増やすよう勧めるのは、独立系ソフトウェアベンダーの責任です。現時点でアップグレードをサポートしているのは、段階的でない価格ベースの契約のみです。プライベートオファーの契約期間は、公開されている製品リスト、または月単位 (最大 60 か月) のカスタム期間と一致する場合があります。

ライセンス使用権限は、購入者がプライベートオファーを承諾した日付から始まります。

柔軟な支払いスケジュールを持つ AMI プライベートオファーでは、契約期間中、契約で合意した年間インスタンスタイプの数を設定できます。

Note

プライベートオファーは月単位の請求契約ではご利用いただけません。

コンテナ製品のプライベートオファー

コンテナベースの製品契約には、プライベートオファー価格を提供できます。

オファーは、以下の期間であれば自由にカスタマイズできます。

- コンテナ (時間単位) または長期プライベートオファー付きのコンテナ (時間単位) - 最長 3 年間 (1,095 日)。柔軟な支払いスケジュールをサポートしているのは、年間プライベートオファー付きのコンテナ (時間単位)、またはコンテナ契約のみです。
- コンテナ契約のプライベートオファー: 最長 5 年 (60 か月)

コンテナ契約の場合、プライベートオファーによって、使用状況が監視されることはありません。コンテナ契約のアップグレードは段階的でない価格を使用している場合にのみ可能です。

購入者はいつでも手動で新しい契約レベルにアップグレードできます。ただし、独立系ソフトウェアベンダーは契約階層を定義して、サービス制限を強制し、購入者に手動で契約をアップグレードしてユニット数を増やすよう勧めます。現時点でアップグレードをサポートしているのは、段階的でない価格ベースの契約のみです。プライベートオファーの契約期間は、公開されている製品リスト、または月単位 (最大 60 か月) のカスタム期間と一致する場合があります。

ライセンス使用権限は、購入者がプライベートオファーを承諾した日付から始まります。柔軟な支払いスケジュールを持つコンテナプライベートオファーでは、契約期間中、契約で合意したユニットの数を設定できます。また、購入者がそれ以上を使用する場合に、同じユニットに対してカスタムの時間料金を定義することもできます。

Note

プライベートオファーは月単位の請求契約ではご利用いただけません。

プロフェッショナルサービス製品のプライベートオファー

プロフェッショナルサービス製品の提供はすべてプライベートオファーを通じて行われます。詳細については、「[プライベートオファーの作成](#)」を参照してください。

SaaS 製品のプライベートオファー

Software as a Service (SaaS) のプライベートオファー製品は、特定の価格帯の価格レベルをタイミングに基づいて変更することはできません。例えば、あるオファーで 3 か月間は 1 時間あたり 0.80 USD を請求し、その後、同じ価格帯で 1 時間あたり 0.60 USD に変更することはできません。SaaS 契約の場合、プライベートオファーによって、使用状況が監視されることはありません。

購入者はいつでも手動で新しい契約レベルにアップグレードできます。ただし、独立系ソフトウェアベンダーは契約階層を定義して、サービス制限を強制し、手動で契約をアップグレードして必要に応

じて契約階層を上げるよう購入者に勧めます。プライベートオファターの契約期間は、公開されている製品リスト、または月単位 (最大 60 か月) のカスタム期間と一致する場合があります。

ML 製品のプライベートオファター

機械学習 (ML) プライベートオファター商品では、特定の購入者に、公開されている価格とは異なる価格を提示します。プライベートオファターにおける出品者と購入者との間の一連の条件と契約は、パブリックオファターや他のプライベートオファターの条件と異なる場合があります。

プライベートオファターは、以下のいずれかの方法を使用します。

- 時間単位 - プライベートオファターは、公開されている時間単位の料金とは異なる時間単位の場合があります。機械学習製品のプライベートオファターは期限切れにならないため、この時間単位の料金は永続的です。将来料金の変更が必要な場合は、購入者は新しいプライベートオファターに切り替える必要があります。製品の既存の実行中のインスタンスまたはエンドポイントには、新しい承認されたオファターで設定された時間単位の料金が自動的に請求されます。プライベートオファター内の契約コンポーネントの有効期限が切れたら、製品の時間単位の料金に設定してください。この時間料金を 0 USD に設定すると、購入者はソフトウェア料金なしで製品を無期限に使用できます。
- 推論単位 - 製品をエンドポイントとしてデプロイする場合の[推論価格](#)を設定している場合、プライベートオファターの推論レートは、公開されている推定レートとは異なる場合があります。
- 契約 - プライベートオファターは、指定された日数の固定前払い料金が適用される契約にできます。購入者は、契約期間中、インスタンス数の制限なく使用できます。契約の終了時に、継続して実行するすべてのインスタンスには、プライベートオファターで設定した時間単位の料金で請求されます。例えば、固定前払い料金で 365 日間無制限に使用できる契約を作成できます。プライベートオファターには時間単位の料金も設定できます。購入者がこのプライベートオファターを受け入れると、購入者はその前払い料金を支払うこととなります。契約が終了すると、まだ実行中のインスタンスはすべてその時間単位で請求されます。無料のプライベートトライアルを提供している場合は、無料の永久ライセンスが利用されないように、無料トライアル期間終了後の正しい時間単価を設定していることを確認してください。

複数のプライベートオファターを作成して、1 人の購入者に提供することができます。プライベートオファターが適用された購入者は、プライベートオファターとパブリックオファターのどちらかを選択できます。購入者が一度に購読できるオファターは 1 つのみです。プライベートオファターとパブリックオファターの両方を同時に購読することはできません。

SageMaker 製品の特定の購入者向けのプライベートオファターを作成するには、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。

プライベートオファーの作成をリクエストするときは、次の情報を提供する必要があります。ProductId、ターゲット購入者 AWS アカウント、日付と時刻 (オファーを承諾する必要がある日時を示す)、カスタム EULA ファイル (オプション)、返金ポリシー、ライセンス期間 (オプション)、ライセンス料金 (オプション)、柔軟な支払いスケジュール (オプション)、およびインスタンスタイプごとの料金。オファーが作成されたら、購入者は指定された日時にオファーを受け入れる必要があります。

プライベートオファーの作成と管理

以下のセクションでは、プライベートオファーを作成および管理する方法を説明します。

トピック

- [新しいプライベートオファーの開始](#)
- [オファーのステータスを理解する](#)
- [プライベートオファーのドラフトと公開](#)
- [購入者へのプライベートオファーの送信](#)
- [プライベートオファーの進行状況を保存する](#)
- [プライベートオファーの有効期限の更新](#)
- [プライベートオファーのキャンセル](#)

新しいプライベートオファーの開始

CreateOffer API 変更リクエストを使用してオファーを作成し、オファー ID を生成するには、次のプロセスを使用します。ドラフト状態で空白のオファーが作成されます。

新しいプライベートオファーの作成を開始するには

1. にサインインし[AWS Marketplace 管理ポータル](#)、オファー を選択します。
2. オファーページで、オファーの作成 を選択します。
3. オファーの作成ページで、プライベートオファーを作成する製品タイプと製品を選択します。処理には最大 30 秒かかります。このプロセス中は、ページを閉じたり更新したりしないでください。

Note

オファーの作成後に製品タイプと製品を変更することはできません。製品タイプごとのプライベートオファーの詳細については、[「サポートされている製品タイプ」](#)を参照してください。

AWS Marketplace チャンネルパートナーの場合は、再販承認から独自の製品または AWS Marketplace チャンネルパートナープライベートオファー (CPPO) のオファーを作成するかどうかを選択します。CPPO の場合は、独立系ソフトウェアベンダー (ISV)、製品、認可を選択します。

4. 続行 を選択して、オファーの詳細 を選択します。step-by-step エクスペリエンスが開き、プライベートオファーの作成を継続できます。

オファーのステータスを理解する

オファーには、ライフサイクルに応じて 3 つのステータスのいずれかがあります。

- ドラフト – オファーは不完全であり、まだ準備中です。オファーを公開して購入者に拡張するには、必要なすべての詳細を完了して送信する必要があります。
- アクティブ – オファーは公開され、購入者に提供されます。オファーの有効期限が切れていないため、購入者はオファーをサブスクライブできます。
- 有効期限切れ – オファーは公開され、購入者に提供されます。オファーの有効期限が切れているため、購入者はオファーをサブスクライブできません。有効期限を更新して、購入者がオファーを受け入れる時間を増やすことができます。オファーの有効期限を更新するには、[「プライベートオファーの有効期限の更新」](#)を参照してください。

Note

オファーが承諾されると、契約タブに契約として表示されます。オファーのステータスは変更されません。

プライベートオファーのドラフトと公開

プライベートオファーのドラフトと公開には、次のプロセスを使用します。

プライベートオファーをドラフトして公開するには

1. オファー情報の提供ページで、オファー名、オファーの詳細、更新タイプ、オファーの有効期限を指定します。更新オファーの場合は、 で作成された既存の契約を更新するための更新に の既存顧客 AWS Marketplaceを選択するか AWS Marketplace、既存の顧客を AWS Marketplaceに移行するための更新に の既存顧客を選択する必要があります AWS Marketplace。

Note

オファーの有効期限は、オファーが null になり無効になった日付です。この日付の 23:59:59 UTC 以降、購入者はこのプライベートオファーを表示して受け入れることができません。

2. [次へ] をクリックします。
3. オファーの料金と期間の設定ページで、料金モデル、契約期間または使用期間、料金、支払いスケジュールを選択します。分割プランがある料金モデルについては、[「分割プラン」](#)を参照してください。
4. 「購入者の追加」ページで、プライベートオファーを拡張する AWS Marketplace 購入者ごとに AWS アカウント ID を指定します。別の ID を追加するには、別の購入者を追加ボタンを選択します。AWS アカウント プライベートオファーごとに最大 24 人の購入者を追加できます。
5. [次へ] をクリックします。
6. 法的条件とオファードキュメントの設定ページで、次のいずれかのオプションを選択します。
 - パブリックオファーエンドユーザーライセンス契約 (EULA) — パブリックオファーの EULA を使用します。
 - AWS Marketplace (SCMP) の標準契約 – が提供する標準契約を使用します AWS Marketplace。
 - カスタム法的条件 – 法的条件、作業明細書、部品表、料金表、その他の付録など、プライベートオファーに関連するファイルを最大 5 つアップロードします。これらのファイルは、オファーの作成時に 1 つのドキュメントにマージされます。
7. 確認と作成ページで、プライベートオファーの詳細を確認します。確認後、オファーの作成を選択してオファーを公開し、選択した購入者に拡張します。オファーの公開には AWS Marketplace Catalog API へのリクエストが含まれているため、オファーの検証と処理に最大 1 時間かかる場合があります。このリクエストは、リクエストページで表示できます。

Note

オファーは、リクエストが成功した場合にのみ公開され、延長されます。リクエストが失敗した場合、そのリクエストはお客様に拡張されません。失敗とは、システムエラーが発生したか、再送信する前に修正する必要があるエラーがあることを意味します。

以下のガイドでは、特定の製品のプライベートオファーの作成について詳しく説明します。

- [AWS Marketplace – SaaS PAYG プライベートオファーを作成する](#)
- [AWS Marketplace – AMI 時間単位/ペイグプライベートオファーを作成する](#)
- [AWS Marketplace — 年間プライベートオファーを使用して AMI を時間単位で作成する](#)
- [AWS Marketplace – SaaS 契約プライベートオファーを作成する](#)

次の動画では、SaaS 契約プライベートオファーの作成について詳しく説明します。

購入者へのプライベートオファーの送信

プライベートオファーが公開されると、購入者は のプライベートオファーページの「利用可能なプライベートオファー」タブに移動してプライベートオファーを表示できます AWS Marketplace 管理ポータル。利用可能なプライベートオファー タブでは、購入者は、登録販売者 列で AWS Marketplace チャンネルパートナーによって拡張されたオファーを確認できます。独立系ソフトウェアベンダー (ISV) がパブリッシャー列に表示されます。購入者はオファーリストで適切な [オファー ID] を選択すると、プライベートオファーに移動できます。

購入者は、[承諾済みまたは有効期限切れのオファー] タブで承諾済みまたは有効期限切れのオファー ID を確認できます。

プライベートオファーが公開されたら、購入者にオファーのフルフィルメントページへの URL を送信できます。

購入者にプライベートオファーを送信するには

1. にサインインし [AWS Marketplace 管理ポータル](#)、オファー を選択します。
2. オファーの横にあるラジオボタンを選択します。
3. アクション を選択し、オファー URL をコピーします。

4. URL を購入者に送信します。

プライベートオファーの進行状況を保存する

次の料金を使用して進行状況を保存し、後で再開します。

作業を保存して再開するには

1. 完了したステップで、保存 を選択して を終了します。ダイアログボックスで、コンテンツをドラフト状態に保存し、検証エラーを確認します。検証エラーや詳細が欠落している場合は、修正を選択してステップに進み、問題を解決できます。準備ができたら、保存 を選択して終了します。

保存して終了すると、リクエストは処理中にレビューされます。処理の終了には、数分または数時間かかることがあります。リクエストが成功するまで、ステップを続行したり、変更を加えたりすることはできません。リクエストが成功したら、保存を完了します。リクエストが失敗した場合、システムエラーが発生したか、再送信する前に修正する必要があるエラーがありました。

2. オファーの作業を再開するには、オファーページを開き、オファーを選択し、オファー作成を再開を選択します。
3. 完了したら、保存して終了して進行状況を保存するか、オファーを作成してプライベートオファーを公開して選択した購入者に拡張するかを選択できます。

プライベートオファーの有効期限の更新

プライベートオファーの有効期限を更新するには、次のプロセスを使用します。

プライベートオファーの有効期限を更新するには

1. にサインインし [AWS Marketplace 管理ポータル](#)、オファー を選択します。
2. オファーページで、更新するオファーを選択します。
3. [編集] を選択します。
4. 新しいオファーの有効期限 を指定します。
5. [送信] を選択します。

更新が完了すると、オファーはアクティブステータスに変わり、購入者はオファーを受け入れることができます。

プライベートオファーのキャンセル

プライベートオファーをキャンセルするには、次のプロセスを使用します。

1. にサインインし [AWS Marketplace 管理ポータル](#)、オファー を選択します。
2. オファーページで、更新するオファーを選択します。

Note

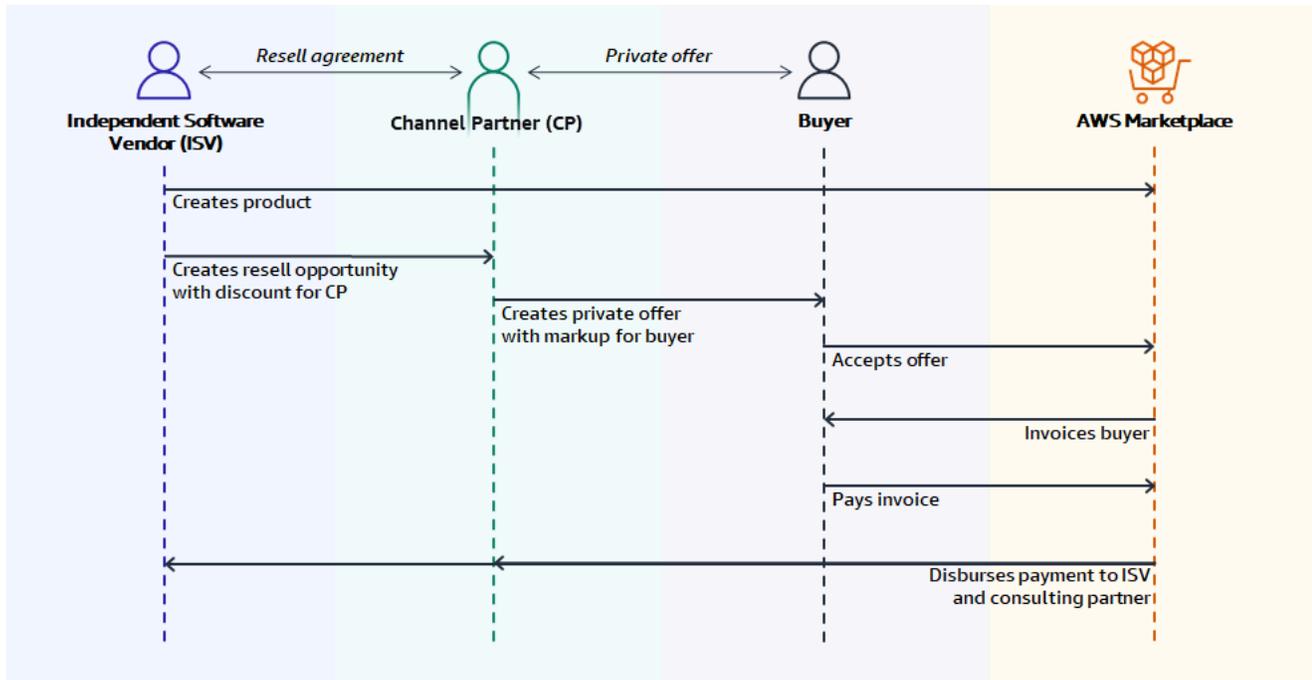
オファーをキャンセルするとオファーの有効期限が変更されるため、このオファーを延長した購入者にはオファーの有効期限が切れたものとして表示されます。

3. アクション を選択し、オファーのキャンセル を選択します。

AWS Marketplace チャンネルパートナーのプライベートオファー

AWS Marketplace チャンネルパートナーのプライベートオファーにより、チャンネルパートナーは独立したソフトウェアベンダーの (ISVs製品を で再販できます AWS Marketplace。AWS Marketplace チャンネルパートナーと ISV は、ISV の 1 つ以上の製品を再販する契約を結び、その製品の購入者にプライベートオファーを延長します。

次の図は、ISV、チャンネルパートナー、購入者の関係を示しています。



Note

ISV としてチャンネルパートナーに再販オポチュニティを作成する方法の詳細については、[「ISV としての AWS Marketplace チャンネルパートナーの再販機会の作成」](#)を参照してください。

各 AWS Marketplace チャンネルパートナーのプライベートオファーは 1 人の購入者にのみ表示され、その購入者のニーズを満たすようにカスタマイズされた料金と固有の商用条件が設定されています。プライベートオファーを作成する場合は、ISV が設定した卸売価格から開始します。次に、その価格を割り増しして購入者のオファー価格を作成します。卸売コストは、次の 2 つの方法のいずれかで決定されます。

- 経常割引 - ISV は、AWS Marketplace チャンネルパートナーが、経常オポチュニティを用いて定価から合意された割引価格で製品を再販することを認可します。この割引により、AWS Marketplace チャンネルパートナーは ISV との価格交渉を行わずに製品を再販し続けることができます。この割引は、指定した日付まで有効になるように設定することも、ISV またはチャンネルパートナーが終了するまで無期限に継続するように設定することもできます。
- 非定期割引 - ISV が AWS Marketplace チャンネルパートナーに提供する機会は、特定の購入者のみで使用することを目的とした 1 回限りの割引です。

いずれの場合も、購入者がプライベートオファーに対して支払いを行った後、は標準プロセス AWS Marketplace を使用して、合意された料金に基づいて資金を AWS Marketplace チャンネルパートナーと ISV に分配します。

Tip

ISV またはチャンネルパートナーは、取得または受け取ったオポチュニティを、[AWS Marketplace 管理ポータル](#)の [パートナー] メニューから確認できます。

プライベートオファーの作成方法の詳細については、[「AWS Marketplace チャンネルパートナーによる作成」](#)を参照してください。

プライベートオファーの第三者融資の詳細については、[「顧客向け融資が AWS Marketplace で利用可能になりました」](#)を参照してください。

追加情報

追加情報と質問については、ISVsとチャネルパートナーが AWS Marketplace チャネルチームに接続することをお勧めします。具体的な問い合わせ先が不明な場合は、aws-mp-channel@amazon.com まで E メールメッセージを送信してください。1 営業日以内にご連絡を差し上げます。

ISV としての AWS Marketplace チャネルパートナーの再販機会の作成

ISV は、チャネルパートナーに再販の機会を作成することで、AWS Marketplace そのパートナーに製品の再販を許可できます。製品ディメンションごとに割引率またはカスタム料金を指定して、チャネルパートナーの卸売価格 AWS Marketplace を作成できます。パートナーは、購入者の AWS Marketplace チャネルパートナーのプライベートオファーを作成するときに卸売価格をマークアップできます。サポートされている製品タイプは次のとおりです。

- AMI ベースの製品
- コンテナベースの製品
- SaaS ベースの製品
- プロフェッショナルサービス製品

以下の手順では、ISVs AWS Marketplace チャネルパートナーの機会を作成する方法の概要を説明します。この機能を使用するには、AWS Marketplace 管理ポータル の [パートナー] タブを使用する権限が必要です。詳細については、「[AWS Marketplace 販売者向けポリシー](#)」を参照してください。

ISV として AWS Marketplace チャネルパートナーにリセラーオポチュニティを作成するには

1. Seller AWS Marketplace アカウント [AWS Marketplace 管理ポータル](#) でサインインします。

Tip

Seller アカウントでサインイン AWS アカウント する前に、必ず別の AWS Marketplace からサインアウトしてください。

2. [パートナー] タブを選択してから、[Create opportunity (オポチュニティの作成)] を選択します
3. [オポチュニティの詳細] ページで、[オポチュニティの名前] と [オポチュニティの説明] を入力します。

Note

[オポチュニティの名前] と [オポチュニティの説明] に入力する情報は、チャンネルパートナーが販売者レポートで確認することができます。

- リセラーでは、ドロップダウンリストから承認する AWS Marketplace チャンネルパートナー (リセラー) を選択します。リセラーは名前またはアカウント ID で選択できます。
- このオポチュニティに含める製品のいずれかを選択します。
- 適用する [割引] を選択します。

[割引タイプ] は以下の複数の方法で公表できます。

- [割引率] - 選択したすべての製品に 1 つの割引率 (パーセンテージ) を適用します。
- [個別の料金] - 特定の製品に特定の割引を適用します。
- 柔軟な支払いスケジュール — AWS Marketplace チャンネルパートナーの機会に対する柔軟な支払いスケジュールを設定します。

Note

プロフェッショナルサービス販売者では、個別の料金と柔軟な支払いスケジュールのみがサポートされています。

- オポチュニティの [期間] を選択します。

Note

リセラーの開始日は、メーカーが再販認可に記載した日付より前でなければなりません。

[オポチュニティの時間の長さ] は以下の複数の方法で公表できます。

- 単一使用 - 1 つのオポチュニティに適用され、AWS Marketplace チャンネルパートナーがプライベートオファーを作成した後は適用されなくなります。
- [特定の期間] - ISV が選択した日付を過ぎると適用されなくなる特定の期間有効です。
- [期間設定なし] - 関係者のいずれかが終了するまで続きます。

8. (オプション) SaaS 契約製品の場合は、カスタムの[製品ディメンション]を追加または削除して、[追加使用量]を変更してオポチュニティをカスタマイズします。
9. (オプション) [購入者アカウント ID] を1つ以上設定して、オポチュニティがそれらの購入者のみを対象とするように指定します。
10. (オプション) [エンドユーザーライセンス契約 (EULA)] バージョンを選択するか、オポチュニティに含める EULA をアップロードします。

 Note

プロフェッショナルサービス販売者では、カスタム EULAs のみがサポートされています。

11. (オプション) [AWS Marketplace (RCMP) のリセラー契約] を選択するか、オポチュニティに含めるカスタム契約をアップロードします。
12. [オポチュニティを確認] を選択して、情報が正しいことを確認します。
13. [オポチュニティを作成] を選択してオポチュニティを確定し、チャンネルパートナーを承認します。

[作成済みのオポチュニティ] テーブルが更新されて、[オポチュニティの名前]、[製品名]、[リセラー名]、[割引]、[作成日時]、[ステータス] などを含み、関連するオポチュニティの詳細が表示されます。

オポチュニティが作成されると、その日付を拡張することはできません。ただし、オポチュニティはいつでも取り消して再作成できます。オポチュニティを取り消すと、新しいオファーではその割引を利用できません。既存のオファーは影響を受けず、オポチュニティの割引が保持されます。

オポチュニティを選択してから [クローン] を選択するとオポチュニティを複製することもできます。これによりすべてが事前入力され、フィールドを編集できるようになります。

分割プラン

分割払いプラン (Flexible Payment Schedule と呼ばれます) を使用すると、カスタム支払いスケジュールでプライベートオファーを拡張できます。スケジュールは、承諾された契約期間に分散でき、顧客は通常の分割払いで支払いを行います。サブスクライブすると、顧客はスケジュールと AWS 請求書ですべての支払いを確認でき、支出を追跡するのに役立ちます。分割プランは、特定の製品および料金タイプのプライベートオファーで利用できます。詳細については、「[プライベートオファーに適した製品のタイプ](#)」を参照してください。

支払いスケジュールの作成

分割払いプランを使用してカスタム支払いスケジュールを作成するプロセスは、プライベートオファーを作成するプロセスの一部です。プライベートオファーの分割プランオプションを有効にするには、プライベートオファーを作成するときに、購入者支払いオプションセクションの分割プランで契約料金を選択します。購入者支払いオプションを選択したら、このオファーの契約期間を選択し、オファーの詳細を指定します。詳細については、「[プライベートオファー](#)」を参照してください。

Note

分割プランによるプライベートオファーの場合、複数年およびカスタム期間の Amazon マシンイメージ (AMI) 製品の場合は、オファーに含まれる各インスタンスタイプのインスタンス数と、追加の起動されたインスタンスの時間単位の料金を設定します。お客様が指定された数のインスタンスを起動した後、起動された追加のインスタンスには、プライベートオファーで指定された時間料金で課金されます。

購入者分割プランで、契約合計、支払い数、初回請求日を指定し、分割プランの生成を選択します。最大 60 件の支払いを追加できます。また、各支払い明細項目を調整するオプションもあります。支払い明細項目を調整するたびに、購入者から支払うべき合計金額が更新されます。

Note

最初の分割払いの請求書日付は、プライベートオファーに対して顧客に初回請求書が送付される日付です。が顧客から支払いを受け取った後 AWS Marketplace、各請求書の支払いを受け取ります。

Flexible Payment Scheduler 機能によって、請求書日付が契約期間内に収まっていることが検証されます。請求書の日付が契約期間より後である場合は、エラーメッセージが表示されます。

すべての請求書の金額と日付を確認したら、購入者から支払う合計金額が、プライベートオファーの過程で顧客に支払う合計価格と一致していることを確認します。プライベートオファーの作成を完了するには、残りのステップを完了します。

Note

顧客に対して拡張しているオファー受諾日付前に設定できる請求書日付は 1 つだけです。

販売者が定義しているスケジュールに基づいて顧客に請求され、顧客がオファーを受諾した後に請求が開始されます。最初の請求日がオファーが承諾される前にスケジュールされている場合、オファーが承諾された直後に処理されます。

Note

拡張済みであり購入者がサブスクライブ済みのプライベートオファーの支払いスケジュールを変更することはできません。支払いスケジュールを変更するには、新しいオファーを作成する必要があります。

分割プランのレポート

柔軟な支払いスケジュールを持つプライベートオファーのレポートは、月次請求収益レポートの [セクション 4: Flexible Payment Schedule による契約](#) にあります。

プライベートオファーのアップグレード、更新、修正

プライベートオファーの契約の管理

Software as a Service (SaaS) 契約および消費製品との SaaS 契約では、アクティブな契約でプライベートオファーを使用することで、アップグレード、更新、修正を提供できます。これにより、新しい権利の付与、料金割引の提供、支払いスケジュールの調整、またはエンドユーザーライセンス契約 (EULA) の変更を行い、[標準化されたライセンス条項](#)を使用できます。ユニット数や支払いスケジュールを変更したり、カスタム終了日を追加したりすることもできます。

オファーと契約の違いは、購入者がその条件を承諾済みであるかどうかです。

- オファーは、購入者による製品の使用に関する一連の条件です。オファーは、パブリックまたはプライベートのいずれかです。
- 契約は、購入者が承諾したオファーです。契約には、販売者がパブリックオファーまたはプライベートオファーを使用して利用可能にした購入済み製品と無料製品が含まれます。

この機能は、独立系ソフトウェアベンダー (ISVs) やチャネルパートナーを含むすべての AWS Marketplace 販売者が利用できます。契約を修正して、元の契約の登録販売者とは異なる登録販売者を指定することはできません。

この機能を使用するには、AWS Marketplace 管理ポータル の [契約] タブを使用する権限が必要です。詳細については、「[AWS Marketplace 販売者のアクセス許可](#)」を参照してください。

プライベートオファ어의修正でサポートされる製品タイプ

契約ページで次の製品タイプを表示できます。

- SaaS 契約
- SaaS 契約 (追加使用量あり)
- SaaS 使用量ベースの製品
- AMI ベースの製品
- コンテナベースの製品
- サーバー契約
- プロフェッショナルサービス製品

ただし、プライベートオファ어의修正をサポートしているのは次の製品タイプのみです。

- SaaS 契約
- SaaS 契約 (追加使用量あり)

アップグレードと更新の提示プロセス

次の手順を使用して、 からプライベートオファ어의アップグレード、更新 AWS Marketplace 管理ポータル、および修正を作成できます。

プライベートオファ어의アップグレード、更新、修正を作成するには

1. [AWS Marketplace 管理ポータル](#) にサインインし、[Agreements (契約)] を選択します。
2. 契約ページで、契約の横にあるチェックボックスを選択し、詳細の表示を選択します。
3. 「契約を表示」ページで、「契約ベースのオファ어의作成」を選択します。
4. 契約詳細の修正ページで、販売者はプライベートオファ어가更新用かどうかを示す必要があります。プライベートオファ어가更新用かどうか尋ねられたら、はいまたはいいえを選択します。

Note

また、契約詳細の変更ページで、サービス日、製品ディメンション、支払いスケジュール、更新ステータス、オファーの有効期限を変更することもできます。

- 完了したら、オファーの作成 を選択し、 を送信します。

Tip

わかりやすいカスタムオファー名を入力すると、[Offers (オファー)] ページで有効なオファー間の区別に役立ちます。カスタムオファー名は購入者にも表示されます。AWS では、独自の IDs や発注書番号など、追加の識別情報を含むカスタムオファー名を指定することをお勧めします。**upgrade** や **renewal** などの概要、カスタム社名などの使用もお勧めします。個人を特定できるデータ (姓名、電話番号、住所など) は使用しないでください。このフィールドには最大 150 文字を入力できます。変更 EULAs する日付、ディメンション、支払いスケジュール、EULA の情報を編集し、次へ を選択します。[Review and create (確認と作成)] ページで、情報を確認します。準備ができたら、契約ベースのオファーの作成 を選択します。

新しいプライベートオファーは、約 45 分で [Manage Private Offer (プライベートオファーの管理)] ページに表示されます。オファーを表示するには、 にサインイン AWS Marketplace 管理ポータルし、オファーを選択してプライベートオファーの管理ページを開きます。

これにより、購入者は新しいオファーを承諾するか、引き続き元の契約に従うかを選択できます。

- 購入者がプライベートオファーのアップグレードまたは更新を受け入れると、新しい契約は直ちに有効になり、契約は AWS Marketplace 管理ポータルの [契約] ページに表示されます。以前の契約の残存する支払いスケジュールはキャンセルされます。

購入者は、プライベートオファーを承諾するのと同じ方法で、契約ベースのプライベートオファーを承諾します。プライベートオファーの購入者体験の詳細については、「AWS Marketplace 購入者ガイド」の「[プライベートオファー](#)」を参照してください。

- 購入者がプライベートオファーのアップグレードまたは更新を期限前に承諾しない場合、元の契約が変更なしで有効に存続します。

アップグレード、更新、修正に関するレポート

アップグレードと更新のプライベートオファーは、既存の販売者レポートとオファーに関連するレポートに表示されます。[日次顧客サブスクライバーレポート](#) レポートと [日別ビジネスレポート](#) レポートは毎日生成されます。[月別請求済み収益レポート](#) レポートは毎月生成されます。

日別顧客サブスクリプションレポートの [Subscription intent (サブスクリプションの目的)] フィールドは、レポートエントリが新しいプライベートオファーであるかどうかを示します。[Previous offer ID (以前のオファー ID)] フィールドは、新しいオファーに先立つ以前のオファーの ID (存在する場合) を表示します。すべてのプライベートオファーについて、エントリはプライベートとマークされます。

将来の日付の契約とプライベートオファー

の将来の日付の契約 (FDA) では AWS Marketplace、購入者は事前に決められた将来の日付に製品ライセンスまたは使用権限を受け取ります。一般的な AWS Marketplace 取引では、購入者はオファーが承諾された後または契約が作成された後すぐに、製品のライセンスまたは権利を受け取ります。

FDA を利用する販売者は、購入者が製品の使用を開始するときではなく、購入者が希望するときに取引を終了できます。FDA は、販売者がでのトランザクションに対して個別に以下のアクションを実行するのに役立ちます AWS Marketplace。

- 販売ニーズに基づいて取引を予約する (購入者がオファーを承諾する)。
- 財務または会計上のニーズに基づいて購入者に請求する。
- 購入者のニーズに基づいて、ライセンスや権利の有効化などの製品へのアクセスを購入者に提供する。

FDA を使用して、購入者との既存の取引の更新を設定できます。

FDA は、software as a service (SaaS) 製品契約、および Flexible Payment Schedule の有無にかかわらず使用料金付き契約 (CCP) をサポートしています。

将来の日付の契約を使用する場合は、以下の日付に注意します。

契約署名日

購入者がオファーを承諾し、契約が作成された日付。

契約開始日

購入者の製品に対するライセンスまたは権利が有効になり、購入者が製品の使用を開始できる日付。

契約終了日

契約が終了する日付。契約と、購入者のライセンスまたは権利は、この日に失効します。

トピック

- [将来の日付の契約の作成](#)
- [将来の日付の契約での分割プランの使用](#)
- [将来の日付の契約の通知を受け取る](#)
- [チャネルパートナーのプライベートオファーの再販に関する将来の日付の契約を使用する](#)

将来の日付の契約の作成

将来の開始日を設定したプライベートオファーを作成する場合、記録上の販売者が契約開始日を設定します。購入者は開始日を変更することはできませんが、AWS Marketplaceでプライベートオファーを承諾する前に開始日を確認することはできます。

将来の開始日のあるプライベートオファーを作成する

1. プライベートオファーを作成するときは、「契約期間」の「将来の日付で開始」を選択します。
2. [サービス日] セクションに、[サービスの開始日] と [サービスの終了日] を入力します。ここで選択したサービスの開始日は、購入者がオファーを承諾したときの、将来の日付の契約の契約開始日になります。

Note

FDA を更新に使用するには、サービスの開始日を、更新する契約の終了日に一致させます。

販売者は、サービスの開始日を 3 年先までの日付から選択できます。

将来の日付の契約での分割プランの使用

FDA で分割払いプランを使用すると、契約の署名日から契約終了日までの間に購入の支払いをいつでも設定できます。これには、契約開始日の前後の支払いも含まれます。

プライベートオファーの支払い日と金額は、記録上の販売者が選択します。分割プランの設定の詳細については、「」を参照してください[the section called “支払いスケジュールの作成”](#)。

将来の日付の契約の通知を受け取る

将来の日付の契約に対して実行された次のアクションについて、指定されたルートアカウントに[電子メール通知](#)が届きます。

- オファーの承認/契約の作成 (契約署名日)
- ライセンスまたは権利のアクティベーション時 (契約開始日)
- 30 日後、60 日後、または 90 日後に失効する契約に関する事前のリマインダー
- 契約失効日 (契約終了日)
- 契約の修正または置き換え時

Note

SaaS 向けの既存の Amazon Simple Notification Service (Amazon SNS) 通知はすべて、FDA でも機能します。FDAs、両方の Amazon SNS トピックは契約開始日に開始されます (契約署名日ではありません)。詳細については、「[the section called “SaaS 製品に関する Amazon SNS 通知”](#)」を参照してください。

チャネルパートナーのプライベートオファーの再販に関する将来の日付の契約を使用する

製造元とリセラーは、AWS Marketplace チャネルパートナーのプライベートオファーに将来の日付の契約を使用できます。

製造元として:

- 標準の AWS Marketplace Channel Partner Private Offers (CPPOsと同様に、製造元は AWS Marketplace チャネルパートナーに再販承認を延長して、将来の開始日で CPPOs を作成することを承認する必要があります。

再販認可を作成する方法の詳細については、[the section called “ISV としての再販オポチュニティの作成”](#) ページの手順に従ってください。

- 再販認可を作成する際、製造元はオプションでサービス開始日として認められる最大値を指定することができます。これは、対応するチャンネルパートナーのプライベートオファーを作成するときに AWS Marketplace チャンネルパートナーが指定できる最大サービス開始日 AWS Marketplace になります。

 Note

製造元が最大日付を指定しない場合、AWS Marketplace チャンネルパートナーは最大 3 年先までの将来のサービス日付を指定できます。

リセラーとして:

- リセラーとチャンネルパートナーの場合、将来の日付のチャンネルパートナーのプライベートオファーを作成する手順と、将来の日付の通常のプライベートオファーを作成する手順は同じですが、重要な違いが 1 つあります。リセラーが指定できる契約開始日は、製造元による再販認可で許可されるサービス開始日の最大値として指定されている日付よりも早い日付でなければなりません。
- チャンネルパートナーのプライベートオファーを作成する方法については、「[the section called “チャンネルパートナーのプライベートオファー”](#)」を参照してください。

AMI ベースの製品

購入者に製品を配信する方法の 1 つは、[Amazon マシンイメージ \(AMI\)](#) を使用することです。AMI は Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを起動するために必要な情報を提供します。製品用のカスタム AMI を作成すると、購入者はそれを使用して、製品がインストール済みですぐに使用できる EC2 インスタンスを作成できます。

購入者が提供した AMI を使用すると、製品用に作成した料金設定と計測オプションに従って、作成したインスタンスに対して請求されます。購入者は、AMI の新しいカスタムバージョンを作成するなど AWS、で他の AMIs を使用するのと同じ方法で製品 AMI を使用できます。AMI から作成された EC2 インスタンスは、引き続き AMI 製品コードに基づいて製品として請求されます。

以下のリソースを参照してください。

- AWS Marketplace 製品の料金の詳細については、「」を参照してください [製品の料金](#)。
- 製品のカスタム計測の作成の詳細については、「[AWS Marketplace Metering Service を使用した AMI 製品のカスタム計量](#)」を参照してください。

AMI ベースの製品配信方法

AMI ベースの製品は、次の 3 つの方法のいずれかで配信できます。

- 単一 AMI - 購入者は AMI を EC2 インスタンスのテンプレートとして選択して使用します。購入者は、Amazon マシンイメージ の配信方法フィルターを使用してこれらの製品を見つけることができます。

詳細については、「[単一 AMI 製品](#)」を参照してください。

- AWS CloudFormation テンプレート - 購入者が、異なるロールを持つ複数のインスタンスのシステムを単一のユニットとしてインストールできるようにするテンプレートを作成します。購入者は、CloudFormation 配信方法フィルターを使用してこれらの製品を見つけることができます。

詳細については、「[を使用した AMI ベースの配信 AWS CloudFormation](#)」を参照してください。

以下のリソースを参照してください。

- 購入者が AMI を使用する際の追跡方法の詳細については、「[AMI 製品コード](#)」を参照してください。

- AMI ベースの製品とそのライフサイクルの詳細については、「[AMI ベースの製品について](#)」を参照してください。

AMI ベースの製品について

このセクションでは、AMI ベースの製品を扱う際の重要な概念について説明します。

トピック

- [製品のライフサイクル](#)
- [AMI 製品コード](#)
- [変更リクエスト](#)
- [製品ロードフォーム](#)
- [年間契約の修正](#)

製品のライフサイクル

AMI ベースの製品には、ソフトウェアの 1 つ以上のバージョンのセットと、製品全体に関するメタデータが含まれます。製品を作成するときは、製品の名前、説明、料金 AWS Marketplace を含むプロパティを設定します。また、関連する検索で商品が表示されるように、商品に適したカテゴリを決定し、キーワードを追加します。

また、ソフトウェアの最初のバージョンが作成されます。ソフトウェアの提供方法に応じて、これは単一の AMI でも、AWS CloudFormation テンプレート付きの 1 つ以上の AMI でも、購入者が独自の AMI を作成する際に使用するソフトウェアパッケージでもかまいません。詳細については、「[AMI ベースの製品配信方法](#)」を参照してください。

有料製品の場合、購入者はインストールされたインスタンスの数に応じて課金されます。ソフトウェアが追跡する別のディメンション (製品のユーザー数など) を計測するには、製品を Metering Service AWS Marketplace と統合します。詳細については、「[AWS Marketplace Metering Service を使用した AMI 製品のカスタム計量](#)」を参照してください。

製品とソフトウェアの最初のバージョンを作成すると、最初は限られた範囲で公開され、自分のアカウントだけがアクセスできるようになります。準備ができたら、AWS Marketplace カタログに公開して、購入者が製品をサブスクライブして購入できるようにします。

[サーバー製品](#) ページでは、製品のリストを表示できます。ステージに応じて、製品のステータスは、次のいずれかです。

- [ステージング] - 情報を追加中の、不完全な製品。セルフサービス作成エクスペリエンスの最初の [保存して終了] で、変更リクエストが成功すると、送信したすべてのステップの情報を含む未公開の製品が作成されます。このステータスから、引き続き商品に情報を追加したり、変更リクエストを通じて送信済みの情報を変更したりできます。
- [制限あり] - 製品がシステムに送信され、システム内のすべての検証に合格すると完成します。その後、製品は [制限あり] 状態にリリースされます。この時点で、製品にはアカウントと許可リストに登録したユーザーだけがアクセスできる詳細ページがあります。商品は詳細ページからテストできます。必要に応じて、詳細やサポートが必要な場合は、[AWS Marketplace Seller Operations チーム](#)にお問い合わせください。
- [パブリック] - 購入者が商品を開覧してサブスクライブできるように商品を公開する準備ができたなら、[可視性を更新] 変更リクエストを使用します。これにより、AWS Marketplace Seller Operations チームが当社の[ポリシー](#)に照らして製品を確認および監査するワークフローが開始されます。製品が承認され、変更リクエストが処理されると、製品のステータスが [制限あり] から [パブリック] に移行します。
- [制限付き] - 新規ユーザーによる製品のサブスクライブを停止する場合は、[可視性を更新] 変更リクエストを使用して製品を制限できます。制限付き ステータスは、既存のユーザーが引き続き製品を使用できることを意味します。ただし、製品は一般には公開されなくなり、新規ユーザーも利用できなくなります。

の AMI ベースの製品のライフサイクル AWS Marketplace は、最初のバージョンを公開した後に終了しません。ソフトウェアの新しいバージョンと基本オペレーティングシステムのセキュリティパッチを適用して、製品を最新の状態に保つ必要があります。

完全な AMI ベースの製品のライフサイクルの例として、販売者が AMI ベースの製品を AWS Marketplace で販売することを考えているとします。以下は、販売者が長期にわたって製品を作成し、保守する方法です。

1. 製品の作成 - 販売者は製品を作成し、バージョン 1.0.0 を に公開します AWS Marketplace。購入者は、バージョン 1.0.0 のインスタンスを作成して使用できます。
2. 新しいバージョンを追加 - その後、販売者は製品に新しい機能を追加し、その機能を含む新しいバージョン 1.1.0 を追加します。購入者は、元のバージョン 1.0.0 を引き続き使用することも、新しいバージョン 1.1.0 を選択することもできます。

Note

新製品とは異なり、新バージョンは完全に公開されます。製品全体が限定リリースの場合にのみ、お客様が確認 AWS Marketplace せずに でテストできます。

3. 製品情報の更新 - バージョン 1.1.0 がリリースされたため、販売者は新機能を説明する新しいハイライトテキストで製品情報を更新して、購入者に新機能について知らせます。
4. マイナーバージョンの追加 - 販売者がバージョン 1.1.0 のバグを修正すると、新しいバージョン 1.1.1 を追加してリリースします。購入者は、バージョン 1.0.0、1.1.0、1.1.1 のいずれかを使用できるようにになりました。
5. バージョンを制限する - 販売者は、購入者にバージョン 1.1.0 を使用させたくないほどバグが深刻であると判断し、そのバージョンを制限します。その場合、新規顧客は 1.1.0 を購入できなくなります (1.0.0 または 1.1.1 しか選択できません)。ただし、既存の購入者は引き続き 1.1.0 にアクセスできます。
6. バージョン情報の更新 - 既存の購入者を支援するために、販売者は 1.1.0 のバージョン情報を更新し、バージョン 1.1.1 へのアップグレードを提案します。
7. 使用状況のモニタリング - 購入者が製品を購入して使用すると、販売者は を使用して売上、使用状況、およびその他のメトリクスをモニタリングします AWS Marketplace [出品者レポート](#)、[データフィード](#)、[ダッシュボード](#)。
8. 製品の削除 - 製品が不要になると、販売者はそれを から削除します AWS Marketplace。

この例では、販売者は製品に 3 つの異なるバージョンの AMI を作成しましたが、(製品を削除する前は) 新規購入者が利用できたのは 2 つのみでした。

バージョンや製品情報を変更するには、AWS Marketplace 管理ポータルで [変更リクエスト](#) を作成します。

AMI ベースの製品を作成および管理するステップの詳細については、「[単一 AMI 製品](#)」を参照してください。

AMI 製品コード

AWS Marketplaceで製品を作成すると、固有の製品コードが割り当てられます。この製品コードは製品の AMI に関連付けられ、製品の使用状況を追跡するために使用されます。製品コードは、購入者がソフトウェアを使用している間に自動的に伝播します。たとえば、お客様は AMI にサブスクリ

ブして起動し、それを構成したり、新しい AMI を生成します。新しい AMI には元の製品コードが含まれているため、依然として正しい使用量の追跡とアクセス許可が維持されます。

Note

製品コードは製品の製品 ID とは異なります。の各製品には、一意の製品 ID AWS Marketplace が割り当てられます。この製品 ID は、AWS Marketplace カタログ、顧客の請求、販売者レポートで製品を識別するために使用されます。製品コードは、AMI から作成されたインスタンスにインスタンスメタデータとして添付されます。その製品コードを含む AMI を使用してインスタンスを作成すると、顧客には関連する製品 ID が記載された請求書が届きます。製品を作成したら、製品の AWS Marketplace 管理ポータル ページで製品コードと製品 ID を見つけます。

販売者は、ソフトウェアが実行中の Amazon Elastic Compute Cloud (Amazon EC2) インスタンスの製品コードをランタイムにインスタンスのメタデータから取得できます。製品の開始時に製品コードを検証するなど、セキュリティを強化するために製品コードを使用できます。AMI の製品コードへの API コールは、製品がテストのために限定された状態で発行されるまでできません。製品コードの検証の詳細については、「[AWS Marketplace AMI でソフトウェアが実行されていることを確認する](#)」を参照してください。

変更リクエスト

で製品またはバージョンを変更するには AWS Marketplace、 を通じて変更リクエストを送信します AWS Marketplace 管理ポータル。変更リクエストはキューに追加され、リクエストのタイプによっては解決までに数分から数日かかる場合があります。リクエストのステータスは、AWS Marketplace 管理ポータルで確認できます。

Note

に加えて AWS Marketplace 管理ポータル、 [AWS Marketplace Catalog API](#) を使用して変更リクエストを作成することもできます。

AMI ベースの製品についてリクエストできる変更のタイプは次のとおりです。

- 購入者に表示される製品情報を更新する。
- 購入者に表示されるバージョン情報を更新する。

- 製品の新しいバージョンを追加する。
- 新しい購入者がそのバージョンにアクセスできないようにバージョンを制限する。
- AWS リージョン 製品が利用可能な を更新します。
- 製品の料金設定とインスタンスタイプを更新する。
- から製品を削除します AWS Marketplace。

詳細については、「[変更リクエストを作成する](#)」を参照してください。

Note

一部の更新リクエストでは、製品ロードフォームを使用してリクエストを作成する必要があります。詳細については、「[製品ロードフォーム](#)」を参照してください。

変更リクエストを更新する

更新で始まる変更リクエストには、現在の詳細が読み込まれます。その後、更新を行うと、既存の詳細が上書きされます。

変更リクエストを追加または制限する

追加リクエストと制限リクエストペアは、リクエストが成功するたびにプロビジョニングされるステップとアップデート専用です。セルフサービスエクスペリエンスで [保存して終了] とアクションを [送信] を選択すると、リクエストは成功します。

例えば、追加したインスタンスとリージョンに AMI アセットをプロビジョニングした場合、それらは完全に削除されるのではなく、制限することしかできません。つまり、既存のサブスクライバーとユーザーは、サブスクリプションまたは契約が終了するまで製品を使い続けることができます。ただし、[制限付き] ステータスの製品には新しいサブスクライバーを追加することはできません。

製品ロードフォーム

通常、製品を作成または編集するときは、AWS Marketplace 管理ポータル ユーザーインターフェイス内で作業して必要な変更を加えます。ただし、一部の操作では製品ロードフォーム (PLF) を使用するよう指示されます。

PLF は、製品に関するすべての情報を含むスプレッドシートです。PLF を入手する方法は複数あります。

- 既存の製品の PLF は、AWS Marketplace 管理ポータル製品の製品詳細ページからダウンロードできます。
- PLF を必要とするアクションのメニュー項目を選択すると、PLF をダウンロードするように求められます。例えば、毎月請求されるサーバー製品を新規作成する場合、適切な PLF をダウンロードするように求められます。

アクションが既存の製品の編集である場合、PLF にはその製品の情報が事前に入力されるため、更新する詳細を変更するだけで済みます。

- 新しい空白の PLF が必要な場合は、AWS Marketplace 管理ポータル [ファイルのアップロード](#) ページに、作成する製品のタイプに基づいて PLFs へのリンクがあります。

PLF が完了したら、AWS Marketplace 管理ポータル [ファイル](#) のアップロードページにアップロードします。PLF 自体の [手順] タブには、より詳細な説明があります。

年間契約の修正

時間単位の年間 (年間) プランの修正により、お客様と購入者は既存のプランに次の変更を加えることができます。

- Amazon EC2 インスタンスタイプファミリーの切り替え
- Amazon EC2 インスタンスタイプサイズを切り替える
- 新しいインスタンスタイプを追加する
- 契約内の既存のインスタンスタイプの数量を増やす

購入者は、変更の日割りコストが 0 より大きい (サブスクリプションの値を下げるできない) 限り、変更を行うことができます。新しく追加された Amazon EC2 インスタンスの日割り計算されたコストは、契約の残りの期間に合わせて調整されたインスタンスタイプの年間コストに基づきます。インスタンスタイプを切り替える場合、削除された Amazon EC2 インスタンスタイプの日割りコストは、新しく追加された Amazon EC2 インスタンスタイプの日割りコストから差し引かれます。

AMI 年間製品の修正を有効にするための追加のアクションは必要ありません。修正は、パブリックオファーから行われたすべての契約と、分割プランを使用しないプライベートオファーからの契約でサポートされます。

購入者による修正は、次のダッシュボードで確認できます。

- [契約と更新のダッシュボード](#) – 修正された契約のリスト。
- [請求収益ダッシュボード](#) – 顧客への請求。
- [回収と支払いダッシュボード](#) – 支払い。

単一 AMI 製品

このセクションでは、シングル AMI 製品ので製品リスト AWS Marketplace を操作する方法について説明します。お客様は Amazon マシンイメージ (AMI) を使用して、製品をインストールして設定済みの Amazon EC2 インスタンスを作成できます。

AMI のセルフサービスエクスペリエンスは、製品リストの作成や変更リクエストを行う際に役立ちます。セルフサービスエクスペリエンスを使用すると、AWS Marketplace Seller Operations チームによる処理に必要な時間を短縮して、製品リストを直接更新できます。セルフサービスエクスペリエンスの多くのステップは、のカタログシステムと一致しているため AWS Marketplace、AWS Marketplace Seller Operations チームによる処理と検証を待つ代わりに、直接検証が容易になります。

Note

いくつかのタスクでは、2つの手順から選択できます。1つはセルフサービスエクスペリエンスにのみ適した手順で、もう1つはセルフサービスエクスペリエンスには関係のない古い手順です。現在の移行期間中は、どちらの手順も使用できます。古い手順では、いつ利用できなくなるかについての情報が記載されたバナーが表示されます。

トピック

- [前提条件](#)
- [セルフサービスエクスペリエンスを理解する](#)
- [単一 AMI 製品の作成](#)
- [変更リクエストを作成する](#)
- [変更リクエストのステータスを取得する](#)
- [製品情報を更新する](#)
- [許可リスト \(プレビューアカウント\) を更新する](#)
- [製品の可視化の更新](#)
- [を追加する AWS リージョン](#)

- [の制限 AWS リージョン](#)
- [将来の AWS リージョンのサポートの更新](#)
- [インスタンスを追加する](#)
- [インスタンスの制限](#)
- [バージョン情報を更新する](#)
- [新しいバージョンを追加する](#)
- [バージョンを制限する](#)
- [料金を更新する](#)
- [国別の可用性の更新](#)
- [EULA を更新する](#)
- [返金ポリシーの更新](#)
- [AMI AWS Marketplace へのアクセスを許可する](#)
- [AWS Marketplaceから製品を削除する](#)
- [変更リクエストを送信する際によくあるエラーのトラブルシューティング](#)

前提条件

AMI 製品リストを作成するには、最初に次の前提条件を満たす必要があります。

1. AWS Marketplace 管理ポータルにアクセスできます。これは、販売者として登録し、で販売する製品を管理するために使用するツールです AWS Marketplace。へのアクセスの詳細については、AWS Marketplace 管理ポータル「」を参照してください[AWS Marketplace 販売者のポリシーとアクセス許可](#)。
2. 販売者として登録し、製品の代金を請求する場合は、税金と銀行情報を送信してください。販売者になる方法については、「[販売者としての開始方法](#)」を参照してください。
3. 売りたい製品がある。AMI ベースの製品の場合、これは通常、サーバーソフトウェアを作成または変更し、顧客が使用できる AMI を作成したことを意味します。で使用する AMI の準備の詳細については、AWS Marketplace 「」を参照してください[AMI 構築のベストプラクティス](#)。

セルフサービスエクスペリエンスを理解する

セルフサービスエクスペリエンスでは、で製品を作成する手順を説明します AWS Marketplace。ステップを進めるときは、製品情報と、AWS リージョンインスタンスタイプ、AMI の詳細などの

AMI デプロイ設定を指定します。また、料金、販売可能な国、EULA、返金ポリシーなどの取引の詳細も設定します。オプションとして、制限付きステータスの間、製品にアクセスしてテストする AWS アカウント IDs の許可リストを指定できます。

始める前に、セルフサービスエクスペリエンスの以下の重要な点を確認してください。

- 次のステップに進むには、現在のステップで必須フィールドに入力する必要があります。この要件は、各ステップの最後にページレベルの検証が行われるためです。不完全なステップは保存または送信できません。
- プロセスのすべてのステップを完了する前にセッションを終了する必要がある場合は、[保存して終了] を選択して、完了したステップをステージングエリアに送信できます。
- 不完全で検証に合格しなかったステップはシステムに送信されません。部分的に完了したステップは無効であり、保存することもできません。
- [保存して終了] を選択すると、[保存して終了] ダイアログボックスに検証チェックに合格したステップが表示されます。最後に完了して検証されたステップまで確認し、保存することを選択できます。検証エラーや詳細が不足している場合は、[修正] を選択してそのステップに戻ることができます。
- [保存して終了] すると、リクエストは処理中にレビューされます。処理の終了には、数分または数時間かかることがあります。リクエストが成功するまで、ステップを続行したり、変更を加えたりすることはできません。最初の [保存して終了] のリクエストは、完了したステップと並行して製品を作成します。
- リクエストが成功したら、保存は完了です。[製品の概要] ページで変更を再開するには、[製品の作成を再開する] を選択するか、[変更をリクエスト] を使用して前回のセッションで送信した詳細を更新します。再開すると、完了した手順に緑色の [成功] というラベルが付いていることに注目してください。以前に送信したステップを更新するには、[変更をリクエスト] を使用してください。このステップを続行するには、前回の [保存して終了] のリクエストを完了する必要があります。
- すべてのステップを完了したら、[次へ] を選択してレビューを確認できます。[送信] を選択すると、システムから最終検証の実行をリクエストできます。[成功] というレスポンスを受け取ると、製品は [制限あり] ステータスに移行します。詳細ページを確認すると、許可リストに登録されている人なら誰でもその製品を利用できるようになったことがわかります。リクエストが失敗した場合、製品は [ステージング] ステータスのままになり、再提出する前に修正を加える必要があります。

単一 AMI 製品の作成

このセクションでは、セルフサービスエクスペリエンスを使用するオプションを含め AWS Marketplace、で単一 AMI 製品のリストを作成するために使用できる手順について説明します。

トピック

- [セルフサービスを使用して単一 AMI 製品を作成する](#)
- [単一 AMI プロダクトの作成](#)
- [追加リソース](#)

セルフサービスを使用して単一 AMI 製品を作成する

以下の手順に従って、AWS Marketplaceで単一 AMI 製品を作成できます。

Note

次のステップに進むには、現在のステップで必須フィールドに入力する必要があります。未完了のステップは保存または送信できません。プロセスのすべてのステップを完了する前にセッションを終了する必要がある場合は、[保存して終了] 機能を使用して、完了したステップをステージングエリアに送信してください。詳細については、「[セルフサービスエクスペリエンスを理解する](#)」を参照してください。

セルフサービスを使用して単一 AMI 製品を作成するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [製品] メニューの [サーバー] を選択します。または、[\[サーバー製品\]](#) ページに直接移動することもできます。
3. [サーバー製品] タブから [サーバー製品の作成] を選択し、[Amazon マシンイメージ (AMI)] を選択して、単一 AMI 製品のライセンスタイプから 1 つを選択します。
 - Bring your own license (BYOL) – ユーザーが の外部からライセンスを取得する製品 AWS Marketplace。有料ライセンスでも無料ライセンスでもかまいません。
 - [無料] - サブスクリバラーが無料で使用できる製品。(関連する Amazon Elastic Compute Cloud (Amazon EC2) インスタンスまたはその他の AWS リソースに対しては引き続き料金が発生します。)

- 時間単位または時間単位の年間料金 – 購入者が時間単位または年間契約で時間単位で支払う製品。AMI AWS の製品コードに基づいて計測を行います。
 - [月額] - 購入者が AWSによって毎月請求される製品。
 - 有料使用料 - ソフトウェアの料金は、販売者がユーザー、データ、帯域幅、またはホストの 4 つの使用カテゴリのいずれかと併せて提供する価値に対して直接請求されます。製品に対して最大 24 のディメンションを定義できます。すべての料金は引き続きお客様のご負担となります。
 - 契約料金の AMI – 購入者が前払い料金を支払う AWS CloudFormation スタックを持つシングル AMI 製品またはシングル AMI。
4. セルフサービスエクスペリエンスでは、AWS Marketplace リストを作成する手順を案内します。製品情報 (メタデータ)、製品デプロイの詳細 (AWS リージョン、インスタンス、AMI の詳細)、およびパブリックオファーの詳細 (料金、EULA、国別の在庫状況、EULA、返金) を入力する必要があります。オプションとして、許可リストにアカウントを追加して製品をテストできます。各ステップを完了すると、プロセスの次のステップに進みます。

Note

プロセスのすべてのステップを完了する前にセッションを終了する必要がある場合は、[保存して終了] を使用して、完了したステップをステージングエリアに送信できます。これにより、入力した情報の検証を求めるリクエストが作成されます。リクエストが処理されている間は、製品を編集することはできません。リクエストが成功したら、[製品の作成を再開する] を選択して製品の作成を続行できます。

リクエストが失敗した場合は、検証エラーにより製品が更新されなかったことを意味します。これは製品のリクエストログに表示されます。リクエストを選択してエラーを表示し、[アクション] の [新規にコピー] を使用してエラーを修正し、リクエストを再送信できます。ステップを再開すると、前回のセッションで保存したステップの後のステップから続行できます。以前のステップを更新するには、製品概要ページに移動し、以前に送信したステップを更新する [\[リクエストを変更\]](#) を送信します。

5. すべての変更リクエストステップに必要な情報を入力したら、[送信] を選択します。この送信により、情報を検証し、検証に合格した場合に製品を [制限あり] 状態にリリースするよう AWS Marketplace カタログシステムにリクエストが作成されます。リクエストが処理されている間は、製品の編集を続けることはできません。リクエストが成功すると、製品は [制限あり] 状態に移行します。

- 製品が最初に公開されると、AWS アカウント（製品の作成に使用したもの）と AWS Marketplace Seller Operations チームのテストアカウントのみがアクセスできます。サーバー製品ページから製品を表示する場合は、[で表示](#) AWS Marketplace を選択して、AWS Marketplace 購入者に表示される製品の詳細を表示できます。この詳細リストは AWS Marketplace 他のユーザーには表示されません。
 - この機能により、製品を公開する前にテストできます（複数のバージョンをテスト用に公開することもできます）。
6. 限定状態で製品をテストし、AWS Marketplace [AMI ベースの製品要件と製品チェックリスト](#) に従っていることを確認します。次に、製品を [パブリック] に公開するようにリクエストするには、[可視性を更新] を選択します。Seller AWS Marketplace Operations チームは、公開を承認する前に製品を確認する必要があります。

Note

製品の検証と公開は手動プロセスであり、AWS Marketplace 販売者オペレーションチームが処理します。エラーがない場合は、最初の製品バージョンを公開するまでに 7 ~ 10 営業日かかることがあります。タイミングについては、「[タイミングと心構え](#)」を参照してください。

単一 AMI 製品と製品情報の両方を準備して提出する方法の詳細については、「[追加リソース](#)」を参照してください。

単一 AMI プロダクトの作成

Important

2023 年 7 月 14 日に、AWS Marketplace は以下の手順を中止します。2023 年 7 月 14 日以降は、[the section called “セルフサービスを使用して単一 AMI 製品を作成する”](#) の手順を使用してください。

単一 AMI 製品 (レガシー) を作成するには

1. AWS Marketplace 管理ポータル [で](https://aws.amazon.com/marketplace/management/tour/) を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。

2. [製品] メニューの [サーバー] を選択します。または、[\[サーバー製品\]](#) ページに直接移動することもできます。
3. [サーバー製品] タブから [サーバー製品の作成] を選択し、[Amazon マシンイメージ (AMI) (レガシーエクスペリエンス)] を選択して、単一 AMI 製品のライセンスタイプから 1 つを選択します。
 - Bring your own license (BYOL) – ユーザーが の外部からライセンスを取得する製品 AWS Marketplace。有料ライセンスでも無料ライセンスでもかまいません。
 - [無料] - サブスクライバーが無料で使用できる製品。(関連する Amazon Elastic Compute Cloud (Amazon EC2) インスタンスやその他の AWS リソースに対しては引き続き料金が発生します。)
 - 時間単位または時間単位の年間料金 – 購入者が時間単位または年間契約で時間単位で支払う製品。AMI AWS の製品コードに基づいて計測を行います。
 - [月額] - 購入者が AWSによって毎月請求される製品。[月額] を選択すると、製品積載フォーム (PLF) のダウンロードを求められます。
 - 有料使用料 - ソフトウェアの料金は、販売者がユーザー、データ、帯域幅、またはホストの 4 つの使用カテゴリのいずれかと併せて提供する価値に対して直接請求されます。製品に対して最大 24 のディメンションを定義できます。すべての料金は引き続きお客様のご負担となります。
 - 契約料金の AMI – 購入者が前払い料金を支払う AWS CloudFormation スタックを持つシングル AMI 製品またはシングル AMI。

Note

AMI ベースの製品には、セルフサービスベース以外の料金設定モデルもあります。これらの料金タイプは、製品を AWS Marketplace 計測サービスと統合して、顧客の使用状況に基づいてカスタム計測を提供する場合に適用されます。従量制の料金設定の製品を作成するには、製品積載フォーム (PLF) をダウンロード、記入、アップロードする必要があります。詳細については、「[AMI 製品のカスタム計量価格](#)」を参照してください。PLF の詳細については、「[製品ロードフォーム](#)」を参照してください。各種ライセンスの詳細については、「[AMI 料金モデル](#)」を参照してください。

4. 選択内容に基づいて新製品の情報を入力し、[送信] を選択します。
5. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。このページに戻って、処理中のリクエストのステータスを確認できます。

Note

製品の検証と公開は手動プロセスであり、AWS Marketplace Seller Operations チームが処理します。エラーがない場合は、最初の製品バージョンを公開するまでに 7~10 営業日かかることがあります。タイミングの詳細については、「[タイミングと心構え](#)」を参照してください。

製品が最初に公開された時点では、AWS アカウント (製品の作成に使用したアカウント) のみがアクセスできます。サーバー製品ページから製品を表示する場合は、[表示](#)を選択して AWS Marketplace、AWS Marketplace 購入者に表示される製品の詳細を表示できます。この詳細リストは AWS Marketplace 他のユーザーには表示されません。

この機能により、製品を公開する前にテストできます (複数のバージョンをテスト用に公開することもできます)。製品を他のテストアカウントでも利用できるようにする必要がある場合、または製品を一般に公開する必要がある場合は、[AWS Marketplace Seller Operations](#) チームに連絡してください。

単一 AMI 製品と製品情報の両方を準備して提出する方法の詳細については、「[追加リソース](#)」を参照してください。

追加リソース

製品に関する情報を準備し、公開用に送信する方法の詳細については、次のリソースを参照してください。

- [製品の準備](#)
- [掲載のための製品の送信](#)

単一 AMI 製品を AWS Marketplace に送信する準備の詳細については、以下のリソースを参照してください。

- [AMI 構築のベストプラクティス](#)
- [AMI 製品チェックリスト](#)
- [AMI ベースの製品要件](#)

変更リクエストを作成する

このセクションでは、セルフサービスエクスペリエンスを使用するオプションを含め AWS Marketplace、 で単一 AMI 製品の変更リクエストを作成するために使用できる手順について説明します。以下の状況では、変更リクエストを作成します。

- セルフサービスエクスペリエンスを使用して単一 AMI 製品リストを作成しているときに、進行中のステップを保存したが、プロセス全体を完了していません。残りのステップを完了するには、変更リクエストを作成します。
- [制限あり] または [パブリック] 状態の製品の製品情報を変更します。情報を更新するには、変更リクエストを作成します。AMI ベースの製品に対してリクエストできる変更のタイプの詳細については、「[変更リクエスト](#)」を参照してください。

Note

に加えて AWS Marketplace 管理ポータル、 [AWS Marketplace Catalog API](#) を使用して変更リクエストを作成することもできます。

トピック

- [変更リクエストを作成する](#)
- [セルフサービスを使用して変更リクエストを作成します。](#)
- [追加リソース](#)

変更リクエストを作成する

Important

2023 年 6 月 15 日に、AWS Marketplace は以下の手順を中止します。2023 年 6 月 15 日以降は、[the section called “セルフサービスを使用して変更リクエストを作成します。”](#) の手順を使用してください。

バージョンや製品情報を変更するには、AWS Marketplace 管理ポータルで変更リクエストを作成します。

変更リクエストを作成するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインしてから、[サーバー製品](#) ページに移動します。
2. [サーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンリストからオプションを選択します。

ほとんどの変更リクエストでは、ユーザーインターフェイスのフォームに記入して送信するだけです。ただし、変更によっては、製品積載フォーム (PLF) をダウンロードして記入し、アップロードする必要があります。これは必要な情報を記入するためのフォームが入ったスプレッドシートです。これらの変更リクエストのいずれかを選択すると、作成しようとしているリクエストに対応する正しい PLF をダウンロードするように求められます。PLF には、既存の製品詳細の情報が事前に入力されています。完成した PLF を AWS Marketplace 管理ポータル [ファイルのアップロード](#) ページにアップロードできます。

Note

最新の PLF をダウンロードして使用することを強くお勧めします。フォームは、インスタンスタイプや利用可能になった AWS リージョン ときに、新しい情報で定期的に更新されます。製品の最新の PLF は、[サーバー製品] ページで製品を選択し、[製品積載フォームをダウンロード] を選択すると確認できます。

変更リクエストのステータスの詳細については、「[the section called “変更リクエストのステータスを取得する”](#)」を参照してください。変更リクエストで発生する可能性のある問題については、「[変更リクエストを送信する際によくあるエラーのトラブルシューティング](#)」を参照してください。

セルフサービスを使用して変更リクエストを作成します。

バージョンや製品情報を変更するには、AWS Marketplace 管理ポータルで変更リクエストを作成します。変更リクエストは、製品に変更を加える際に使用するセルフサービスリストの構成要素です。ステップから [保存して終了] を選択して手順を終了するか、更新の [送信] を選択するたびに、変更リクエストを送信することになります。リクエストは AWS Marketplace 管理ポータル、[リクエスト](#) タブで確認できます。

セルフサービスを使用して変更リクエストを作成するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインしてから、[サーバー製品](#) ページに移動します。

2. [サーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンからオプションを選択します。
4. 変更リクエストを行った後、システムがリクエストを処理するまで待ち時間があり、[レビュー中] と表示されます。リクエストが完了すると、[成功] または [失敗] と表示されます。
 - リクエストが送信されると、[レビュー中]、[変更を準備中]、[変更を適用中] のステータスで処理が開始されます。
 - [成功] とは、リクエストした変更が処理され、変更がシステムに反映されたことを意味します。
 - [失敗] とは、リクエストに何か問題が生じたため、変更が処理されなかったことを意味します。ステータスが、[失敗] の場合は、リクエストを選択して、エラーの修正方法に関する推奨事項を示すエラーコードを検索できます。この時点で、エラーのトラブルシューティングを行い、変更の新しいリクエストを作成できます。処理を速めるには、[新しいリクエストにコピー] を選択して失敗したリクエストの詳細をコピーできます。その後、調整を行ってリクエストを再送信できます。

追加リソース

特定のタイプの更新の変更リクエストの詳細については、以下のリソースを参照してください。

- [製品情報を更新する](#)
- [バージョン情報を更新する](#)
- [新しいバージョンを追加する](#)
- [バージョンを制限する](#)

変更リクエストのステータスを取得する

Important

2023年6月15日に、AWS Marketplace は以下の手順を中止します。この手順は、セルフサービスエクスペリエンスでは不要になります。

変更リクエストを送信すると、AWS Marketplace 管理ポータル [\[サーバー製品\]](#) ページの [リクエスト] タブでリクエストのステータスを確認できます。ステータスは、次のいずれかになります。

- [レビュー中]とは、リクエストがレビュー中であることを意味します。一部のリクエストでは、AWS Marketplace チームによる手動によるレビューが必要ですが、ほとんどのリクエストはシステムで自動的にレビューされます。
- [成功]とは、リクエストが完了したことを意味します。製品またはバージョンがリクエストどおりに更新されました。
- [アクションが必要]とは、問題を解決したり、リクエストに関する質問に回答したりするために、リクエストを更新する必要があるということを意味します。リクエストを選択すると、問題を含む詳細が表示されます。
- [失敗]とは、リクエストに何か問題があったことを意味し、同じデータを使用して新しい変更リクエストを作成する必要があります。

製品情報を更新する

製品を作成した後に、AWS Marketplaceでその製品に関連する情報の一部を変更する必要がある場合があります。例えば、新しいバージョンで製品の説明やハイライトが変更された場合は、新しいデータで製品情報を編集できます。

製品情報を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[サーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [製品情報を更新] を選択します。
4. 変更する必要がある次のフィールドのいずれかを更新します。
 - [製品のタイトル]
 - [SKU]
 - 短い説明
 - [詳しい説明]
 - [製品ロゴイメージ URL]
 - [ハイライト]
 - [製品カテゴリ]
 - キーワード
 - [製品動画 URL]

- リソース
- [サポート情報]

Note

ロゴフォーマットの詳細については、「[会社と製品のロゴ要件](#)」を参照してください。

5. [Submit] (送信) を選択します。
6. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。リストにリクエストが表示されるようにするには、ページの更新が必要になる場合があります。

リクエストのステータスは、[サーバー製品](#) ページの [リクエスト] タブからいつでも確認できます。詳細については、「[変更リクエストのステータスを取得する](#)」を参照してください。

許可リスト (プレビューアカウント) を更新する

制限付き状態で製品を表示できる AWS アカウント IDs のリストを変更するには、更新許可リストを使用します。

許可リストを更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [サーバー製品](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [許可リストを更新] を選択します。現在のリストには、現在、許可リストに登録されているアカウントのリストが表示されます。
4. 表示に推奨される AWS アカウント IDs を追加し、IDsをカンマで区切ります。
5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

製品の可視化の更新

で製品を表示できる購入者を変更するには AWS Marketplace、可視性の更新 を使用できます。

可視性を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [可視性を更新] を選択します。

Note

この変更リクエストを使用して、製品を [制限あり] ステータスから [パブリック] ステータスに移行するようリクエストできます。ただし、変更リクエストをパブリックに移行するには、AWS Marketplace Seller Operations チームの承認プロセスを経る必要があります。

4. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
5. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

を追加する AWS リージョン

購入者が製品を使用できるリージョンを追加できます。

リージョンを追加するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [リージョンを追加] を選択します。
4. 使用可能なリージョンのリストから、追加するリージョンを選択します。
5. [リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

Note

新しいのサポートを追加すると AWS リージョン、製品のプライベートオファーに既にサブスクライブしているお客様は、新しく追加されたリージョンに自動的にアクセスできなくなります。顧客がアクセスするリージョンで別のプライベートオファーを作成する必要があります。新しいオファーを承諾すると、お客様は新しく追加されたリージョンにアクセスできます。将来の日付に製品をサブスクライブするお客様は、リージョンがプライベートオファーに含まれている限り、リージョンにアクセスすることもできます。新しいプライベートオファーを作成する方法の詳細については、[「プライベートオファーのアップグレード、更新、および修正」](#)を参照してください。

の制限 AWS リージョン

新しい購入者が特定の で製品を使用できないようにするには AWS リージョン、リージョンを制限できます。リージョンは後で追加し直すことができます。リージョン内の製品の既存のサブスクライバーは、サブスクライブしている限り、そのリージョンの製品を引き続き使用できます。

リージョンを制限するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [リージョンを制限] を選択します。
4. ドロップダウンメニューを選択すると、製品が現在販売されているリージョンのリストが表示されます。
5. 制限するリージョンを選択します。
6. 選択したリージョンはトークンとして表示されます。制限するリージョンのリストを確認し、制限しないリージョンには X を入力します。
7. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
8. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

リクエストが成功すると、既存のユーザーには、制限されるリージョンを通知する次の E メールメッセージが届きます。登録している間は製品を使い続けることができますが、サブスクライブをキャンセルすると再サブスクライブすることはできません。

Greetings from AWS Marketplace,

This message is a notification detailing a recent change for <ProductName>. {{{sellerName}}} has opted to restrict the <ProductType> product in <Restricted Region(s)> beginning <DateOfChange>.

This impacts you in the following ways:

1. As long as you're subscribed to the product, you can continue using the software product in the restricted Region.
2. You can't begin new instances of the software product in the restricted Region.
3. You can continue using the software product in all available AWS Regions.

Regards,
The AWS Marketplace Team

Amazon Web Services, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com (<http://amazon.com/>) is a registered trademark of Amazon.com, Inc. This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave. North, Seattle, WA 98109-5210.

将来の AWS リージョンのサポートの更新

製品を新しく起動した にオンボードする場合は AWS リージョン、将来のリージョンサポートの更新 を使用できます。

将来のリージョンのサポートを更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [将来のリージョンのサポートの更新] を選択します。
4. 今後のリージョンサポートをアクティブ化して、AWS Marketplace が AWS リージョン ユーザーに代わって新しく起動された に製品をオンボードできるようにすることができます。
5. この機能を有効にすると、将来リリースされるすべてのリージョンを選択するか、米国リージョンのみに限定するかを選択できます。
6. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
7. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

インスタンスを追加する

購入者が単一 AMI を使用できる新しいインスタンスを追加できます。

インスタンスを追加するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[\[現在のサーバー製品\]](#) タブで、変更する製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンから [\[インスタンスを追加\]](#) を選択します。
4. 使用可能なインスタンスのリストから追加するインスタンスを選択します。
5. [\[リクエストを送信\]](#) を選択して、レビューのリクエストを送信します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) になります。

Note

現在の料金モデルが無料ではない場合や、Bring Your Own License (BYOL) モデルを使用している場合は、料金を追加する追加の手順があります。

新しいインスタンスの料金を指定して [\[インスタンスを追加\]](#) を作成した場合、または料金を引き上げるために [\[料金を更新\]](#) した場合は、変更を行った日から 90 日間はセルフサービスを使用して [\[インスタンスを追加\]](#) することはできません。これらの変更を行うには、[AWS Marketplace Seller Operations チーム](#) に連絡してください。

新しいインスタンスタイプのサポートを追加すると、製品のプライベートオファーに既にサブスクライブしているお客様は、新しく追加されたインスタンスを自動的に起動できなくなります。顧客がアクセスするインスタンスで別のプライベートオファーを作成する必要があります。新しいオファーを承諾すると、お客様は新しく追加されたインスタンスを起動できます。将来の日付に製品をサブスクライブするお客様は、インスタンスがプライベートオファーに含まれている限り、インスタンスを起動することもできます。新しいプライベートオファーを作成する方法の詳細については、[「プライベートオファーのアップグレード、更新、および修正」](#) を参照してください。

インスタンスの制限

新規購入者が特定のインスタンスの 1 つの AMI 製品を使用できないようにする場合は、インスタンスを制限できます。インスタンスは、必要に応じて後で追加し直すことができます。制限付きインス

タンス上の単一 AMI の既存のユーザーは、サブスクライブしている限り、そのリージョンの製品を引き続き使用できます。

インスタンスを制限するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [インスタンスの制限] を選択します。
4. 制限するインスタンスを選択し、[制限] を選択します。
5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

Note

チェックボックスがグレー表示されている場合は、そのインスタンスが推奨インスタンスタイプとして1つまたは複数のバージョンに関連付けられていることを意味します。このようなインスタンスを制限するには、[バージョンの更新] を使用して別の推奨インスタンスタイプを選択してください。変更リクエストが完了し、制限するインスタンスが推奨インスタンスタイプではなくなったら、[インスタンスの制限] に戻って選択したインスタンスを制限できます。

バージョン情報を更新する

バージョンが作成されたら、そのバージョンに関連する情報を変更して、更新された情報を購入者に提供すると便利です。例えば、バージョン 1.1 がリリースされた後にバージョン 1.0 を制限する予定であれば、バージョン 1.0 の説明を更新して、そのバージョンが制限される日付を指定して購入者をバージョン 1.1 に誘導できます。バージョン情報は AWS Marketplace 管理ポータルから更新します。

バージョン情報を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[サーバー製品] タブで、変更する製品を選択します。

3. [変更をリクエスト] ドロップダウンから [バージョン情報を更新] を選択します。
4. [バージョンを更新] ページで、更新するバージョンを選択します。
5. 変更する必要がある次の情報を更新します。
 - リリースノート
 - [使用手順]
 - 64 ビット (x86) Amazon マシンイメージ (AMI) - 使用状況とセキュリティグループの詳細
6. [Submit] (送信) を選択します。
7. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。

Note

この手順を使用して、バージョンタイトルやバージョンに関連付けられている AMI を更新することはできません。代わりに、[新しいバージョンを作成し、以前のバージョンを制限してください。](#)

リクエストのステータスは、[サーバー製品](#) ページの [リクエスト] タブからいつでも確認できます。詳細については、「[変更リクエストのステータスを取得する](#)」を参照してください。

新しいバージョンを追加する

製品、ベースイメージに変更を加えたとき、または製品の AMI を変更する必要があるときはいつでも、製品の新しいバージョンを追加できます。から製品の新しいバージョンを追加します AWS Marketplace 管理ポータル。

Note

の AMI の作成については、AWS Marketplace 「」を参照してください [AMI 構築のベストプラクティス](#)。

新しいバージョンを追加するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。

2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [新しいバージョンを追加] を選択します。[新しいバージョンを追加] フォームが表示され、最新バージョンの情報が事前入力されます。
4. [バージョン情報] セクションで、次の情報を入力します。
 - [バージョンタイトル] - 有効な文字列 (**1.1** や **##### 2.0** など) を入力します。製品全体で一意である必要があります。
 - [リリースノート] - このバージョンに関する詳細を説明するテキストを入力します。
5. [新しい Amazon マシンイメージ (AMI)] セクションで、以下の情報を入力します。
 - [Amazon マシンイメージ ID] - このバージョンに使用する AMI の AMI ID を入力します。AMI ID は、[コンソールの AMI のリスト](#) から確認できます。AMI は、米国東部 (バージニア北部) リージョンと AWS Marketplace 販売者アカウントに存在する必要があります。この AMI に関連付けられているスナップショットは暗号化できません。
 - IAM アクセスロール ARN - が AMI にアクセス AWS Marketplace できるようにする (IAM) ロールの Amazon リソースネーム AWS Identity and Access Management (ARN) を入力します。IAM ロールの作成手順については、「[AMI AWS Marketplace へのアクセスを許可する](#)」を参照してください。IAM ARN の標準形式を使用します。例: **arn:aws:iam::123456789012:role/RoleName**。ARN は AWS Marketplace Seller アカウントに存在する必要があります。
 - [OS ユーザー名] - Linux ベースの AMI の場合は、インスタンスへのサインインに使用できるユーザーの名前を入力します。ec2-user を使用することをお勧めします。
 - [スキャンポート] - オペレーティングシステムへのログインに使用できるポート番号 (Linux AMI の場合は SSH ポート、Windows AMI の場合は RDP ポート) を入力します。
6. まだの場合は、設定を展開して AMI を AWS Marketplace カスタマーウェブサイトセクションに公開し、次の情報を入力します。
 - [使用手順] - AMI の使用方法、または AMI の使用に関する詳細情報へのリンクを入力します。例: **#####https://example.com/usage.htm #####**
 - [エンドポイント URL] - 購入者がインスタンスを作成した後にソフトウェアにアクセスする方法についての情報を入力します。購入者が製品へのアクセスに使用できる [プロトコル] (https または http)、相対 URL (例: **/index.html**)、およびポート (例:**443**) を入力します。(ホスト名は EC2 インスタンスによって異なるため、相対パスのみを指定する必要があります)。
 - [オペレーティングシステム (OS)] - AMI が使用する OS の名前 (**Amazon Linux** など) を入力します。

- [OS バージョン] - AMI の OS の特定のバージョンを入力します。
 - [推奨インスタンスタイプ] - 購入者がデフォルトで利用できるインスタンスタイプを選択します。
 - [セキュリティグループの推奨事項] - プロトコル (TCP または UDP)、許可するポートの範囲、IPv4 CIDR IP のリスト (xxx.xxx.xxx.xxx/nn の形式、例えば **192.0.2.0/24**) など、1 つ以上の推奨事項の情報を入力します。
7. [送信] を選択し、新しいバージョンを追加するリクエストを送信します。
 8. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。修正する必要のあるエラーがある場合、ページ上部の表にエラーが表示され、更新が必要な特定のフィールドは赤で表示されます。

リクエストのステータスは、[\[サーバー製品\]](#) ページの [リクエスト] タブからいつでも確認できます。新しいバージョンはレビューされ、承認された場合は製品の新しい公開バージョンとして公開されます。問題がある場合は、ステータスが [アクションが必要] になる場合があります。リクエストを選択すると、問題を含む詳細が表示されます。

リクエストが成功すると、既存のユーザーには以下の E メールメッセージが届きます。このメッセージは、新しいバージョンが利用可能になったことをユーザーに通知し、そのバージョンのリリースノートにリンクし、最新バージョンへのアップグレードを提案します。AWS アカウント ルートユーザーとして、に関連付けられている E メールアカウントにも E メールメッセージのコピーを受け取ります AWS アカウント。

Greetings from AWS Marketplace,

Thank you for subscribing to <product-title>

We are writing to inform you that <seller-name> has added a new version to <product-title> on AWS Marketplace.

As an existing customer, your subscription to the product, any running instances and access to previous versions are unaffected. However, <seller-name> does recommend you to update to the latest version, <product-title>/<version-title> by visiting <product-detail-page-of-new-listing>.

For additional questions or upgrade information, please contact <seller-name> directly. Click here <link of seller page on MP> to visit the seller's profile page on AWS Marketplace.

Release notes for <product-title>/<version-title>:

```
<release-notes>
```

```
Thank you,  
The AWS Marketplace Team  
https://aws.amazon.com/marketplace
```

```
Amazon Web Services, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a  
registered trademark of Amazon.com, Inc.  
This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave.  
North, Seattle, WA 98109-5210
```

バージョンを制限する

購入者が公開している製品の特定のバージョンにアクセスできないようにする場合は、そのバージョンを制限できます。

Note

制限状況にかかわらず、すべてのサブスクライバーが現在のバージョンを使用できます。AWS Marketplace ガイドラインでは、バージョンを制限してから 90 日間、既存の購入者へのサポートを継続することが義務付けられています。バージョンが制限されると、AMI は非推奨としてマークされます。詳細については、「Windows インスタンス用 Amazon Elastic Compute Cloud ユーザーガイド」の「[AMI を非推奨にする](#)」を参照してください。

バージョンを制限するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [バージョンを制限] を選択します。
4. [バージョンを制限] ページで、制限するバージョン (1 つまたは複数) を選択します。
5. [送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

Note

製品のすべてのバージョンを制限することはできません。最後に残っているパブリックバージョンの製品を制限しようとすると、エラーが発生します。製品を完全に削除する方法については、「[the section called “AWS Marketplaceから製品を削除する”](#)」を参照してください。

リクエストのステータスは、[\[サーバー製品\]](#) ページの [リクエスト] タブからいつでも確認できます。詳細については、「[変更リクエストのステータスを取得する](#)」を参照してください。

Note

バージョンの制限には、最大で3日かかることがあります。

リクエストが成功すると、既存のユーザーには、バージョン制限を通知し、入手可能な最新バージョンを使用することを勧める次の E メールメッセージが届きます。AWS アカウント ルートユーザーには、に関連付けられている E メールアカウントにも E メールメッセージのコピーが送信されます AWS アカウント。

Greetings from AWS Marketplace,

Thank you for subscribing to <product-title>.

We are writing to inform you that, as of <Version-Restriction-Date>, <Seller Name> will no longer offer version(s) "<version-title>" to new subscribers. Your use and subscription is unaffected for this version(s), however it is recommended that users upgrade to the latest version on AWS Marketplace.

For additional questions or upgrade information, please contact <seller-name> directly. Click here<link of seller page on MP> to visit the seller's profile page on AWS Marketplace.

Thank you,
The AWS Marketplace Team
<https://aws.amazon.com/marketplace>

Amazon Web Services, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a registered trademark of Amazon.com, Inc. This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave. North, Seattle, WA 98109-5210

料金を更新する

単一 AMI 製品のインスタンスタイプごとの料金を変更する場合は、料金を更新できます。

Note

新しいインスタンスの料金を指定して [インスタンスの追加] または [料金を更新] して料金を引き上げること (料金を下げるのではなく) をリクエストした場合、変更を加えた日から 90 日間は、セルフサービスを使用して料金を更新することはできません。さらにサポートが必要な場合は、[AWS Marketplace Seller Operations チーム](#)に連絡して変更を行ってください。

製品料金を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [料金を更新] を選択します。
4. テキストボックスには、現在の料金情報が事前に入力されています。現在の料金を削除して、新しい料金を入力できます。リクエストしている料金を見直して、正しい料金を確認することをお勧めします。

Note

いずれかのインスタンスの料金を引き上げると、今後 90 日間は [インスタンスを追加] と [料金を更新] がロックされます。

5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

Note

新しいインスタンスの料金を指定して [インスタンスを追加] を作成した場合、または料金を引き上げて (料金を引き下げるのではなく) [料金を更新] した場合、変更を加えた日から 90 日間は、セルフサービスを使用して料金を更新することはできません。さらにサポートが必要な場合は、[AWS Marketplace Seller Operations チーム](#)に連絡して変更を行ってください。

国別の可用性の更新

製品をサブスクライブして提供できる国を変更する場合は、[可用性の更新] を使用してください。

国別の可用性を更新するには

1. AWS Marketplace 管理ポータルで <https://aws.amazon.com/marketplace/management/tour/> を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [可用性の更新] を選択します。
4. 以下のオプションのいずれかを選択します。
 1. [すべての国] - サポートされているすべての国で利用できます。
 2. [除外対象のすべての国] - 選択した国を除くすべてのサポート対象国で利用できます。
 3. [カスタムリスト] - 製品が販売されている特定の国のリスト。
5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

EULA を更新する

エンドユーザー使用許諾契約 (EULA) を変更する場合は、[EULA の更新] を使用できます。このアクションにより、製品を購読する新規ユーザーおよび製品の更新の EULA が更新されます。このアクションにより、製品をサブスクライブする新規ユーザーおよび製品更新の EULA が更新されます。

EULA を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) タブを選択し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから、[エンドユーザー使用許諾契約の更新] を選択します。
4. [AWS Marketplace の標準契約 \(SCMP\)](#) を選択するか、カスタム EULA を送信してください。カスタム EULA の場合は、Amazon S3 バケットからのカスタム契約を提供する必要があります。

Note

Amazon S3 バケットでパブリックアクセシビリティを有効にする必要があります。

5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

返金ポリシーの更新

製品の返金ポリシーを変更する場合は、[返金ポリシーを更新] を使用できます。

返金ポリシーを更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページに移動し、[現在のサーバー製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [返金ポリシーを更新] を選択します。
4. 現在の返金ポリシーの詳細がテキストボックスに表示され、編集できます。リクエストを送信すると、現在の返金ポリシーが上書きされます。
5. [変更リクエストを送信] を選択して、レビューのリクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

AMI AWS Marketplace へのアクセスを許可する

への新しい AMI の追加を含むリクエストを作成する場合 AWS Marketplace、AMI を AWS Marketplace システムにコピーし、セキュリティ上の問題がないかスキャンする必要があります。AMI でアクションを実行するアクセス許可を持つ AWS Identity and Access Management (IAM) ロールと、 がロールを引き受け AWS Marketplace を許可する信頼ポリシーを作成して、AMI へのアクセスを許可 AWS Marketplace する必要があります。IAM ロールは一度作成するだけで済みます。

AWS Marketplace AMI アセット取り込み用のロールを作成するには

1. にサインインし AWS Management Console、IAM コンソールを開き、[ロールページ](#) に移動します。
2. [Create role] (ロールの作成) を選択します。
3. [ロールを作成] ページで、以下の選択を行います。
 - [信頼されたエンティティのタイプの選択] で、[AWS のサービス] を選択します。
 - [ユースケースの選択] - [AWS Marketplace] を選択します。
 - [ユースケースの選択] - [Marketplace - AMI アセットの取り込み] を選択します。
 - 次のページに移動して [次: アクセス] ページに移動します。
4. [AWSMarketplaceAmiIngestion] ポリシーを選択します。必要に応じてアクセス許可の境界を追加し、[次へ: タグ] を選択して続行します。

Note

アクセス許可の境界を使用して、このロールで AWS Marketplace に与えるアクセスを制限できます。詳細については、「AWS Identity and Access Management ユーザーガイド」の「[IAM エンティティのアクセス許可の境界](#)」を参照してください。

5. 続行するには、[次へ: レビュー] を選択します。
6. ロールの名前を入力し、[ロールを作成] を選択します。
7. ページの上部に「ロールの####が作成されました」と表示され、ロールがロールのリストに表示されます。

このページで、作成したロールを選択すると、その ARN が `arn:aws:iam::123456789012:role/exampleRole` という形式で表示されます。製品に[新しいバージョンを追加する](#)場合など、変更リクエストを作成するときは、[IAM アクセスロール ARN] に ARN を使用します。

AWS Marketplaceから製品を削除する

製品が公開されたら、から削除 (サンセット とも呼ばれます) できます AWS Marketplace。製品を削除するには、製品を特定して削除する理由と連絡先の E メールアドレスと一緒に削除リクエストを送信します。また、現在の製品を新しい製品と交換する場合は、代替製品 ID を提供することもできます。製品の削除をリクエストした後、新しいお客様は製品にサブスクライブできなくなります。既存のお客様は、最低 90 日間サポートする必要があります。からの製品削除のリクエスト AWS Marketplace は、以下の条件で処理されます。

- 製品は、AWS Marketplace 検索、参照、その他の検出ツールから削除されます。[サブスクライブ] ボタンまたは機能は無効になっており、ページ上のメッセージは製品が使用できなくなったことを明確に示しています。製品の詳細ページは引き続き URL を使用してアクセスでき、公開されている検索エンジンでインデックスに登録されている可能性があります。
- 削除の理由を指定する必要があります (サポートの終了、製品の更新の終了、交換製品など)。削除された製品に対する継続的なサポートの要件については、「[AWS Marketplace 出品者の利用規約](#)」を参照してください。
- AWS Marketplace は、製品の削除、削除の理由、販売者の連絡先情報を知らせる E メールメッセージを通じて、現在の購入者に連絡します。
- 現在の購入者がサブスクリプションをキャンセルするまでソフトウェアへのアクセスは維持されます。これらのお客様は、製品の削除による影響を受けません。

を使用して作成された製品を削除するには AWS Marketplace 管理ポータル

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [製品] タブを選択し、次に [サーバー] を選択します。
3. [現在のサーバー製品] の下にある製品ページで、削除する製品を探します。リクエストの変更ドロップダウンリストから、製品の可視性の更新を選択します。
4. 製品可視性の更新ページで、制限付き を選択します。
5. (オプション) 削除する製品の代わりとなる別の製品がある場合は、[代替製品 ID] を入力します。
6. 情報が正しいかどうかを確認し、「送信」を選択します。

製品の削除リクエストを送信した後、[What's next (最新情報)] 情報ページが表示されます。Seller AWS Marketplace Operations は、リクエストを確認して処理します。[リクエスト] を表示して、送信のステータスを確認します。

製品が削除されると、製品は AWS Marketplace 管理ポータル の [現在の製品] リストに表示されます。[Current Products (現在の製品)] で利用できる唯一のアクションは製品のスプレッドシートをダウンロードすることです。別のサンセットリクエストの編集または送信はできません。

製品の削除についてご不明な点がございましたら、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

変更リクエストを送信する際によくあるエラーのトラブルシューティング

製品に関する情報を変更すると、エラーが発生する可能性があります。以下は、一般的なシナリオとその修正方法です。

- AMI のスキャン AMI のスキャン中にいくつかの問題が発生する可能性があります。
 - AMI をスキャンする AWS Marketplace アクセス許可が付与されていません。AWS Marketplace にアクセス許可を付与してください。または、アクセス許可を付与しましたが、アクセス許可の境界が制限されすぎています。詳細については、「[AMI AWS Marketplace へのアクセスを許可する](#)」を参照してください。
 - スキャンで AMI にセキュリティ上の問題や一般的な脆弱性と漏えい (CVE) が見つかった場合は、イメージ内のオペレーティングシステム用の最新のパッチを使用していることを確認してください。詳細については、「[AMI ベースの製品要件](#)」を参照してください。

AMI の構築に関する一般的なガイドラインについては、「[AMI 構築のベストプラクティス](#)」を参照してください。

- AWS Marketplace 管理ポータル fields - の一部のフィールドには、非常に具体的な情報 AWS Marketplace 管理ポータル が必要です。
 - フィールドが何をリクエストしているのかわからない場合は、コンソールで詳細を確認してみてください。ほとんどのフィールドには、フィールドの上にはテキストによる説明があり、フィールドの下にはフォーマット要件があります。
 - 1 つ以上の無効なフィールドを含むフォームを送信しようとする、問題のリストが表示されます。問題の解決に役立つ推奨アクションが表示されます。
 - ARN の入力を求められた場合、通常、これはコンソールの別の場所にあります。例えば、AMI AWS Marketplace へのアクセスを許可するために作成した IAM ロールの ARN は、IAM コン

ソールの[ロールページ](#)にあります。ARN はすべて同様の形式です。例えば、IAM ロール ARN は、arn:aws:iam::123456789012:role/exampleRole というような形式になります。

- ロゴと動画は、コンテンツへの URL として直接提供する必要があります。ロゴ形式の詳細については、「[会社と製品のロゴ要件](#)」を参照してください。

製品およびバージョン変更リクエストの送信について詳しくは、「[掲載のための製品の送信](#)」を参照してください。

- 製品積載フォーム (PLF) の問題 - PLF には、スプレッドシートに含まれる指示が含まれています。全体的な説明は説明表に記載されています。各フィールドには入力方法が記載されています。フィールドを選択すると指示が表示されます。
- 進行中のリクエスト - 一部のリクエストは並行して実行できません。製品の特定の処理中の情報を更新するリクエストは、一度に 1 つしか送信できません。レビュー中のすべてのリクエストは、AWS Marketplace 管理ポータル の [サーバー製品] ページの [リクエスト] タブで確認できます。意図しない保留中のリクエストがある場合は、そのリクエストをキャンセルして、必要な変更を加えた新しいリクエストを送信できます。
- バージョンを更新 (追加または制限する) が進行中の場合は、バージョン情報を更新できません。
- Seller AWS Marketplace Operations チームから保留中のリクエストがある場合、新しい変更を送信することはできません。
- 原因不明のエラー - 説明もなく送信が失敗した場合は、もう一度行ってください。サーバーの負荷により送信が失敗することがあります。

それでも変更リクエストに問題がある場合は、[AWS Marketplace Seller Operations](#) チームに連絡してください。

を使用した AMI ベースの配信 AWS CloudFormation

Important

AWS Marketplace は、2024 年 8 月に AWS CloudFormation テンプレートを使用して、複数の Amazon マシンイメージ (AMI) 製品の配信方法を中止します。この配信方法は、廃止されるまでの間、既存のサブスクライバーのみが利用できます。を使用した単一の AMI など CloudFormation、を使用する他の AWS Marketplace 製品は影響を受け CloudFormation ません。詳細については、「AWS Marketplace 購入者ガイド」の「[AWS CloudFormation テンプレート](#)」を参照してください。

AWS Marketplace 販売者は、AWS CloudFormation テンプレートを使用して AWS Marketplace 購入者に配信される AMI ベースの製品を一覧表示できます。テンプレートを使用して、各製品の分散アーキテクチャまたはクラスターを定義する、または異なる AMI の組み合わせや製品設定を選択できます。CloudFormation テンプレートは、単一の Amazon マシンイメージ (AMI) または複数の AMIs、関連する設定ファイルおよび Lambda 関数とともに配信するように設定できます。購入者は、で選択したソリューションを参照し AWS Marketplace、ワンクリックで購入し、指定したテンプレートを使用して CloudFormation デプロイできます。

マルチ AMI ソリューションには、最大 20 AMIs と最大 20 個の CloudFormation テンプレートを含めることができます。各 CloudFormation テンプレートは、ソリューションに含まれる AMIs の任意の組み合わせまたはサブセットを参照できます。購入者は、1 つのソリューションを購入すると、その製品に含まれているすべての AMI に対する権利を付与されます。製品に複数の AMI があるとき、各 AMI に独自の製品コードがあり、価格設定と測定を別々に実行できます。ただし、ソリューションの個々のコンポーネントは、製品のコンテキスト外で検出可能つまり購買可能ではありません。

既存の単一 AMI 製品がある場合は、製品を新しいマルチ AMI 出品に移行または結合することはできません。ただし、新しいソリューションは、既存の製品を使用した、AMI のソフトウェアまたはコピーと同じ特徴を持つことができます。で作成された各リスト AWS Marketplace は、新しい製品コードを含むリストです。

購入者が を通じてデプロイできるように、AMI でサーバーレスアプリケーションに Lambda 関数を含めることもできます CloudFormation。Lambda 関数とサーバーレスアプリケーションを AMI に含める手順については、「[サーバーレスアプリケーションコンポーネントの追加](#)」を参照してください。

製品の出品の構築

製品を送信するには、(複数) AMI を準備して検証し、(複数) AWS CloudFormation テンプレートを作成して、アーキテクチャ図を作成します。さらに Product Load Form を完了して、AWS Marketplace にマテリアルを送信する必要があります。まず AMIs、CloudFormation テンプレートを完了して検証することをお勧めします。これらの手順が完了したら、アーキテクチャ図を作成し、ソフトウェアとインフラストラクチャの料金を見積もる必要があります。AWS Marketplace は、送信内容を検証し、販売者と連携して製品を公開します。[AWS Pricing Calculator](#) を使用すると、テンプレートのインフラストラクチャのコスト見積もりに役立ちます。保存した計算ツール設定へのリンク AWS Marketplace を に提供します。マルチ AMI ソリューションの製品には、以下の制限があります。

- 既存の AWS Marketplace 製品をスタンドアロン製品からマルチ AMI 製品に更新することはサポートされていません。製品をマルチ AMI 製品で利用可能にするには、AMI をコピーし、コンポーネ

ントとして新しいマルチ AMI 製品に送信します。生成された AMI には、前回の製品のコードとは異なる一意の製品コードが割り当てられます。

- マルチ AMI ソリューションは、Amazon Elastic Compute Cloud (Amazon EC2) コンソールの [インスタンスの起動] ページにある [AWS Marketplace] タブには表示されません。
- CloudFormation テンプレートは、マルチ AMIs ソリューションに記載されているもの以外の AMI を起動しないでください。
- CloudFormation テンプレートはパブリック URL の形式で送信する必要があります。テンプレートに含まれているネストされたすべてのテンプレート URL もパブリックにアクセス可能である必要があります。

CloudFormation テンプレートの準備

CloudFormation テンプレートを構築するには、テンプレートの前提条件を満たし、必要な入力パラメータとセキュリティパラメータを指定する必要があります。CloudFormation テンプレートを送信するときは、以下のセクションのガイドラインを使用してください。

テンプレートの前提条件

- 製品で有効になっているすべてので、AWS CloudFormation コンソールからテンプレートが正常に起動されたことを確認します。AWS リージョン ツールを使用してテンプレートを [TaskCat](#) テストできます。
- 単一の AMI 製品を作成する場合は、テンプレートに 1 つの AMI のみが含まれている必要があります。
- AMI は各リージョンの [マッピングテーブル](#) に存在する必要があります。AWS Marketplace AMI IDs は、クローン作成後に更新されます。ソース AMI が us-east-1 にあり、他のリージョンがプレースホルダーを使用できる必要があります。次の YAML の例を参照してください。

```
Mappings:
  RegionMap:
    us-east-1:
      ImageId: ami-0123456789abcdef0
    us-west-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxx
    eu-west-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxx
    ap-southeast-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxx
```

- 特定のアベイラビリティゾーン (AZ) の使用に依存しないようにテンプレートを構築します。すべてのアベイラビリティゾーンにすべての顧客がアクセスできるとは限りません。AZ はさまざまなアカウントに別々にマッピングされます。
- AMI には Lambda 関数、設定ファイル、スクリプトなどの依存関係を含めることができます。詳細については、「[サーバーレスアプリケーションを作成する](#)」を参照してください。
- Auto Scaling グループを使用してクラスター化されたソリューションを構築している場合は、スケールイベントを考慮することをお勧めします。新規ノードは、自動的に動作クラスターを結合する必要があります。
- 単一ノードの製品であっても、[Auto Scaling グループ](#)を使用することをお勧めします。
- ソリューションに、複数のインスタンスのクラスターが含まれる場合で、インスタンス間でネットワークレイテンシーが低い、またはネットワークスループットが高い、あるいはその両方の場合、プレイメントグループを使用することを検討してください。
- ソリューションに Docker コンテナが含まれる場合は、AMI に Docker イメージを組み込む必要があります。
- AWS Marketplace チームによるレビューと顧客への透明性を高めるため、UserDataセクションにコメントを追加することをお勧めします。

テンプレート入力パラメータ

- テンプレートへの入力パラメータには、AWS Marketplace 顧客の認証情報 (パスワード、パブリックキー、プライベートキー、証明書など) AWS を含めないでください。
- パスワードなどの機密入力パラメータには、NoEcho プロパティを選択し、強力な正規表現を有効にします。他の入力パラメータでは、最も一般的な入力を適切なヘルパーテキストで設定します。
- 使用可能な入力には AWS CloudFormation パラメータタイプを使用します。
- `AWS::CloudFormation::Interface` を使用し入力パラメータをグループ化およびソートします。
- 以下の入力パラメータにはデフォルト値を設定しないでください。

Note

顧客は、これらを入力パラメータとして指定する必要があります。

- パブリックインターネットからリモートアクセスポートへの進入を許可するデフォルトの CIDR 範囲
- パブリックインターネットからデータベース接続ポートへの進入を許可するデフォルトの CIDR 範囲
- ユーザーまたはデータベースのデフォルトパスワード

ネットワークおよびセキュリティパラメータ

- デフォルトの SSH ポート (22) または RDP ポート (3389) が 0.0.0.0 に対して開かれていないことを確認します。
- デフォルトの VPC (仮想プライベートクラウド) を使用する代わりに、適切な ACL (アクセスコントロールリスト) とセキュリティグループを設定した VPC を作成することをお勧めします。
- AWS Identity and Access Management (IAM) ロールを使用して [AssumeRole](#) から を呼び出すことで、お客様の AWS 環境へのアクセスを有効にします AWS Security Token Service。
- IAM ロールとポリシーに [最小特権を付与](#) し、絶対に必要な場合のみに書き込みアクセスを有効にします。たとえば、アプリケーションで S3:GET、PUT および DELETE オペレーションのみが必要な場合に、これらのアクションのみを指定します。この場合に、S3:* の使用はお勧めしません。

テンプレートを受信すると、 は製品設定と情報 AWS Marketplace を検証し、必要なリビジョンに関するフィードバックを提供します。

テンプレートインフラストラクチャの料金見積りの取得

顧客に提示される各テンプレートのインフラストラクチャの料金見積もりは、[AWS Pricing Calculator](#) を使用して指定した見積もりに基づくものです。この見積もりには、一般的なデプロイ向けのデフォルト値とともにテンプレートの一部として、デプロイされるサービスのリストが含まれている必要があります。

テンプレートの推定月額コストを計算したら、米国東部 (バージニア北 AWS Marketplace 部) リージョンの Save and Share リンクを提供します。これは、送信プロセスの一部です。

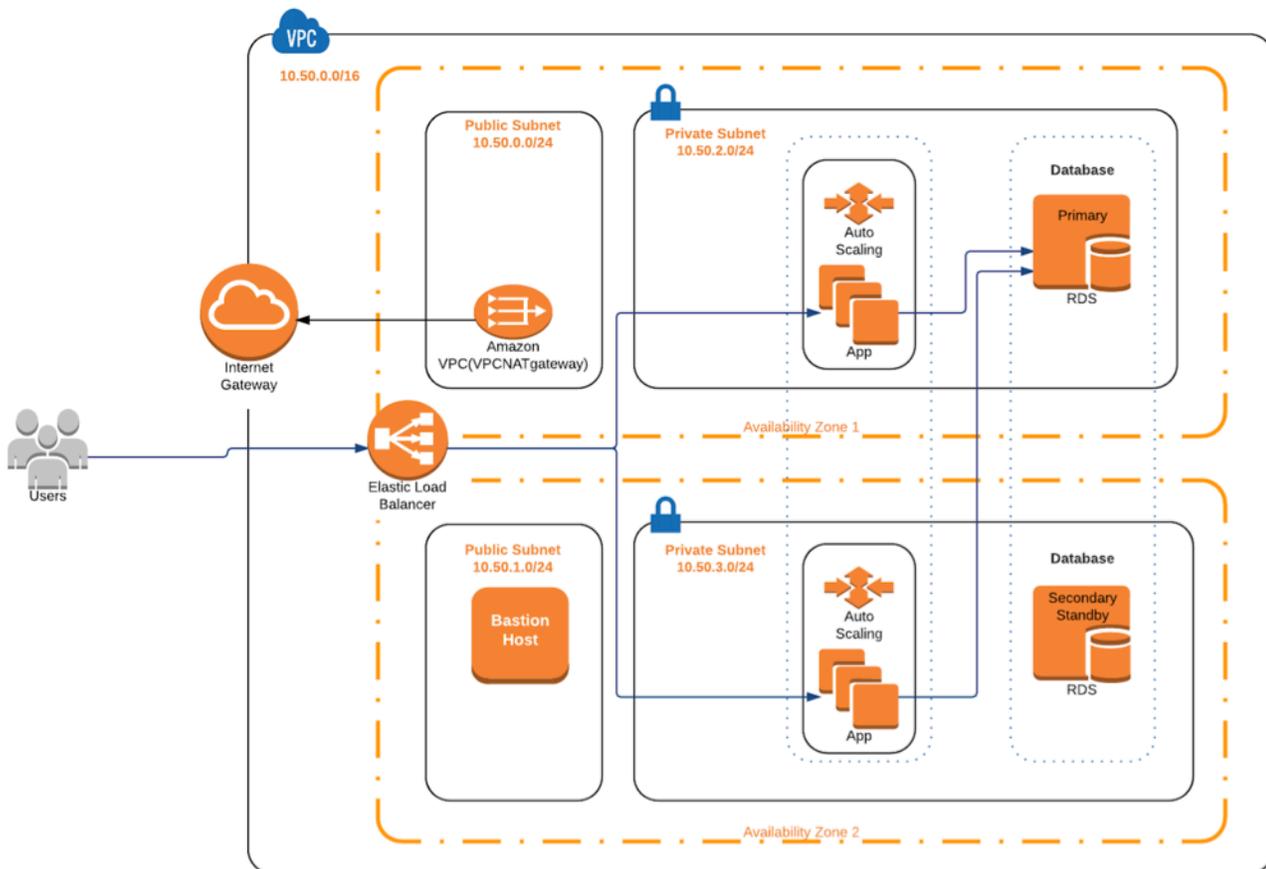
アーキテクチャ図

各テンプレートのアーキテクチャ図を用意する必要があります。図は次の基準を満たしている必要があります。

- での標準デプロイ (複数可) の図 AWS
- AWS CloudFormation テンプレートを介して AWS のサービス デプロイされた各 [AWS の製品アイコン](#)を使用する
- AWS CloudFormation テンプレートによってデプロイされたすべてのサービスのメタデータを含める
- AWS CloudFormation テンプレートによってデプロイされたすべてのネットワーク、VPCs、サブネットを含める
- サードパーティーのアセット、API およびオンプレミス、ハイブリッドアセットを含む、統合ポイントを表示
- サイズが 1100 x 700 ピクセルである

Note

トリミングや拡大が必要ないように、次に示すとおり、図がこのサイズ要件を満たしていることを確認します。



送信要件の一致

AWS CloudFormation テンプレートを使用して によって配信される製品を送信するには、次のリソースを指定する必要があります。

- CloudFormation テンプレート
 - 単一 AMI 製品には 1~3 つの CloudFormation テンプレートを含めることができます
 - マルチ AMI 製品には最大 20 個の CloudFormation テンプレートを含めることができます
- 各テンプレートのデフォルト設定に対するインフラストラクチャの見積もり価格。
- アーキテクチャ図とアーキテクチャメタデータ
- 入力済みの Product Form ([AWS Marketplace 管理ポータル](#) から利用可能)
 - 単一 AMI 製品では、[商用製品](#) フォームを使用します。
 - マルチ AMI 製品では、[Multi-AMI Product](#) フォームを使用します。

製品フォームには、リファレンスとして送信情報の例が含まれます。

各製品では、必須製品データとメタデータの大半は、従来の単一 AMI 製品の場合と同じです。したがって、テンプレートを使用して配信される各 AMI CloudFormation は、で説明されている標準と要件を満たし続ける必要があります AWS Marketplace。

テンプレートごとに CloudFormation、次の情報も指定する必要があります。

フィールド	説明	制限事項
タイトル	アーキテクチャのタイトル。これは詳細ページとフルフィルメントページ、およびアーキテクチャ詳細を表示するポップアップに表示されません。	50 文字
短い説明	詳細およびフルフィルメントのページに表示されます。	200 文字
長い説明	これはアーキテクチャ詳細ポップアップに表示されません。	2000 文字

マルチ AMI 製品については、以下のフィールドが必須です。

- ソリューションのタイトル
- ソリューションの短い説明
- ソリューションの長い説明
- CloudFormation テンプレートの場合 (ソリューションあたり最大 20)
 - デプロイメントタイトル (テンプレートごと)
 - 短い説明 (テンプレートごと)
 - 長い説明 (テンプレートごと)
 - アーキテクチャの図 (テンプレートごと)
 - インフラストラクチャの料金表見積もり (テンプレートごと)
 - この CloudFormation テンプレートに含まれる製品/コンポーネントのリスト
 - この CloudFormation テンプレートでサポートされているリージョンのリスト

製品リクエストの送信

[AWS Marketplace 管理ポータル](#) を使用して、製品を送信します。[アセット] タブで、[ファイルのアップロード] を選択します。送信するファイルをアップロードし、簡単な説明を入力します。YAML と JSON の両方のファイル形式に対応しています。リクエストの処理には 3~5 週間かかります。これには、以下の処理が含まれます。

- AMI と CloudFormation テンプレートのテンプレート、AMI CloudFormation、メタデータの確認
- AWS Marketplace 製品への CloudFormation テンプレートの発行

サーバーレスアプリケーションコンポーネントの追加

1 つ以上の AWS CloudFormation テンプレートを使用して配信される 1 つ以上の Amazon マシンイメージ (AMIs) を含む製品を作成し、サーバーレスコンポーネントを製品に組み込むことができます。例えば、1 つの AMI をコントローラーサーバーとして設定し、別の AMI をワーカーサーバーとして設定して、AWS CloudFormation スタックとして配信する製品を作成します。スタックの作成に使用される AWS CloudFormation テンプレートには、いずれかのサーバーのイベントによってトリガーされる AWS Lambda 関数を設定する定義を含めることができます。

このアプローチを使用して製品を設計すると、アーキテクチャを簡素化し、購入者が簡単に起動できるようになります。また、このアプローチにより、製品の更新が容易になります。

製品用の AMI の作成の詳細については、「[AMI ベースの製品](#)」を参照してください。製品の AWS CloudFormation テンプレートを完了する方法については、「[」を参照してください](#) [を使用した AMI ベースの配信 AWS CloudFormation](#)。

サーバーレスアプリケーションを定義するときは、に保存する AWS Serverless Application Model (AWS SAM) テンプレートを使用します AWS Serverless Application Repository。は、サーバーレスアプリケーションを構築するためのオープンソースフレームワーク AWS SAM です。デプロイ中に、は AWS Serverless Application Model 構文を AWS SAM 変換して AWS CloudFormation 構文に拡張します。AWS Serverless Application Repository は、サーバーレスアプリケーション用のマネージド型のリポジトリです。再利用可能なアプリケーションを保存して共有できるため、購入者はサーバーレスアーキテクチャを構築してデプロイできます。このタイプの製品を作成して提供するには、次の手順を実行します。

ステップ

- [サーバーレスアプリケーションを作成する](#)
- [アプリケーションをリポジトリに公開する](#)

- [CloudFormation テンプレートを作成する](#)
- [CloudFormation テンプレートと設定ファイルを送信する](#)
- [AWS Serverless Application Repository アプリケーションのアクセス許可を更新する](#)
- [AMI を共有する](#)
- [AMI とサーバーレスアプリケーションで CloudFormation 製品を送信する](#)

AWS Marketplace は、リストが作成される前に製品を確認して検証します。オファーがリスト化される前に解決しなければならない問題がある場合、メールメッセージでお知らせします。

サブスクリプションの受理の一環として、AMIs、サーバーレスアプリケーション、テンプレート AWS CloudFormation を各の AWS Marketplace 所有リポジトリにコピーします AWS リージョン。購入者が製品をサブスクライブすると、購入者にアクセス権が付与され、ソフトウェアの更新時にも通知されます。

サーバーレスアプリケーションを作成する

最初のステップは、サーバーレスアプリケーションの作成に使用される AWS Lambda 関数をパッケージ化することです。アプリケーションは、Lambda 関数、イベントソース、その他のリソースを組み合わせたもので、協調して動作することによりタスクを実行します。サーバーレスアプリケーションは、1つの Lambda 関数と同じくらい単純な場合もあれば、API、データベース、イベントソースマッピングなどの他のリソースと共に複数の関数を含む場合もあります。

を使用して AWS SAM、サーバーレスアプリケーションのモデルを定義します。プロパティ名とタイプの詳細については、「の AWS ラボ [AWS::Serverless::Application](#)」の「」を参照してください GitHub。以下は、単一の Lambda 関数と AWS Identity and Access Management (IAM) ロールを持つ AWS SAM テンプレートの例です。

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: An example of SAM template with Lambda function and IAM role

Resources:
  SampleFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: 'com.sampleproject.SampleHandler::handleRequest'
      Runtime: java8
      CodeUri: 's3://DOC-EXAMPLE-BUCKET/2EXAMPLE-1234-4b12-ac37-515EXAMPLEe5-
Lambda.zip'
```

```
Description: Sample Lambda function
Timeout: 120
MemorySize: 1024
Role:
  Fn::GetAtt: [SampleFunctionRole, Arn]

# Role to execute the Lambda function
SampleFunctionRole:
  Type: "AWS::IAM::Role"
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Effect: "Allow"
          Principal:
            Service:
              - "lambda.amazonaws.com"
          Action: "sts:AssumeRole"
    ManagedPolicyArns:
      - "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole"
  Policies:
    - PolicyName: SFNXDeployWorkflowDefinitionPolicy
      PolicyDocument:
        Statement:
          - Effect: "Allow"
            Action:
              - "s3:Get*"
            Resource: "*"
    RoleName: "SampleFunctionRole"
```

アプリケーションをリポジトリに公開する

アプリケーションを公開するには、最初にアプリケーションコードをアップロードします。アカウントが所有する Amazon S3 バケットにコードアーティファクト (Lambda 関数、スクリプト、構成ファイルなど) を保存します。アプリケーションをアップロードすると、最初はプライベートに設定されます。つまり、アプリケーションを AWS アカウント 作成した のみ使用できます。アップロードしたアーティファクトへのアクセス AWS Serverless Application Repository 許可を付与する IAM ポリシーを作成する必要があります。

サーバーレスアプリケーションをサーバーレスアプリケーションリポジトリに公開するには

1. <https://console.aws.amazon.com/s3/>でAmazon S3 コンソールを開きます。
2. アプリケーションのパッケージ化に使用した Amazon S3 バケットを選択します。

3. [アクセス許可] タブを選択します。
4. [バケットポリシー] を選択します。
5. 次のポリシーステートメントの例をコピーして貼り付けます。

 Note

ポリシーステートメントの例では、次の手順で `aws:SourceAccount` および `Resource` の値が更新されるまでエラーが生成されます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "serverlessrepo.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

- a. `Resource` プロパティ値の `DOC-EXAMPLE-BUCKET` をバケットのバケット名に置き換えます。
 - b. `Condition` 要素の `123456789012` を AWS アカウント ID に置き換えます。 `Condition` 要素により、は指定された からアプリケーションにアクセスするアクセス許可 AWS Serverless Application Repository のみを持つようになります AWS アカウント。
6. [保存] を選択します。
 7. で AWS Serverless Application Repository コンソールを開きます <https://console.aws.amazon.com/serverlessrepo>。

8. [My Applications (マイアプリケーション)] ページで、[Create application (アプリケーションの作成)] を選択します。
9. 必要に応じて、必須フィールドと任意のフィールドに入力します。必須フィールドは次のとおりです。
 - アプリケーション名
 - 筆者
 - 説明
 - ソースコード URL
 - SAM template
10. [Publish application (アプリケーションの公開)] を選択します。

アプリケーションの後続バージョンを公開するには

1. で AWS Serverless Application Repository コンソールを開きます <https://console.aws.amazon.com/serverlessrepo>。
2. [ナビゲーションペイン] で、[My Applications (マイアプリケーション)] からアプリケーションを選択します。
3. [新しいバージョンを発行] を選択します。

詳細については、[「AWS SAM CLI を使用したサーバーレスアプリケーションの公開」](#)を参照してください。

CloudFormation テンプレートを作成する

CloudFormation テンプレートを構築するには、テンプレートの前提条件を満たし、必要な入力パラメータとセキュリティパラメータを指定する必要があります。詳細については、「AWS CloudFormation ユーザーガイド」の[「テンプレートの構造分析」](#)を参照してください。

CloudFormation テンプレートでは、サーバーレスアプリケーションと AMI を参照できます。ネストされた CloudFormation テンプレートを使用し、ルートテンプレートとネストされたテンプレートの両方でサーバーレスアプリケーションを参照することもできます。サーバーレスアプリケーションを参照するには、AWS SAM テンプレートを使用します。からアプリケーションの AWS SAM テンプレートを自動的に生成できます AWS Serverless Application Repository。以下はテンプレートの例です。

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: An example root template for a SAR application

Resources:
  SampleSARApplication:
    Type: AWS::Serverless::Application
    Properties:
      Location:
        ApplicationId: arn:aws:serverlessrepo:us-east-1:1234567890:applications/
TestApplication
  SemanticVersion: 1.0.0
SampleEC2Instance:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: "ami-79fd7eee"
    KeyName: "testkey"
    BlockDeviceMappings:
      - DeviceName: "/dev/sdm"
        Ebs:
          VolumeType: "io1"
          Iops: "200"
          DeleteOnTermination: "false"
          VolumeSize: "20"
      - DeviceName: "/dev/sdk"
        NoDevice: {}
```

AWS SAM テンプレートには以下の要素が含まれています。

- ApplicationID - アプリケーションの Amazon リソースネーム (ARN)。この情報は、AWS Serverless Application Repositoryの [マイアプリケーション] セクションにあります。
- SemanticVersion - サーバーレスアプリケーションのバージョン。このフォームは、AWS Serverless Application Repositoryの [マイアプリケーション] セクションにあります。
- Parameter (オプション) - アプリケーションのパラメータ。

Note

ApplicationID および SemanticVersion では、[組み込み関数](#)はサポートされていません。これらの文字列はハードコーディングする必要があります。ApplicationID は、によってクローンされると更新されます AWS Marketplace。

テンプレート内の CloudFormation設定ファイルとスクリプトファイルを参照する場合は、次の形式を使用します。ネストされたテンプレート (AWS::CloudFormation::Stack) では、組み込み関数がない TemplateURLs でのみサポートされます。テンプレートの Parameters の内容に注意してください。

```
AWSTemplateFormatVersion: '2010-09-09'
Metadata:
  Name: Seller test product
Parameters:
  CFRefFilesBucket:
    Type: String
    Default: "seller-bucket"
  CFRefFilesBucketKeyPrefix:
    Type: String
    Default: "cftsolutionFolder/additionCFfiles"
Resources:
  TestEc2:
    Type: AWS::EC2::Instance
    Metadata:
      AWS::CloudFormation::Init:
        addCloudAccount:
          files:
            /etc/cfn/set-aia-settings.sh:
              source:
                Fn::Sub:
                  - https://${CFRefFilesBucket}.${S3Region}amazonaws.com/
                    ${CFRefFilesBucketKeyPrefix}/sampleScript.sh
                  - S3Region:
                      !If
                        - GovCloudCondition
                        - s3-us-gov-west-1
                        - s3
                  owner: root
                  mode: '000700'
                  authentication: S3AccessCreds
```

```

..
..
..
SampleNestedStack:
  Type: AWS::CloudFormation::Stack
  Properties:
    TemplateURL: 'https://sellerbucket.s3.amazonaws.com/sellerproductfolder/
nestedCft.template'
    Parameters:
      SampleParameter: 'test'
  Transform: AWS::Serverless-2016-10-31

```

CloudFormation テンプレートと設定ファイルを送信する

CloudFormation テンプレート、設定、スクリプトファイルを送信するには、これらのファイルが保存されている Amazon S3 バケットを読み取るアクセス許可を付与 AWS Marketplace します。これを行うには、バケットポリシーを更新して以下のアクセス許可を含めます。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "assets.marketplace.amazonaws.com"
      },
      "Action": ["s3:GetObject", "s3:ListBucket"],
      "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"]
    }
  ]
}

```

AWS Serverless Application Repository アプリケーションのアクセス許可を更新する

AWS Serverless Application Repository アプリケーションを送信するには AWS Marketplace、アプリケーションを読み取るアクセス AWS Marketplace 許可を付与する必要があります。これを行うには、サーバーレスアプリケーションに関連付けられているポリシーにアクセス許可を追加します。アプリケーションポリシーを更新するには 2 つの方法があります。

- [AWS Serverless Application Repository](#) に移動します。リストからサーバーレスアプリケーションを選択します。[共有] タブを選択し、[ステートメントを作成] を選択します。[ス

テートメントの設定] ページの [アカウント ID] フィールドに、次のサービスプリンシパル **assets.marketplace.amazonaws.com** を入力します。次に、[Save] (保存) を選択します。

- 次の AWS CLI コマンドを使用して、アプリケーションポリシーを更新します。

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=assets.marketplace.amazonaws.com,Actions=Deploy
```

AMI を共有する

に構築および送信されるすべての AMIs AWS Marketplace は、すべての製品ポリシーに従う必要があります。AWS Marketplace 管理ポータルではセルフサービス AMI スキャンを利用できます。この機能により、AMI のスキャンを開始できます。スキャン結果を迅速に (通常は 1 時間以内に) 受信でき、1 か所で明確なフィードバックを得ることができます。AMI が正常にスキャンされたら、製品ロードフォームをアップロードして、AWS Marketplace Seller Operations チームによる処理のために AMI を送信します。

AMI とサーバーレスアプリケーションで CloudFormation 製品を送信する

製品を送信する前に、次の点に注意してください。

- 各テンプレートのアーキテクチャ図を用意する必要があります。この図では、AWS CloudFormation テンプレートを通じてデプロイされた各 AWS サービスの製品アイコンを使用する必要があります。また、ダイアグラムにはサービスのメタデータが含まれている必要があります。公式の AWS アーキテクチャアイコンをダウンロードするには、「[AWS アーキテクチャアイコン](#)」を参照してください。
- 購入者に表示される各テンプレートのインフラストラクチャの料金見積もりは、[AWS 料金計算ツール](#)を使用して提示した見積もりに基づきます。見積もりには、一般的なデプロイ向けのデフォルト値と共にテンプレートの一部としてデプロイされるサービスのリストを含めます。
- 製品ロードフォームに入力します。AWS Marketplace 管理ポータルから製品ロードフォームを見つけることができます。単一 AMI 製品と複数 AMI 製品には、異なる製品ロードフォームが必要です。製品ロードフォームでは、テンプレートへの CloudFormationパブリック URL を指定します。CloudFormation テンプレートはパブリック URL の形式で送信する必要があります。
- AWS Marketplace 管理ポータル を使用してリストを送信します。[Assets (アセット)] から [File upload (ファイルのアップロード)] を選択し、ファイルをアタッチして、[Upload (アップロード)]

を選択します。テンプレートとメタデータを受け取ると、はリクエストの処理 AWS を開始します。

リストを送信すると、は製品のロードフォーム AWS Marketplace を確認して検証します。さらに、は AMIs とサーバーレスアプリケーションを AWS Marketplace リージョン化し、ユーザーに代わって AWS CloudFormation テンプレートのリージョンマッピングを更新します。問題が発生した場合は、AWS Marketplace Seller Operations チームが E メールで連絡します。

AMI 構築のベストプラクティス

このトピックでは、で使用する Amazon マシンイメージ (AMIs) の構築に役立つベストプラクティスとリファレンスをいくつか紹介します AWS Marketplace。に構築および送信される AMIs AWS Marketplace は、すべての AWS Marketplace 製品ポリシーに従う必要があります。

再販権の確保

が提供する Amazon Linux、RHEL、SUSE、および Windows AMIs を除き、無料ではない Linux AWS ディストリビューションの再販権を保護する責任はお客様にあります。

AMI の構築

AMI の構築に関する以下のガイドラインを使用してください。

- AMI がルートログインの無効化を含むすべての AWS Marketplace ポリシーを満たしていることを確認します。
- 米国東部 (バージニア北部) リージョンで AMI を作成します。
- AWS Marketplace などの信頼できる評判の高いソースから入手した、明確に定義されたライフサイクルを持つ、Amazon Elastic Block Store (Amazon EBS) にサポートされた、よく管理された既存の AMI から製品を作成します。
- ほとんどの up-to-date オペレーティングシステム、パッケージ、ソフトウェアを使用して AMIs を構築します。
- すべての AMI は、ハードウェア仮想マシン (HVM) 仮想化と 64 ビットアーキテクチャを使用するパブリック AMI から始める必要があります。
- AMI の構築、更新、再公開のための繰り返し可能なプロセスを開発します。
- すべてのバージョンと製品で一貫したオペレーティングシステム (OS) ユーザー名を使用します。ec2-user をお勧めします。

- 実行中のインスタンスを最終的な AMI からエンドユーザーエクスペリエンスに設定し、AWS Marketplace への送信前にすべてのインストール方法、機能、およびパフォーマンスをテストします。
- 以下のようにポート設定を確認してください。
 - Linux ベースの AMI - 有効な SSH ポートが開いていることを確認します。デフォルトの SSH ポートは 22 です。
 - Windows ベースの AMI - RDP ポートが開いていることを確認します。デフォルトの RDP ポートは 3389 です。また、WinRM ポート (デフォルトは 5985) は 10.0.0.0/16 と 10.2.0.0/16 に開いている必要があります。

AMI の作成の詳細については、以下の関連リソースを参照してください。

Amazon EC2 ユーザーガイド」の「[独自の AMI の作成](#)」

Amazon EC2 [ユーザーガイド](#)」の「[カスタム Windows AMI の作成](#)」

[EBS-backed インスタンスから Amazon マシンイメージ \(AMI\) を作成する方法。](#)

[Amazon Linux AMI](#)

[Amazon EC2 インスタンスタイプ](#)および[インスタンスタイプ](#)

の AMI の準備と保護 AWS Marketplace

セキュアな AMI を作成するために、次のガイドラインをお勧めします。

- Amazon EC2 ユーザーガイドの[共有 Linux AMIsのガイドライン](#)を使用する Amazon EC2
- 攻撃の危険性を減らすため、最低限のインストールとしてデプロイするように AMI を設計します。不要なサービスやプログラムを無効にするか、削除します。
- 可能な限り、ネットワークトラフィックには end-to-end 暗号化を使用してください。例えば、Secure Sockets Layer (SSL) を使用して、購入者との間の HTTP セッションを保護します。サービスが有効な 証明書と up-to-date 証明書のみを使用していることを確認します。
- AMI 製品に新しいバージョンを追加するときは、インスタンスへのインバウンドトラフィックアクセスを制御するセキュリティグループを設定します。セキュリティグループが、サービスに必要な機能を提供するために必要な最小ポートセットにのみアクセスできるように設定されていることを確認します。最小セットのポートと必要な送信元 IP アドレス範囲に対してのみ管理アクセスを許可します。AMI 製品に新しいバージョンを追加する方法の詳細については、「[新しいバージョンを追加する](#)」を参照してください。

- AWS コンピューティング環境に対して定期的にペネトレーションテストを実施することを検討してください。または、ユーザーに代わってそのようなテストを実行するためにサードパーティーを雇用することを検討してください。ペネトレーションテストリクエストフォームなど詳細は、「[AWS ペネトレーションテスト](#)」を参照してください。
- ウェブアプリケーションのトップ 10 の脆弱性に気づき、必要に応じてアプリケーションを構築してください。詳細については、「[Open Web Application Security Project \(OWASP\) - Top 10 Web Application Security Risks](#)」を参照してください。新しいインターネットの脆弱性が発見された場合は、AMI に同梱されているウェブアプリケーションを速やかに更新してください。この情報を含むリソースの例は、[SecurityFocus](#)と [NIST National Vulnerability Database](#) です。

セキュリティの詳細については、以下の関連リソースを参照してください。

- [AWS クラウドセキュリティ](#)
- [インターネットセキュリティセンター \(CIS\): セキュリティベンチマーク](#)
- [オープンなウェブアプリケーションセキュリティプロジェクト \(OWASP\): セキュアなコーディングプラクティスクイックリファレンスガイド](#)
- [OWASP トップ 10 ウェブアプリケーションのセキュリティリスク](#)
- [SANS \(SysAdmin、監査、ネットワーク、セキュリティ\) 共通脆弱性列挙 \(CWE\) の最も危険なソフトウェアエラーの上位 25 件](#)
- [セキュリティフォーカス](#)
- [NIST 全国脆弱性データベース](#)

公開要件を確認するための AMI のスキャン

AMI を新製品または新バージョンとして提出する前に確認するには、セルフサービススキャンを使用できます。セルフサービススキャナーは、パッチが適用されていない共通脆弱性識別子 (CVE) をチェックし、セキュリティのベストプラクティスに従っていることを確認します。詳細については、「[the section called “の AMI の準備と保護 AWS Marketplace”](#)」を参照してください。

から AWS Marketplace 管理ポータル、アセットメニューから Amazon マシンイメージを選択します。[Add AMI] を選択してスキャンプロセスを開始します。AMI のスキャンステータスは、このページに戻ると確認できます。

Note

AMI AWS Marketplace へのアクセスを許可する方法については、「」を参照してください。[AMI AWS Marketplace へのアクセスを許可する](#)。

AWS Marketplace AMI でソフトウェアが実行されていることを確認する

AMI 製品から作成された Amazon EC2 インスタンスで実行されていることを、ランタイムにソフトウェアに確認させたい場合があります。

Amazon EC2 インスタンスが AMI 製品から作成されていることを確認するには、Amazon EC2 に組み込まれているインスタンスメタデータサービスを使用します。以下のステップでは、この検証を順を追って説明します。メタデータサービスの使用の詳細については、「Amazon Elastic Compute Cloud ユーザーガイド」の「[インスタンスメタデータとユーザーデータ](#)」を参照してください。

1. インスタンスアイデンティティドキュメントの取得

実行中の各インスタンスには、インスタンスからアクセス可能な、インスタンス自体に関するデータを提供するアイデンティティドキュメントが用意されています。次の例は、インスタンスから curl を使用してインスタンスのアイデンティティドキュメントを取得する方法を示しています。

```
curl http://169.254.169.254/latest/dynamic/instance-identity/document
{
  "accountId" : "0123456789",
  "architecture" : "x86_64",
  "availabilityZone" : "us-east-1e",
  "billingProducts" : null,
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : [ "0vg00000000000000000000000" ],
  "imageId" : "ami-0123456789abcdef1",
  "instanceId" : "i-0123456789abcdef0",
  "instanceType" : "t2.medium",
  "kernelId" : null,
  "pendingTime" : "2020-02-25T20:23:14Z",
  "privateIp" : "10.0.0.2",
  "ramdiskId" : null,
  "region" : "us-east-1",
  "version" : "2017-09-30"
```

```
}
```

2. インスタンスアイデンティティドキュメントの検証

署名を使用して、インスタンスアイデンティティが正しいことを確認できます。このプロセスの詳細については、「Amazon Elastic Compute Cloud ユーザーガイド」の「[インスタンスアイデンティティドキュメント](#)」を参照してください。

3. 製品コードの検証

AMI 製品を最初に公開用に提出すると、製品には AWS Marketplace によって [製品コード](#) が割り当てられます。製品コードは、インスタンスアイデンティティドキュメントの `marketplaceProductCodes` フィールドを確認すると検証できます。または、メタデータサービスから直接取得することもできます。

```
curl http://169.254.169.254/latest/meta-data/product-codes
0vg00000000000000000000000000000000
```

製品コードが AMI 製品のコードと一致する場合、インスタンスは製品から作成されたものです。

`instanceId` やインスタンス `privateIp` など、インスタンスアイデンティティドキュメントから他の情報を確認したい場合もあります。

AMI 製品の料金

AWS Marketplace には、Amazon マシンイメージ (AMI) 製品の複数の料金モデルがあります。販売者のプライベートオファーでは、複数年契約とカスタム期間契約のオプションがあります。

複数年契約とカスタム期間契約の詳細については、[プライベートオファーの準備および分割プラン](#)を参照してください。次の表に、料金モデルに関する一般的な情報を示します。

Note

[販売者登録プロセス](#) に記載されているように、W-9 フォーム (米国居住者) または W-8 フォーム (米国非居住者) を提出する必要があります。

AMI 料金モデル

次の表に、AMI ベースの製品の料金モデルに関する一般的な情報を示します。

料金モデル	説明
無料	<p>お客様は Amazon Elastic Compute Cloud (Amazon EC2) でサポートされている量のインスタンスを実行することができます。ソフトウェアの追加料金はかかりません。</p> <div data-bbox="646 422 1507 642" style="border: 1px solid #add8e6; border-radius: 15px; padding: 10px; margin-top: 10px;"> <p> Note</p> <p>無料トライアルと年間料金は、月額料金と組み合わせることはできません。</p> </div>
自分のライセンス使用 (BYOL)	<p>AWS Marketplace は、本ソフトウェアの使用に対してお客様に課金しませんが、製品をアクティブ化するにはライセンスキーを指定する必要があります。このキーは の外部で購入されます AWS Marketplace。資格とライセンスの実施、およびすべての料金と課金は、販売者によって処理されます。</p>
時間単位料金または時間あたりの年間料金	<p>[時間単位] - ソフトウェアの料金は、時間単位で請求されます。各インスタンスタイプの料金は個々に異なり (ただし、必須ではない)、その使用量は最も近い時間に切り上げられます。</p> <p>[時間単位 (無料トライアルあり)] - お客様は、ソフトウェアのインスタンスを 1 つだけ実行できるという制限があり、料金は発生しません。期間は 5~30 日の間で定義します。無料トライアルは、実行中の最も高価なインスタンスタイプに適用され、1 インスタンスを超える同時使用はすべて、時間料金で請求されます。注: これは、Amazon EC2 の AWS 無料利用枠とは別のモデルです。これにより、お客様には毎月 750 時間の無料利用が提供されます。</p> <p>[時間単価と月額料金] - 時間単価と月額料金の両方が別々に適用されます。月額料金は使用量にかかわらず毎月適用され、時間料金は時間単位の使用量のみに基づいて適用されます。</p>

料金モデル	説明
	<p>[年間で毎時]: お客様は、1つのインスタンスタイプの1つの Amazon EC2 インスタンスに対して1年間の使用量を購入することができます。販売者は、インスタンスタイプごとの料金を設定し、時間単価よりも価格を抑えることができます。購入された年間サブスクリプションを超えるお客様の使用量は、そのインスタンスタイプの販売者によって設定された時間単価で請求されます。</p> <p>[時間単位 (複数年間およびカスタム期間)] - このタイプのオファーは、販売者のプライベートオファーを通じてのみ利用できます。販売者のプライベートオファーを使用して、カスタム契約期間 (最大3年) を指定します。前払い料金を指定することも、柔軟な支払いスケジュールを含めることもできます。各インスタンスタイプの料金を設定します。オファーに柔軟な支払いスケジュールがある場合は、オファーに含まれる各インスタンスタイプの請求日、支払額、インスタンス数も設定します。柔軟な支払いスケジュールを持つアクティブな販売者のプライベートオファーの場合、お客様が指定された数のインスタンスを起動した後、起動された追加のインスタンスには、販売者のプライベートオファーで指定された時間料金で課金されます。複数年契約とカスタム期間契約の詳細については、プライベートオファーの準備および the section called “分割プラン” を参照してください。</p> <p>[時間単位 (無料トライアル付き)] - 年間オプション付きの時間単位モデルと同じです。ただし、販売者が設定した日数で、お客様がインスタンスタイプのインスタンスを無料で1つ実行できる無料トライアルを含む点を除きます。年間サブスクリプションはいつでも購入することができます。また、無料トライアルサブスクリプションと組み合わせることができます。</p> <p>[年間料金 (時間単位)] 時間単位の料金モデルと同じ。お客様は、1つのインスタンスタイプの1つの Amazon EC2 インスタンスに対して1年間の使用量を購入することができます。販売者は、インスタンスタイプごとの料金を設定し、時</p>

料金モデル	説明
	<p>間単価よりも料金を抑えることができますが、減額は必須ではありません。購入された年間サブスクリプションを超えるお客様の使用量は、そのインスタンスタイプの販売者によって設定された時間単価で請求されます。</p> <p>[複数年間およびカスタム期間 (時間単位)] - これは、プライベートオファーの準備 を通じてのみ利用できます。販売者のプライベートオファーを使用して、最長 3 年のカスタム期間契約を指定できます。前払いを要求することも、顧客に柔軟な支払いスケジュールを提供することもできます。契約期間中の各インスタンスタイプの料金と、起動される追加のインスタンスの時間料金を設定します。柔軟な支払いスケジュールを提供する場合は、オファーに含まれる各インスタンスタイプの請求日、支払額、インスタンス数も設定します。柔軟な支払いスケジュールを持つアクティブなプライベートオファーの場合、指定された数のインスタンスが起動された後、お客様が起動する追加のインスタンスには、プライベートオファーで指定された時間料金で課金されます。複数年契約とカスタム期間契約の詳細については、プライベートオファーの準備 および the section called “分割プラン” を参照してください。</p> <div data-bbox="646 1226 1507 1440" style="border: 1px solid #0070C0; border-radius: 10px; padding: 10px;"><p> Note</p><p>無料トライアルと年間料金は、月額料金と組み合わせることはできません。</p></div>

料金モデル	説明
月額	<p>[月額] - ソフトウェア料金は、顧客が実行するインスタスの数に関係なく、固定月額料金で請求されます。月額料金はサインアップおよびキャンセル時に按分されます。例：その月の1日分をサブスクライブした顧客には、その月の1/30の料金が請求されます。</p> <p>[時間単価を伴う月額料金] - 時間単価と月額料金の両方が別々に適用されます。月額料金は使用量にかかわらず毎月適用され、時間料金は時間単位の使用量のみに基づいて適用されます。</p> <div style="border: 1px solid #add8e6; border-radius: 15px; padding: 10px; margin-top: 10px;"> <p> Note</p> <p>無料トライアルと年間料金は、月額料金と組み合わせることはできません。</p> </div>
有料使用料金	<p>ソフトウェアの料金は、販売者がユーザー、データ、帯域幅、またはホストの4つの使用カテゴリのいずれかと併せて提供する価値に対して直接請求されます。製品に対して最大24のディメンションを定義できます。料金はすべて、お客様によって時間料金で請求されます。</p> <p>すべての使用量は毎月計算され、既存のAWS Marketplaceソフトウェアと同じメカニズムを使用して毎月請求されます。利用料金は、AWS Marketplace Metering Service と呼ばれます</p>
契約料金モデル	<p>契約料金でのAMI – 購入者AWS CloudFormation が前払い料金を支払うスタック付きのシングルAMI製品またはシングルAMI。</p>

AWS 料金とソフトウェア料金

Amazon マシンイメージ (AMI) ベースの製品料金は2つのカテゴリに分類されます。

- インフラストラクチャ料金の詳細 — すべての AMI ベースの製品には、使用するサービスと AWS インフラストラクチャに応じて、関連するインフラストラクチャ料金が発生します。これらのレートと料金は、によって定義および管理され AWS、によって異なる場合があります AWS リージョン。詳細については、「[Amazon EC2 の料金](#)」を参照してください。
- ソフトウェアの料金の詳細 - 有料製品の場合、販売者はソフトウェアの使用料金を定義します。

これら 2 つの製品料金カテゴリは、購入者が製品使用の潜在的なコストを理解するのに役立つように、AWS Marketplace 詳細ページに個別に表示されます。

AMI 時間単位製品の無料トライアル

AMI 時間単位の製品は、オプションの無料トライアルプログラムの対象となります。無料トライアルでは、お客様は製品のソフトウェア料金を支払うことなく、製品をサブスクライブし、単一のインスタンスを最大 31 日間使用できます。適用される AWS インフラストラクチャ料金が引き続き適用されます。無料トライアルは、有効期限が切れると自動的に有料サブスクリプションに切り替わります。お客様には、提供された無料単位を超える追加使用分が課金されます。時間単位の製品無料トライアルを提供するには、トライアル期間を定義し、[AWS Marketplace Seller Operations](#) チームに通知してください。試用期間は 5~31 日です。

無料トライアル製品をサブスクライブしている場合、お客様は、無料トライアルの期間、計算された有効期限、およびアンサブスクライブに関する詳細を含むようこそメールメッセージを受け取ります。有効期限が切れる 3 日前に通知メールが送信されます。

で無料トライアル製品を提供する場合 AWS Marketplace、返金ポリシー に記載されている特定の返金ポリシーに同意したものと見なされます。

AMI 製品のカスタム計量価格

Metering Service の概念

AWS Marketplace Metering Service を使用すると、ソフトウェア販売者は、計測レコードをエンドポイントに送信して使用量をキャプチャするようにソフトウェアを変更できます。販売者は、1 つの使用カテゴリと、そのカテゴリの最大 24 個のディメンションを選択できます。それらのディメンションは 1 時間に 1 回計測および集計され、販売者によって定義された料金プランに従って課金されます。販売者は、使用するディメンションを決定する必要があります。AMI が発行された後は、ディメンションを変更できません。重要なサービスの概念は、次のとおりです。

- [使用カテゴリ] - Metering Service を使用して価格が設定されるソフトウェア製品はすべて、1つの使用カテゴリに従って分類され、それによって顧客への適切な請求方法が決まります。使用カテゴリには次のものがありますが、これらに限定されるものではありません。
- [ユーザー] - 単一の ID に関連付けられた、定義済みのアクセス許可のセット。このカテゴリは、顧客のユーザーが直接ソフトウェアに接続するソフトウェア (たとえば、顧客関係管理またはビジネスインテリジェンス報告) に適切です。
- ホスト - サーバー、ノード、インスタンス、エンドポイント、またはコンピューティングシステムの他の部分。このカテゴリは、多くの顧客所有インスタンス (たとえば、パフォーマンスまたはセキュリティモニタリング) を監視またはスキャンするソフトウェアに適切です。
- データ - MB、GB、または TB で測定される、ストレージまたは情報。このカテゴリは、保存されたデータを管理する、またはバッチでデータを処理するソフトウェアに適切です。
- [帯域幅] - Mbps または Gbps 単位で測定されます。このカテゴリは、お客様がプロビジョニングする帯域幅を指定できるソフトウェアに適しています。
- [単位] - 測定単位。次に説明する例を参照してください。
- 使用量の単位 - 選択された使用カテゴリに対応するソフトウェア製品の特定の使用量の単位。この使用量の単位はソフトウェアの課金の単位について説明するものです。その例を以下に示します。
 - NodesHrs (ホストカテゴリに対応)
 - UserHrs (ユーザーカテゴリに対応)
 - [GBStored] (データカテゴリに対応)
- 消費量 - Metering Service を使用して価格設定されているソフトウェアは、次の3つのいずれかの方法で消費量に対して課金されます。
 - プロビジョニング - ソフトウェアで、リソースの特定の量を使用するようにお客様が設定できます (例えば、ユーザー数や固定量の帯域幅)。お客様は、プロビジョニングした量に対して1時間ごとに支払います。
 - 同時実行 - ソフトウェアで、任意の数のホストまたはユーザーがそのソフトウェアに接続できるようにします。お客様は、ソフトウェアにアクセスしたユーザー数またはホスト数に基づいて1時間ごとに支払います。
 - 累積 - ソフトウェアで、任意の量のデータを処理または保管に使用できるようにします。お客様は、集計された量に対して1時間ごとに支払います。
- 料金表 - Metering Service を使用して価格設定したソフトウェア製品では、単一の料金を指定するか、または独自の料金がある最大24個のディメンションを定義する必要があります。価格設定オプションの詳細には以下が含まれます。

- 単一ディメンション - 最もシンプルな料金オプションです。顧客は、サイズまたはボリュームに関わらず、1 時間あたりリソース単位あたりの単一の料金を支払います (たとえば、1 時間あたりユーザーあたり 0.014 USD、または 1 時間あたりホストあたり 0.070 USD)。
- 複数ディメンション - この料金オプションは、選択した使用カテゴリが複数の軸で変化する場合に適しています。たとえば、ホストモニタリングでは、ホストのサイズに応じて異なる料金を設定できます。または、ユーザーベースの料金では、ユーザーのタイプ (管理者、パワーユーザー、読み取り専用ユーザー、など) に応じて異なる料金を設定できます。
- 計測 - 1 時間に 1 回、すべての使用量が計測イベントとして記録されます。適切なディメンションおよび使用量を AWS Marketplace Metering Service に送信するようにソフトウェアが設定されている必要があります。
- 配分 - オプションで、追跡する施設ごとに分けて使用量を配分できます。これらの配分は、購入者にはタグとして表示されます。タグを使うと、購入者はコストをタグ別の使用量に分けて表示できます。例えば、ユーザーごとに課金し、ユーザーに「Department」プロパティがある場合、「Department」のキーを持つタグを使用して使用量の割り当てを作成し、値ごとに 1 つの割り当てを行うことができます。この方法では、レポートする価格、ディメンション、または合計使用量は変わりません。ただし、お客様は商品に適したカテゴリ別にコストを確認できます。

ソフトウェアの料金

AWS Marketplace Metering Service でソフトウェアの料金を設定するには、まず使用カテゴリとその使用方法を指定する必要があります。このサービスは 6 つの異なる個別料金シナリオをサポートしています。これらのシナリオのうちいずれかを製品向けに選択する必要があります。

- プロビジョニングされたユーザー (1 時間あたり)
- 同時ユーザー (1 時間あたり)
- プロビジョニングされたホスト (1 時間あたり)
- 同時ホスト (1 時間あたり)
- プロビジョニングされた帯域幅 (1 時間あたり)
- 蓄積されたデータ (1 時間あたり)

次に、選択した使用カテゴリの料金を指定する方法を判断する必要があります。

- 単一価格
- 複数のディメンション (最大 24 個)

「[への製品の追加 AWS Marketplace](#)」では、ディメンションと料金についてわかりやすい説明を提供する方法について説明します。

例: 非線形料金設定によりプロビジョニングされた帯域幅

ネットワークアプライアンスソフトウェアを提供するとします。プロビジョニングされた帯域幅で請求します。使用カテゴリで、[帯域幅]を選択します。帯域幅による請求に加えて、購入者のスケールアップに応じて別料金を請求するとします。帯域幅カテゴリ内で複数のディメンションを定義することができます。25 Mbps、100 Mbps、および 1 Gbps 向けに個別の料金を定義できます。

例: 複数のディメンションにおける同時ホスト

他の Amazon EC2 インスタンスを監視するソフトウェアを提供するとします。監視中のホスト数で請求するとします。使用カテゴリで、[ホスト]を選択します。ホストごとの請求に加えて、規模の大きなホストを監視する場合は特別料金を請求するとします。ホストカテゴリ内で複数のディメンションを使用することができます。マイクロ、スモール、ミディアム、ラージ、xラージ、2XL、4XL、8XL インスタンス向けに個別の料金を定義することができます。お客様のソフトウェアを使用して、特定の各ホストを定義されたディメンションのいずれかにマッピングします。また、該当する場合は、使用カテゴリのディメンションごとに個別にレコードを計測します。

への製品の追加 AWS Marketplace

Metering Service を利用するには、[が一覧表示 AWS Marketplace](#) する新しい製品を作成する必要があります。製品がすでにある場合は AWS Marketplace、現在の製品に加えて新しい AWS Marketplace Metering Service 製品が利用可能になるか、新しいユーザーが利用できる唯一のバージョンとして現在の製品を置き換えるかを決定する必要があります。置き換えを選択すると、既存の製品は から削除 AWS Marketplace され、新しい購入者は使用できなくなります。既存の顧客は、引き続き古い製品とインスタンスにアクセスできますが、必要に応じて新しい製品に移行することができます。新製品では、「[Metering Service を使用するためのソフトウェアの変更](#)」に説明しているように、AWS Marketplace Metering Service の使用料金を測定する必要があります。

AMI を取得したら、標準プロセスに従い、セルフサービスツールを使用して AMI を共有およびスキャンします。さらに、管理ポータルで使用可能なテンプレートを使用して製品ロードフォームに記入後、アップロードして処理を開始します。

AWS Marketplace Metering Service の製品ロードフォームのフィールドに入力するには、次の定義を使用します。製品ロードフォームでは、これらのフィールドは、時間単位の製品および月額製品の区別するために [Flexible Consumption Pricing (FCP)] というラベルが付けられています。

- タイトル – にすでに製品があり AWS Marketplace、 で同じ製品を追加する場合は AWS Marketplace Metering Service、それらを区別するために括弧内に FCP カテゴリとディメンションを含めます (例: 「製品タイトル (データ) 」)。
- [料金モデル] - ドロップダウンリストより、[使用量] を選択します。
- [FCP カテゴリ] - [使用量] 料金コンポーネントを含む出品製品の請求対象となるカテゴリ。ドロップダウンリストより、[ユーザー]、[ホスト]、[データ]、または [帯域幅] を選択します。
- [FCP 単位] - [使用量] 料金コンポーネントを含む出品製品の請求対象となる測定の単位。選択した FCP カテゴリに基づき、ドロップダウンリストにオプションが表示されます。カテゴリごとに有効な単位を次のテーブルに示します。

カテゴリ	有効な単位
[ユーザー]	UserHrs
[ホスト]	HostHrs
[データ]	MB、GB、TB
[帯域幅]	Mbps、Gbps

- [FCP ディメンション名] - MeterUsage オペレーションを呼び出して計測レコードを送信する際に使用される名前。これは請求レポートに記載されます。ただし、外部向けではないため、わかりやすい名前である必要はありません。名前は 15 文字以内でなければならず、英数字とアンダースコアのみ含めることができます。名前を設定して製品を公開した後は、変更できません。この名前を変更するには、新しい AMI が必要です。
- [FCP ディメンションの説明] - 製品のディメンションを記述した顧客向けステートメント。説明は 70 文字以内でなければならず、ユーザーフレンドリである必要があります。説明の例としては、「1 時間あたりの管理者数」や「プロビジョニングされた Mbps あたりの帯域幅」などがあります。製品が発行された後に、この説明を変更することはできません。
- [FCP レート] - この製品のユニットあたりのソフトウェア料金。このフィールドは、小数点以下 3 桁をサポートしています。

注記：

- 時間単位および年間単位の料金フィールドを入力する必要はありません。

- 無料トライアルと年間料金は互換がありません。
- 複数の AMIs とクラスターおよび AWS リソース機能を使用する製品は、AWS Marketplace Metering Service を使用できません。
- 料金、インスタンスタイプ、または AWS リージョン 変更は、他の AWS Marketplace 製品と同じプロセスに従います。
- AWS Marketplace Metering Service を使用した製品は、時間単位、月別、または Bring-Your-Own-License (BYOL) などの他の料金モデルに変換することはできません。
- AWS Marketplace では、使用方法またはドキュメントに IAM ポリシー情報を追加することをお勧めします。
- 合計で最大 24 個の FCP デイメンションを含めることができます。作成して公開すると、既存のデイメンションを変更することはできませんが、新しいデイメンションを追加できます (最大 24 個)。

ご質問がある場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

Metering Service を使用するためのソフトウェアの変更

顧客の使用状況の記録、Metering Service への時間単位の使用状況レポートの送信、新しい障害モードの処理を行うには、ソフトウェアを変更する必要があります。このソフトウェアは料金とは関係なく使用できますが、ソフトウェアの使用カテゴリ、使用方法、デイメンションについて把握する必要があります。

使用量の測定

ソフトウェアを使用して、選択された使用カテゴリと顧客が使用したデイメンションを判断する必要があります。この値は、Metering Service に 1 時間ごとに送信されます。いずれの場合も、Metering Service に時間単位で送信する目的で、リソースの消費量の測定、記録、読み取りの機能がソフトウェアに搭載されていることを前提としています。

プロビジョニングされた消費の場合、この値は通常、サンプル値としてソフトウェア構成から読み取られますが、1 時間ごとに記録される最大設定値でもあります。同時消費の場合、この値は、定期的なサンプルか、1 時間ごとに記録される最大値のいずれかになります。累積消費の場合、この値は 1 時間ごとに累積されます。

複数のデイメンションの料金については、複数の値を測定し、デイメンションごとに 1 つずつ Metering Service に送信する必要があります。そのためには、AMI を指定する際に、既知のデイメン

シオンセットを使用してソフトウェアをプログラミングまたは設定する必要があります。製品が作成された後は、一連のディメンションを変更することはできません。

次のテーブルには、1時間あたりの消費量を測定するための推奨方法を料金シナリオごとに示しています。

シナリオ	測定方法
プロビジョンドユーザー	<p>現在のプロビジョンドユーザー数 (サンプリング)。</p> <p>-または-</p> <p>プロビジョンドユーザーの最大数 (その時間を参照)</p>
同時ユーザー	<p>現在の同時ユーザー数 (サンプリング)。</p> <p>-または-</p> <p>同時ユーザーの最大数 (その時間を参照)</p> <p>-または-</p> <p>個々のユーザーの合計数 (その時間を参照)</p>
プロビジョンドホスト	<p>現在のプロビジョンドホスト数 (サンプリング)。</p> <p>-または-</p> <p>プロビジョンドホストの最大数 (その時間を参照)</p>
同時ホスト	<p>現在の同時ホスト数 (サンプリング)。</p> <p>-または-</p> <p>同時ホストの最大数 (その時間を参照)</p> <p>-または-</p>

シナリオ	測定方法
プロビジョンド帯域幅	個々のホストの合計数 (その時間を参照) 現在のプロビジョニングされた帯域幅設定 (サンプリングされた)。 -または- その時間に発生するプロビジョニングされた最大帯域幅 (その時間を参照)。
蓄積されたデータ	現在の GB に格納されたデータ (サンプリングされた)。 -または- 格納されたデータの最大 GB (その時間を参照) -または- その時間に追加または処理されたデータの合計 (GB)。 -または- その時間に処理されたデータの合計 (GB)。

ベンダー計測タグ付け (オプション)

ベンダー計測タグ付けは、独立系ソフトウェアベンダー (ISV) がソフトウェアの使用状況についてより詳細な情報を購入者に提供し、コスト配分を行うのに役立ちます。

購入者のソフトウェア使用状況をタグ付けするには、コストの配分方法を決定する必要があります。まず、コスト配分に何を期待するかを購入者にたずねます。次に、購入者のアカウントで追跡するプロパティ全体で使用量を分割できます。プロパティの例には、Account ID、Business Unit、Cost Centers、および商品に関連する他のメタデータが含まれます。これらのプロパティはタグとして購入者に公開されます。タグを使用すると、購入者は AWS 請求コンソール (<https://console.aws.amazon.com/billing/>) のタグ値別にコストを使用量に分割して表示できます。ベンダー

計測タグ付けでは、報告する価格、ディメンション、合計使用量は変わりません。お客様は商品に適したカテゴリ別にコストを確認できます。

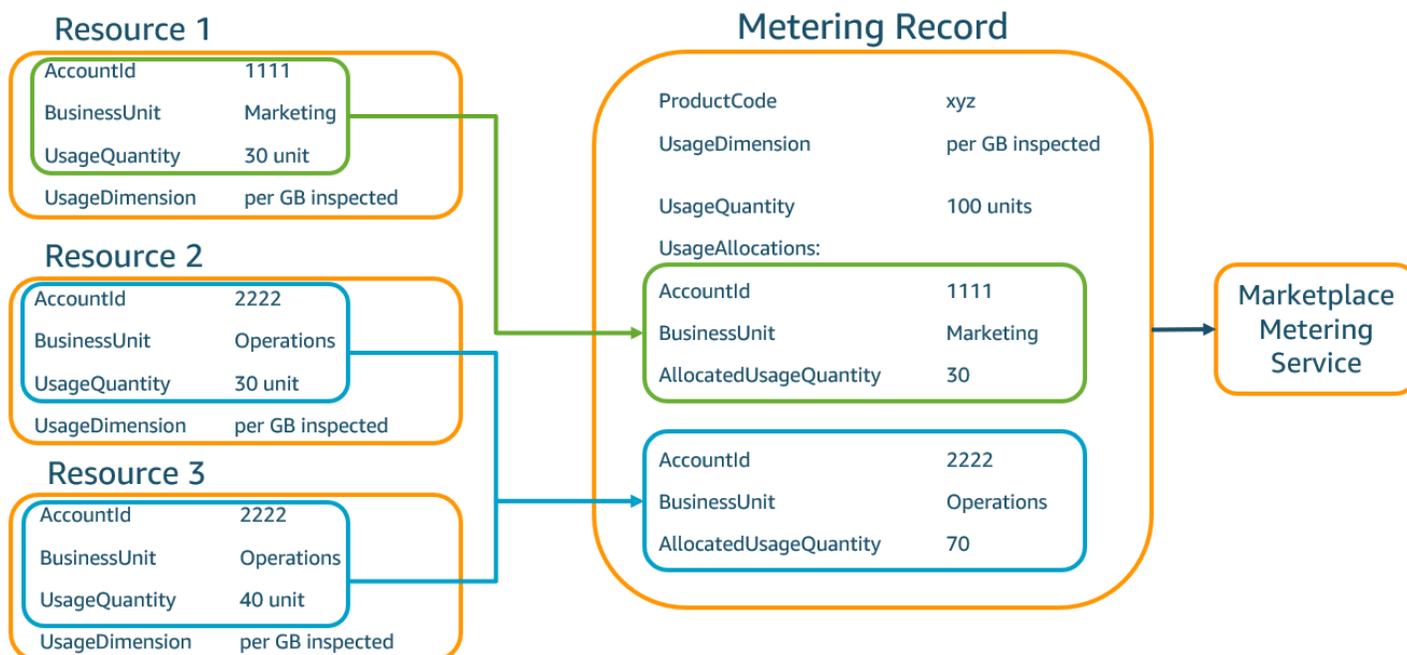
一般的なユースケースでは、購入者が1つのAWSアカウントを使って商品を購入します。また、購入者には同じ製品サブスクリプションに関連するユーザーが多数います。Account IDのキーを持つタグを使用して使用量の割り当てを作成し、各ユーザーに使用量を割り当てることができます。この場合、購入者はBilling and Cost ManagementコンソールでAccount IDタグを有効にして、個々のユーザーの使用状況を分析できます。

販売者のエクスペリエンス

販売者は、すべてのリソースの使用量を集計する代わりに、同じタグセットを使用してリソースの計測レコードを集計できます。例えば、販売者はUsageAllocationsの異なるバケットを含む計測レコードを作成できます。各バケットは、AccountIdやBusinessUnitなどの、タグのセットのUsageQuantityを表します。

以下の図では、[リソース 1]には固有のAccountIdとBusinessUnitタグセットがあり、[計測レコード]に1つのエントリとして表示されます。

[リソース 2]と[リソース 3]にはどちらも同じAccountIdタグ、2222、および同じBusinessUnitタグOperationsがあります。その結果、これらは[計測レコード]の1つのUsageAllocationsエントリにまとめられます。



また、販売者は、タグのないリソースを割り当てられた使用量の単一のUsageAllocationにまとめて、UsageAllocationsのエントリの1つとして送信することもできます。

制限には以下が含まれます。

- タグの数 - 5
- UsageAllocations のサイズ (カーディナリティ) - 2,500

検証には次の項目が含まれます。

- タグのキーと値に使用できる文字 - a-zA-Z0-9+ -=._:\/@
- UsageAllocation リスト全体の最大タグ数 - 5
- 2 つの UsageAllocations タグに同じタグ (つまり、同じタグキーと値の組み合わせ) を持つことはできません。その場合は、同じ UsageAllocation を使用する必要があります。
- UsageAllocation の AllocatedUsageQuantity の合計は、総使用量である UsageQuantity と等しくなる必要があります。

購入者エクスペリエンス

次の表は、購入者が AccountId および BusinessUnit ベンダータグを有効化した後の購入者エクスペリエンスの例を示しています。

この例では、購入者は [コスト使用状況レポート] で割り当てられた使用量を確認できます。ベンダー計測タグにはプレフィックス “aws:marketplace:isv” が使用されます。購入者は、請求情報とコスト管理の、[コスト配分タグ]、[AWS生成コスト配分タグ] でそれらを有効化できます。

[コスト使用状況レポート] の最初と最後の行は、出品者が Metering Service に送信する内容に関係します ([販売者のエクスペリエンス](#) 例を参照)。

コスト使用状況レポート (簡略版)

ProductCode	購入者	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	70	2222	オペレーショ ン

ProductCode	購入者	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	30	3333	財務
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	20	4444	IT
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	20	5555	マーケティング
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	30	1111	マーケティング

コード例については、「[使用量配分タグ付き MeterUsage \(オプション\)](#)」を参照してください。

AMI 製品の契約料金

Amazon マシンイメージ (AMI) ベースの製品の契約料金とは、購入者が単一の AMI 製品または AWS CloudFormation スタック付きの単一の AMI に対して前払い料金を支払うことを意味します。契約料金の AMI ベースの製品の場合、は、お客様とお客様の間の契約に基づいて、前払いまたは定義した支払いスケジュールでお客様に AWS Marketplace 請求します。その時点で、それらのリソースの使用権限が付与されます。

料金を設定するには、顧客に提供する 1 つ以上の契約期間を選択します。契約期間ごとに異なる価格を入力できます。契約期間は 1 か月、12 か月、24 か月、36 か月です。プライベートオファーでは、カスタム期間を月単位 (最大 60 か月) で指定できます。

製品の価格を最もよく表すカテゴリを選択します。料金カテゴリは、AWS Marketplace ウェブサイトの顧客に表示されます。[帯域幅] (GB/s、MB/s)、[データ] (GB、MB、TB)、[ホスト]、[リクエスト]

ト]、[階層]、[ユーザー]のいずれかを選択できます。事前定義カテゴリのどれもニーズに適合しない場合は、汎用的な [単位] カテゴリを選択できます。

このオファーには最大 24 のディメンションを追加できます。各ディメンションには次のデータが必要です。

- 契約カテゴリ - 消費ベースの料金設定のない契約製品の場合は、契約内のディメンションのカテゴリに最も近いカテゴリを選択するか、契約内のディメンションの単位に似た値がない場合は [単位] を選択できます。
- 契約単位 - 選択したカテゴリに基づいて、ディメンションに最も近い単位の値を 1 つ選択します。
- 契約ディメンションによる複数購入が可能 - このフィールドは、オファーが階層型の料金設定オファーか非階層型のオファーかを示すために使用されます。

階層型オファー - 購入者がオファー内の利用可能なディメンションのうちの 1 つだけをサブスクライブできるようにします。階層型オファーのディメンションには数量という概念はありません。特定のディメンションで契約を結ぶことは、基本的に、購入者がそのディメンションが示す特定の機能を選択したことを意味します。

非階層型オファー - 顧客は契約の一環として複数のディメンションを調達でき、各ディメンションの複数のユニットを調達できます。

このフィールドの値を true に設定すると、そのオファーは非階層型オファーであることを示します。このフィールドに false を設定すると、そのオファーは階層型オファーであることを示します。

製品積載フォーム (PLF) を使用して AMI 製品の契約を作成する場合、料金ディメンションに次のフィールドを定義する必要があります。

- 契約DimensionX API 名 — 購入者の AWS License Manager アカウントで生成されたライセンスに表示される名前。この名前は Checkoutlicense API コールの Entitlement Name の値としても使用されます。
- [契約 DimensionX 表示名] - AWS Marketplace ウェブサイトの製品詳細ページと調達ページに表示される、顧客向けのディメンション名です。わかりやすい名前を付けてください。名前の最大長は 24 文字です。リストが公開されたら、Name の値を変更することはできません。
- [契約 DimensionX 説明] - 特定のディメンションが提供する機能など、製品のディメンションに関する追加情報を提供する、顧客向けのディメンションの説明です。説明の最大長は 70 文字です。

- [契約 DimensionX 数量] - 商品の契約が修正された場合の比例配分の計算に使用されます。このフィールドのこの値は、すべての契約オファーで 1 に設定する必要があります。この値は編集しないでください。
- [契約 DimensionX 1 か月料金] - このディメンションの 1 か月分の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 12 か月料金] - ディメンションに対する 12 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 24 か月料金] - ディメンションに対する 24 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 36 か月料金] - ディメンションに対する 36 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。

例: データストレージアプリケーション

	1 か月料金	12 か月料金	24 か月料金	P36-month price
暗号化されていないデータ (GB)	1.50 USD/GB	16.00 USD/GB	30.00 USD/GB	60.00 USD/GB
暗号化されているデータ (GB)	1.55 USD/GB	16.60 USD/GB	31.20 USD/GB	61.20 USD/GB

例: ログモニタリング製品

	1 か月料金	12 か月料金	24 か月料金	36 か月料金
Basic (10 ホストのモニタリング、5 コンテナのモニタリング)	100 USD	1,000 USD	2,000 USD	4000 USD
Standard (20 ホストのモニタリ	200 USD	2,000 USD	4000 USD	8,000 USD

	1 か月料金	12 か月料金	24 か月料金	36 か月料金
ング、10 コンテナのモニタリング)				
Pro (40 ホストのモニタリング、20 コンテナのモニタリング)	400 USD	4000 USD	8,000 USD	16,000 USD
1 時間あたりのモニタリング対象追加ホスト	10 USD	100 USD	\$200	400 USD
1 時間あたりのモニタリング対象追加コンテナ	10 USD	100 USD	\$200	400 USD

Note

価格は、1 か月、12 か月、24 か月、36 か月の期間に対して定義できます。製品にこれらのオプションを 1 つ以上提供することができます。期間は各ディメンション間で同じである必要があります。

Example

例えば、ReadOnlyUsersと AdminUsersディメンションがある場合、に年間料金を提供する場合は ReadOnlyUsers、AdminUsersにも年間料金を提供する必要があります。

自動更新

お客様が AMI 契約 AWS Marketplace を使用して を通じて製品を購入すると、契約条件を自動的に更新することに同意できます。顧客は 1 年、2 年、または 3 年の期間中に、使用権限に対して毎月支払い続けます。

お客様はいつでも更新の設定を変更できます。詳細については、「AWS Marketplace 購入者ガイド」の「[既存の契約の変更](#)」を参照してください。

AMI 製品の請求、計測、ライセンスの統合

以下のトピックでは、AMI ベースの製品の請求、計測、ライセンスの統合についての情報を説明しています。

トピック

- [AWS Marketplace Metering Serviceを使用した AMI 製品のカスタム計量](#)
- [AWS License Managerとの AMI 製品の契約料金](#)

AWS Marketplace Metering Serviceを使用した AMI 製品のカスタム計量

販売者が使用カテゴリに従って自身のソフトウェアに対して直接請求するために使用できる料金表および計測機能として、AWS Marketplace Metering Service が提供されています。ユーザー、データ、帯域幅、ホスト、ユニットの 5 つの使用カテゴリがあります。Metering Service は、Amazon マシンイメージ (AMI) ベース、コンテナベース、および Software as a Service (SaaS) ベースの製品で利用できます。詳細については、「[AWS Marketplace Metering Service API Reference](#)」を参照してください。

Metering Service を使用するすべての AMI ベースのソフトウェアは次の要件を満たしている必要があります。

- ソフトウェアは、Amazon マシンイメージ (AMI) AWS Marketplace を介して から起動する必要があります。
- に既存の製品がある場合は AWS Marketplace、新しい AMI を送信し、新しい製品を作成してこの機能を有効にする必要があります。
- すべてのソフトウェアは、AWS Identity and Access Management (IAM) ロールでプロビジョニングする必要があります。エンドカスタマーは、ユーザーがそのソフトウェアでプロビジョニングしている Amazon Elastic Compute Cloud (Amazon EC2) インスタンスに IAM ロールを追加する必要があります。を使用してソフトウェアをデプロイする場合、IAM ロールの使用はオプションです AWS Marketplace。これは AWS Marketplace Metering Service ソフトウェアをデプロイするときに必要です。
- ソフトウェアでは、何らかの方法で消費量を決定できる必要があります。

Metering Service を使用する製品では、単一の使用カテゴリごとにお客様に請求する必要がありますが、単一のカテゴリの最大 24 個のディメンションを定義できます。カテゴリに応じて、プロビジョニングされたリソース、同時実行リソース、または累積リソースの消費量によってソフトウェアを価

格設定できます。料金はすべて、お客様によって時間料金で請求されます。すべての使用量は、既存の AWS Marketplace ソフトウェアと同じメカニズムを使用して毎月計算され、請求されます。

AWS Marketplace Metering Service では、いくつかの新しいシナリオが利用可能になります。たとえば、ソフトウェアでホストをモニタリングしている場合は、モニタリングされたホストごとに料金を請求できます。ホストのサイズに基づいて異なる料金を設定でき、モニタリングされる 1 時間ごとの同時実行ホスト数に対して課金できます。同様に、組織全体で多数のユーザーがソフトウェアにサインインできる場合は、ユーザー数によって課金できます。1 時間ごとに、プロビジョニングされたユーザーの合計数が請求されます。

AWS Marketplace Metering Service の呼び出し

ソフトウェアを使用して、時間単位で Metering Service を呼び出し、その時間の消費値を記録する必要があります。

ソフトウェアが起動すると、起動 minute-of-the-hour した が記録されます。これは、start-minute と呼ばれます。起動時間の 1 時間ごとに、ソフトウェアはその時間の消費量を判断し、Metering Service を呼び出します。この値を取得する方法の詳細については、「[Metering Service を使用するためのソフトウェアの変更](#)」を参照してください。

1 時間ごとの開始時間に起動するには、以下のいずれかのアプローチをソフトウェアで使用する必要があります。

- ソフトウェア内のスレッド。
- インスタンスまたはソフトウェアで起動するデーモンプロセス。
- アプリケーションのスタートアップ時に設定される cron ジョブ。

Note

お客様のソフトウェアで、顧客のインスタンスに設定された IAM ロールを使用して AWS Marketplace Metering Service を呼び出し、消費のディメンションと量を指定する必要があります。

ソフトウェアは AWS SDK を使用して、次の実装例のように AWS Marketplace Metering Service を呼び出すことができます。

1. サービスクライアントを作成するにはインスタンスプロファイルを使用します。そのためには、EC2 インスタンス用に設定されたロールが必要です。このロールの認証情報は、SDK によって自動的に更新されます。
2. 1 時間ごとに、ソフトウェア設定とステータスを確認し、その時間の消費値を決定します。これには、 の収集が含まれる場合があります value-per-dimension。
3. 次のパラメータを指定して、SDK クライアントの meterUsage メソッドを呼び出します (使用量があるディメンションごとに追加で呼び出す)。
 - timestamp - 記録中の時間のタイムスタンプ (UTC で)。
 - productCode - ソフトウェアに割り当てられた製品コード。
 - dimension - ソフトウェアに割り当てられた 1 つのディメンション (または複数のディメンション)
 - quantity - ある時間の消費値。
 - allocations - (オプション) 追跡するプロパティ全体にわたって使用量を割り当てることができます。これらの割り当てを合計すると、レコード内の総消費量になる必要があります。購入者には、これらは請求ツール (AWS Billing and Cost Management コンソールなど) に潜在的なコスト配分タグとして表示されます。購入者がこれらのタグを使用してコストを追跡するには、アカウントでタグを有効にする必要があります。

さらに、ソフトウェアを使用して、リージョン内の AWS Marketplace Metering Service エンドポイントを呼び出す必要があります。製品に適切なリージョンのエンドポイントが設定されている必要があります。そのため、記録は us-east-1 より us-east-1 エンドポイント、us-west-2 より us-west-2 エンドポイントに送信されます。リージョン内呼び出しを行うと、購入者にはより安定したエクスペリエンスが提供され、無関係なリージョンのアベイラビリティが別のリージョンで実行されているソフトウェアに影響を及ぼすことを防ぐことができます。

サービスに計測レコードを送信するときは、リージョンの AWS Marketplace Metering Service に接続する必要があります。getCurrentRegion() ヘルパーメソッドを使用して、EC2 インスタンスが実行されているリージョンを特定し、このリージョンの情報を MeteringServiceClient コンストラクタに渡します。SDK コンストラクタ AWS リージョンで を指定しない場合、デフォルトの us-east-1 リージョンが使用されます。アプリケーションでサービスへのクロスリージョンコールを行うと、拒否されます。詳細については、[「アプリケーションの現在のリージョンの決定」](#) および [「\(\)」](#) を参照してください。 [getCurrentRegion](#)

障害処理

製品は、公共のインターネットエンドポイントであるサービスに計測レコードを送信する必要があるため、使用状況をキャプチャして請求することができます。顧客は計測レコードの配信を妨げる方法でネットワーク設定を変更する可能性があるため、製品は障害モードを選択することで考慮する必要があります。

Note

一部の計測障害は、への接続における一時的な問題である可能性があります AWS Marketplace Metering Service。AWS Marketplace では、短期的な停止やネットワークの問題を避けるため、最大 30 分間の再試行をエクスポネンシャルバックオフで実装することを強くお勧めします。

通常、ソフトウェアはフェイルオープン (警告メッセージを表示しながら完全な機能を維持する) か、またはフェイルクローズ (接続が再確立されるまでアプリケーション内のすべての機能を無効にする) を行うことができます。フェイルオープン、フェイルクローズ、またはアプリケーション固有の機能を選択できます。2 時間以内の計測に失敗した場合は、フェイルクローズを行わないことをお勧めします。

部分的なフェイルオープンの場合の一例として、ソフトウェアへのアクセスを引き続き許可することはできますが、購入者がソフトウェア設定を変更することはできません。または、購入者はソフトウェアに引き続きアクセスできますが、追加のユーザーを作成することはできません。ソフトウェアを使用して、この障害モードを定義して実行する必要があります。AMI の送信時、ソフトウェアの障害モードを含む必要があります。また、後に変更することはできません。

制限事項

Metering Service 対応ソフトウェアを設計および送信する場合、次の制約事項に留意してください。

- お客様の IAM ロールとインターネットゲートウェイの要件 - お客様はインターネットゲートウェイを持ち、特定のアクセス許可を持つ IAM ロールでソフトウェアを起動する必要があります。詳細については、「[AWS Marketplace 計測と使用権限 API のアクセス許可](#)」を参照してください。これらのうち 2 つの条件が満たされない場合、ソフトウェアは Metering Service に接続できません。
- 既存の Metering Service product 製品への新しい使用カテゴリの追加および変更の不可 - お客様は、ソフトウェア製品をサブスクライブすると、利用規約に同意したものと見なされま

す。Metering Service を使用して製品の使用カテゴリを変更するには、新しい製品と新しいサブスクリプションが必要です。

- 既存の Metering Service 製品のディメンションの変更の不可 - お客様は、ソフトウェア製品をサブスクライブすると、利用規約に同意したものと見なされます。Metering Service を使用して製品のディメンションを変更するには、新しい製品と新しいサブスクリプションが必要です。既存の製品には、最大 24 個まで新しいディメンションを追加できます。
- 無料トライアルおよび年間サブスクリプションは利用不可 - Metering Service 製品は、起動時に無料トライアルおよび年間サブスクリプションをサポートしていません。
- マルチインスタンスまたはクラスターベースのデプロイメントに関する考慮事項 - 一部のソフトウェアは、マルチインスタンスデプロイメントの一部としてデプロイされます。ソフトウェアの設計時には、消費量を計測する方法と場所、および計測レコードが出力される場所を検討してください。

コード例

次のコード例は、製品の公開およびメンテナンスに必要な AWS Marketplace API と AMI 製品を統合するのに役立つことを目的として提供されています。

使用量配分タグ付き **MeterUsage** (オプション)

次のコード例は、消費料金モデルを持つ AMI 製品に関連しています。Python の例では、適切な使用量割り当てタグを含む計測レコードを AWS Marketplace に送信して、顧客に料金を請求します pay-as-you-go。

```
# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
#       AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited

# Import AWS Python SDK
import boto3
import time

usageRecord = [
    {
        "AllocatedUsageQuantity": 2,
        "Tags":
            [
```

```
        { "Key": "BusinessUnit", "Value": "IT" },
        { "Key": "AccountId", "Value": "123456789" },
    ]
},
{
    "AllocatedUsageQuantity": 1,
    "Tags":
    [
        { "Key": "BusinessUnit", "Value": "Finance" },
        { "Key": "AccountId", "Value": "987654321" },
    ]
}
]

marketplaceClient = boto3.client("meteringmarketplace")

response = marketplaceClient.meter_usage(
    ProductCode="testProduct",
    Timestamp=int(time.time()),
    UsageDimension="Dimension1",
    UsageQuantity=3,
    DryRun=False,
    UsageAllocations=usageRecord
)
```

の詳細についてはMeterUsage、「Metering Service API リファレンス[MeterUsage](#)」の「」を参照してください。AWS Marketplace

レスポンスの例

```
{ "MeteringRecordId": "string" }
```

AWS License ManagerとのAMI製品の契約料金

契約料金のAmazon マシンイメージ (AMI) ベースの製品の場合、AWS License Manager を使用してライセンスを製品に関連付けます。

AWS License Manager は、お客様が購入したライセンス (エンタイトルメントとも呼ばれます) をアプリケーションが追跡および更新できるようにするライセンス管理ツールです。このセクションでは、製品についての情報を提供します。このセクションでは、製品をAWS License Managerと統

合する方法について説明します。統合が完了したら、AWS Marketplaceに製品リストを公開できます。

の詳細については AWS License Manager、[AWS License Manager 「ユーザーガイド」](#) および AWS CLI 「コマンドリファレンス」の[AWS License Manager 「」](#) セクションを参照してください。

Note

- 契約の有効期限を過ぎると、顧客は AMI の新しいインスタンスを起動できなくなります。ただし、契約期間中は、インスタンスをいくつでも起動できます。これらのライセンスはノードロックされたり、特定のインスタンスに結び付けられたりすることはありません。
- [プライベートオファターの作成] - 販売者は、AWS Marketplace 管理ポータルプライベートオファター作成ツールを使用して、製品のプライベートオファターを作成できます。
- [レポート] - AWS Marketplace 管理ポータルの [レポート] セクションで Amazon S3 バケットを設定して、データフィードを設定できます。詳細については、「[出品者レポート、データフィード、ダッシュボード](#)」を参照してください。

ライセンスモデル

AWS Marketplace との統合では、次の 2 つのライセンスモデル AWS License Manager がサポートされています。

- [設定可能なライセンスモデル](#)
- [階層型ライセンスモデル](#)

設定可能なライセンスモデル

設定可能なライセンスモデル (定量化可能なライセンスモデルとも呼ばれます) では、購入者がライセンスを調達した後に、特定の量のリソースを購入者に付与できます。

料金設定ディメンションと単価を設定します。次に、購入者は購入するリソースの数量を選択できます。

Example 料金ディメンションと単価の

料金設定ディメンション (データバックアップなど) と単価 (1 ユニットあたり 30 USD など) を設定できます。

購入者は 5 単位、10 単位、または 20 単位の購入を選択できます。

製品は使用状況を追跡して測定し、消費されたリソースの量を測定します。

設定モデルでは、使用権限は次の 2 つのいずれかにカウントされます。

- [ドローダウンライセンス](#)
- [フローティングライセンス](#)

ドローダウンライセンス

ライセンスは、使用時に許可されたライセンス数のプールから引き出されます。その使用権限は永久にチェックアウトされ、ライセンスプールに戻すことはできません。

Example 限られた量のデータを処理する

ユーザーには 500 GB のデータを処理する権限があります。ユーザーがデータを処理し続けると、500 GB のライセンスがすべて消費されるまで 500 GB のプールからデータが引き出されます。

ドローダウンライセンスの場合は、CheckoutLicense API オペレーションを使用して、消費されたライセンスユニットを確認できます。

Example S3 へのバックアップの年間単位数

Amazon Simple Storage Service (Amazon S3) に、1 年間最大 1,024 単位のデータをバックアップできるストレージ製品があります。アプリケーションは、複数の Amazon EC2 インスタンスを使用して起動できます。アプリケーションには、データを追跡して集約するメカニズムがあります。ソフトウェアは、バックアップのたびに、または一定の間隔で、製品 ID を使用して CheckoutLicense API オペレーションを呼び出し、消費量を更新します。

ソフトウェアが CheckoutLicense を呼び出して 10 ユニットのデータをチェックアウトします。合計容量が顧客が購入したバックアップ制限に達すると、API コールは失敗します。

リクエスト

```
linux-machine ~]$ aws license-manager checkout-license\  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PERPETUAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  
--entitlements "Name=DataConsumption, Value=10, Unit=Count" \  
--client-token "AKIAIOSFODNN7EXAMPLE"
```

レスポンス

```
{
  "CheckoutType": "PERPETUAL",
  "EntitlementsAllowed": [
    {
      "Name": "DataConsumption",
      "Count": 10,
      "Units": "Count",
      "Value": "Enabled"
    }
  ],
  "Expiration": "2021-04-22T19:02:36",
  "IssuedAt": "2021-04-22T18:02:36",
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

フローティングライセンス

ライセンスは、使用後に許可されたライセンス数のプールに返却されます。

Example 固定された上限を超えるユーザー数

1人のユーザーには、アプリケーションで500人の同時ユーザーを割り当てることができます。ユーザーがログインしてログアウトすると、そのユーザーは引き出され、500人のユーザープールに戻ります。ただし、500人の同時ユーザーが固定の上限であるため、アプリケーションはプールから500人を超えるユーザーを引き出すことはできません。

フローティングライセンスの場合は、CheckInLicense API オペレーションを使用してライセンス単位を使用権限プールに戻すことができます。

Example 1年間の同時ユーザー数

製品の料金は、同時接続ユーザー数に基づいて決定されます。顧客は10ユーザー分のライセンスを1年間購入します。顧客はAWS Identity and Access Management (IAM) アクセス許可を与えてソフトウェアを起動します。ユーザーがログインすると、アプリケーションがCheckoutLicense API オペレーションを呼び出して数を1つ減らします。ユーザーがログアウトすると、アプリケーションはCheckInLicense API オペレーションを呼び出してそのライセンスをプールに戻します。CheckInLicense を呼び出さない場合、ライセンス単位は1時間後に自動的にチェックインされます。

Note

次のリクエストでは、key-fingerprint はプレースホルダー値ではなく、すべてのライセンスが公開されるフィンガープリントの実際の値です。

リクエスト

```
linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \
--entitlements "Name=ReadOnlyUSers, Value=10, Unit=Count" \
--client-token "AKIAIOSFODNN7EXAMPLE"
```

レスポンス

```
{
  "CheckoutType": "PROVISIONAL",
  "EntitlementsAllowed": [
    {
      "Name": "ReadOnlyUsers",
      "Count": 10,
      "Units": "Count",
      "Value": "Enabled"
    }
  ],
  "Expiration": "2021-04-22T19:02:36",
  "IssuedAt": "2021-04-22T18:02:36",
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

階層型ライセンスモデル

階層型ライセンスモデルでは、購入者がライセンスを調達した後に、特定のレベル (階層) のアプリケーション機能を利用できるようになります。

製品には、ベーシック、中級、プレミアムなどの階層を作成します。次に、購入者はあらかじめ定義されている階層の1つを選択します。

アプリケーションはアプリケーションの使用状況を追跡したり、測定する必要はありません。

階層型ライセンスモデルでは、使用権限はカウントされず、代わりに顧客が購入したサービスの階層が示されます。

バンドルされた機能を同時に提供する場合は、階層型ライセンスモデルを使用することをお勧めします。

Example ベーシック、中級、プレミアムの各レベル

お客様は、ソフトウェアの3つのレベル(ベーシック、中級、プレミアム)のいずれかの契約に署名できます。これらの階層にはそれぞれ独自の料金設定があります。CheckoutLicense API オペレーションを呼び出し、リクエストで可能なすべての階層を指定して、ソフトウェアが顧客がサインアップした階層を特定できます。

リクエストのレスポンスには、顧客が調達した階層に対応する使用権限が含まれています。この情報に基づいて、ソフトウェアは適切なカスタマーエクスペリエンスを提供できます。

リクエスト

```
linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \
--entitlements "Name=BasicTier, Unit=None" "Name=IntermediateTier, Unit=None" \
"Name=PremiumTier, Unit=None"
```

レスポンス

```
{
  "CheckoutType": "PROVISIONAL",
  "EntitlementsAllowed": [
    {
      "Name": "IntermediateTier",
      "Units": "None"
    }
  ],
  "Expiration": "2021-04-22T19:02:36",
  "IssuedAt": "2021-04-22T18:02:36",
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

ワークフロー統合

以下の手順は、AMI 製品を AWS License Manager と統合するためのワークフローを示しています。

1. 販売者は、AWS License Manager 統合を使用して製品を作成します。
2. 販売者は に製品を一覧表示します AWS Marketplace。
3. 購入者は で製品を見つけ AWS Marketplace て購入します。
4. ライセンスは、AWS アカウントの購入者に送付されます。
5. 購入者は、Amazon Elastic Compute Cloud (Amazon EC2) インスタンス、Amazon Elastic Container Service (Amazon ECS) タスク、または Amazon Elastic Kubernetes Service (Amazon EKS) ポッドソフトウェアを起動してソフトウェアを使用します。顧客は IAM ロールを使用してデプロイします。
6. ソフトウェアは購入者の AWS License Manager アカウントでライセンスを読み取り、購入した使用権限を検出し、それに応じて機能をプロビジョニングします。

Note

License Manager は追跡や更新を行いません。これは販売者のアプリケーションによって行われます。

License Manager 統合の前提条件

製品を公開する前に、次を行う必要があります。

1. で新しい AMI 製品を作成し AWS Marketplace 管理ポータル、その製品コードを書き留めます。
2. 必要な料金情報を製品積載フォーム (PLF) に記入し、処理のために当社に返送してください。
3. CheckoutLicense、ExtendLicenseConsumption、および CheckInLicense を呼び出すために必要な IAM アクセス許可を持つアプリケーションを実行するタスクまたはポッドの IAM ロールを使用します。

必要な IAM アクセス許可の詳細は、次の IAM ポリシーに記載されています。

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

```
"Sid": "VisualEditor0",
"Effect": "Allow",
"Action": [
    "license-manager:CheckoutLicense",
    "license-manager:GetLicense",
    "license-manager:CheckInLicense",
    "license-manager:ExtendLicenseConsumption",
    "license-manager:ListReceivedLicenses"
],
"Resource": "*"
}
]
```

4. 定義したすべての料金ディメンションのレコードを使用して、RegisterUsage API オペレーションをテストコールします。

AMI ベースの製品を と統合する AWS License Manager

[AWS License Manager](#) API を使用して AMI ベースの製品を License Manager と統合できます。AWS Marketplace AMI ベースの製品を使用して Amazon EC2 インスタンスを起動します。

Note

次の手順を実行する前に、[the section called “License Manager 統合の前提条件”](#) を完了していることを確認します。

AMI ベースの製品を License Manager と統合するには

1. [the section called “License Manager でのテストライセンスの作成”](#) のステップを完了します。統合をテストするには、License Manager でテストライセンスを作成する必要があります。
2. ステップ 1 で取得したライセンス Amazon リソースネーム (ARN) を使用して [GetLicense](#) API オペレーションを実行します。後で使用できるように、GetLicense レスポンスの KeyFingerprint 属性の値を書き留めておきます。
3. 最新のパブリック AWS SDK をダウンロードしてアプリケーションに含めます。
4. 購入者がアプリケーションにライセンスを使用する権限があることを確認するには、[CheckoutLicense](#) API オペレーションを実行します。ステップ 1 で取得したテストライセンスの使用権限の詳細とキーフィンガープリントを使用します。

ライセンスに該当する使用権限が見つからないか、使用権限の最大数を越えた場合は、CheckoutLicense API オペレーションから NoEntitlementsAllowedException が返されます。使用権限が有効であるか、使用可能な場合、CheckoutLicense オペレーションはリクエストされた使用権限とその値を含む成功レスポンスを返します。

5. (フローティングエンタイトルメントにのみ必須) CheckoutLicenseレスポンスでLicenseConsumptionToken受信した を使用して [CheckinLicense](#) API オペレーションを実行します。このアクションは、以前にチェックアウトした使用権限を、使用可能な使用権限のプールに戻します。
6. ステップ 1 で作成したテストライセンスと License Manager の統合を正常に確認したら、コード内のキーフィンガープリントを `aws:294406891311:AWS/Marketplace:issuer-fingerprint` に更新します。これで、AWS Marketplaceによって発行されたライセンスを使用する準備ができました。

AMI 製品にアプリケーションを構築するリリースプロセスに従い、製品の公開プロセス AWS Marketplace に従って製品を に送信します。

License Manager でのテストライセンスの作成

(AWS Command Line Interface AWS CLI) のバージョン 2 を使用して、 でテストライセンスを作成します AWS License Manager。このテストライセンスは、AWS License Manager 統合の検証とテストにのみ使用されます。テストが完了したら、テストライセンスを削除できます。実際のライセンスは、別のキーフィンガープリントを使用して AWS Marketplace によって生成されます。

AWS Marketplace は、 で 2 種類の使用権限をサポートします AWS License Manager。ただし、1 つの製品に対して有効にできるのは 1 つのタイプだけです。テストライセンスを含むライセンスを作成するときは、次のどちらかのタイプの資格を指定する必要があります。

階層型ライセンス - 階層型ライセンスモデルでは、顧客に特定のアプリケーション機能の使用権が付与されます。顧客は購入するユニットの数量を定義することはできません。ただし、定義済みのパッケージまたは階層を 1 つ選択することはできます。顧客は後で契約を変更して別の階層にサブスクライブできます。

設定可能な使用権限 - 設定可能なライセンスモデルでは、顧客がライセンスを購入すると、一定量のリソースに使用権限が付与されます。お客様はサブスクリプションプロセス中に購するユニット数を選択し、そのユニット料金に基づいて請求されます。顧客は複数のディメンションをサブスクライブすることもできます。

CheckoutLicense API オペレーションで使用するために必要なパラメータは次のとおりです。

- CheckoutType - 有効な値は Perpetual または Provisional です。
 - Perpetual - チェックアウトされた使用権限の数がプールからなくなる場合に使用されます。
例: 購入者には 500 GB のデータを処理する権限があります。データの処理を続けると、500 GB のプールからその量が引き出され、使い果たされます。購入したライセンスの有効期限が切れている、または間もなく切れるかどうかのステータスを取得して、顧客に通知を送信します。
 - Provisional - 使用権限がプールからチェックアウトされ、使用後に返却されるフローティングライセンス使用権限に使用されます。例: ユーザーには、アプリケーション内で同時に 500 人のユーザーを使用する権利があります。ユーザーがログインしてログアウトすると、そのユーザーは引き出され、500 人のユーザープールに戻ります。フローティングライセンス資格の詳細については、「[AWS License Manager の販売者発行ライセンス](#)」を参照してください。
- ClientToken - 大文字と小文字が区別される一意の識別子。これにより、何度試しても正確な結果が得られ、同じになります。リクエストごとにランダムな共通の一意の識別子 (UUID) を使用することをお勧めします。
- Entitlements - チェックアウトする使用権限のリスト。
 - 階層型使用権限の場合は、Name プロパティと Unit プロパティを次のように指定します。

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "None"
}
```

- 設定可能な使用権限には、Name プロパティ、Unit プロパティおよび Value プロパティを次のように指定します。

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "<Entitlement_Unit>",
  "Value": <Desired_Count>{
}
```

- KeyFingerprint - このキーフィンガープリントを使用して、ライセンスが AWS Marketplace によって発行されたものであることを確認します。によって発行されたライセンスのキーフィンガープリント AWS Marketplace は次のとおりです。

```
aws:294406891311:AWS/Marketplace:issuer-fingerprint
```

- Product SKU – 製品に関連付けられているグローバル一意識別子 (GUID) 形式の AWS Marketplace 製品 ID。

Example 設定可能な使用権限

以下は、CheckoutLicense API オペレーションを使用して PowerUsers という名前の設定可能な使用権限をチェックアウトするリクエストの例です。

```
aws license-manager checkout-license \  
  product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
  checkout-type "PROVISIONAL" \  
  client-token "79464194dca9429698cc774587a603a1" \  
  "Statement":[ \  
  entitlements "Name=PowerUsers,Value=1,Unit=Count" \  
  key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

Example 階層型使用権限

以下は、CheckoutLicense API オペレーションを使用して EnterpriseEdition という名前の機能使用権限をチェックアウトするリクエストの例です。

```
aws license-manager checkout-license \  
  --product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
  --checkout-type "PROVISIONAL" \  
  --client-token "79464194dca9429698cc774587a603a1" \  
  --entitlements "Name=EnterpriseEdition,Unit=None" \  
  --key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

AMI ベース製品のテストライセンスを作成するには

1. AWS CLI v2 がインストールされたローカル環境から、次のスクリプトを実行します。このスクリプトはテストライセンスを作成し、適切な製品詳細を設定します。

Note

ソフトウェアをデプロイしてテスト AWS アカウント するテスト AWS アカウント とは異なる を使用します。ライセンスの作成、付与、チェックアウトを同じ AWS アカウントで行うことはできません。

```
#!/bin/bash

# Replace with intended product ID on AWS Marketplace
PRODUCT_ID=<REPLACE-WITH-PRODUCT-ID>

# Replace with license recipient's AWS Account ID
BENEFICIARY_ACCOUNT_ID=<REPLACE-WITH-BENEFICIARY-ACCOUNT-ID>

# Replace with your product's name
PRODUCT_NAME="Test Product"

# Replace with your seller name on AWS Marketplace
SELLER_OF_RECORD="Test Seller"

# Replace with intended license name
LICENSE_NAME="AWSMP Test License"

# Replace the following with desired contract dimensions
# More info here: https://docs.aws.amazon.com/license-manager/latest/APIReference/API\_Entitlement.html
# Example "configurable entitlement"
ENTITLEMENTS='[
  {
    "Name": "ReadOnly",
    "MaxCount": 5,
    "Overage": false,
    "Unit": "Count",
    "AllowCheckIn": true
  }
]'
# Example "tiered entitlement"
# ENTITLEMENTS='[
#   {
#     "Name": "EnterpriseUsage",
```

```
#      "Value": "Enabled",
#      "Unit": "None"
#    }
#  ]'

# Format "yyyy-mm-ddTHH:mm:ss.SSSZ"
# This creates a validity period of 10 days starting the current day
# Can be updated to desired dates
VALIDITY_START=$(date +%Y-%m-%dT%H:%M:%S.%SZ)
VALIDITY_END=$(date --date="+10 days" +%Y-%m-%dT%H:%M:%S.%SZ)

# Configuration for consumption of the license as set on Marketplace products
CONSUMPTION_CONFIG='{
  "RenewType": "None",
  "ProvisionalConfiguration": {
    "MaxTimeToLiveInMinutes": 60
  }
}'

# License's home Region
HOME_REGION=us-east-1

# License issuer's name
ISSUER=Self

# Run AWS CLI command to create a license
aws license-manager create-license \
  --license-name "${LICENSE_NAME}" \
  --product-name "${PRODUCT_NAME}" \
  --product-sku "${PRODUCT_ID}" \
  --issuer Name="${ISSUER}" \
  --beneficiary "${BENEFICIARY_ACCOUNT_ID}" \
  --validity 'Begin="'"${VALIDITY_START}"'",End="'"${VALIDITY_END}"'"' \
  --entitlements "${ENTITLEMENTS}" \
  --home-region "${HOME_REGION}" \
  --region "${HOME_REGION}" \
  --consumption-configuration "${CONSUMPTION_CONFIG}" \
  --client-token $(uuidgen)
```

2. AWS License Manager コンソールを使用してライセンスを付与します。詳細については、「License Manager ユーザーガイド」の「[使用権限の配布](#)」を参照してください。

3. ソフトウェアをデプロイしてテスト AWS アカウント する購入者アカウントとして機能するにサインインします。これは、ライセンスを作成して付与 AWS アカウントした AWS アカウントとは異なる必要があります。
4. AWS License Manager コンソールに移動して、付与されたライセンスを受け入れてアクティブ化します。詳細については、「AWS License Manager ユーザーガイド」の「[付与されたライセンスの管理](#)」を参照してください。
5. 環境で次のコマンドを実行します。

```
# The following example uses a key fingerprint that should match the test license
you created.
# When checking out an actual AWS Marketplace created license, use the following
fingerprint:
# aws:294406891311:AWS/Marketplace:issuer-fingerprint
aws license-manager checkout-license \
  --product-sku <REPLACE-WITH-PRODUCT-ID> \
  --checkout-type PROVISIONAL \
  --key-fingerprint "aws:<ACCOUNT-ID-WHERE-YOU-CREATED-TEST-LICENSE>:Self:issuer-
fingerprint" \
  --entitlements "Name=ReadOnly,Value=1,Unit=Count" \
  --client-token $(uuidgen)
```

前のコマンドでは、PROVISIONAL を CheckoutType パラメータの値として使用しています。使用権限でドロージャウンライセンスを使用している場合は、値には PERPETUAL を使用します。

License Manager API コール

顧客の License Manager アカウントに保存されているライセンスを管理するために、ソフトウェアは次の API コールを使用できます。

- GetLicense - ライセンスの有効期限が切れているか、もうすぐ期限切れになるかについて、購入したライセンスのステータスを取得して、顧客に通知を送信します。
- CheckoutLicense - ユーザーが購入したライセンスを検出します。ユーザーがいくらかのライセンスを消費した時に、API コールを使用してライセンス数量を更新することもできます。CheckoutLicense を使用すると、顧客が使用したライセンスの数量を継続的に確認できます。顧客がすべてのライセンスを使い果たすと、この呼び出しはエラーを返します。CheckoutLicense を実行するための推奨ケイデンスについては、「[the section called “ライセンスの更新とアップグレード”](#)」を参照してください。

- `ExtendLicenseConsumption` - フローティングディメンションの場合、ソフトウェアがライセンスをチェックアウトすると、60 分後に自動的にライセンスがプールに返却されます。ライセンスがチェックアウトされたままの期間を延長する場合は、ソフトウェアがライセンスをさらに 60 分間延長するように `ExtendLicenseConsumption` を呼び出すことができます。
- `CheckInLicense` - フローティングディメンションの場合、ライセンスを使用権限プールに戻す場合は、`CheckInLicense` を使用してください。
- `ListReceivedLicenses` - 購入者が購入したライセンスを一覧表示します。

ライセンスの更新とアップグレード

顧客は AWS Marketplace 管理ポータルでライセンスを更新またはアップグレードできます。追加購入を行うと、新しい使用権限を反映する新しいバージョンのライセンス AWS Marketplace を生成します。ソフトウェアは同じ API コールを使用して新しい使用権限を読み取ります。License Manager の統合に関しては、更新やアップグレードを処理するために特別なことをする必要はありません。

ライセンスの更新、アップグレード、キャンセルなどが発生するため、製品の使用中は定期的に `CheckoutLicense` API コールを実行することをお勧めします。`CheckoutLicense` API オペレーションを定期的に使用して、製品はアップグレードや有効期限などの使用権限の変更を検出できます。

`CheckoutLicense` API コールは 15 分ごとに実行することをお勧めします。

AMI 製品に関する Amazon SNS 通知

通知を受け取るには、製品の作成時に提供される AWS Marketplace Amazon Simple Notification Service (Amazon SNS) トピックをサブスクライブします。このトピックでは、製品の顧客のサブスクリプションの変更に関する通知が提供されます。例えば、顧客がプライベートオファーをいつ受け入れるかを知ることができます。

Note

製品の作成プロセス中に SNS トピックの実際の Amazon リソースネーム (ARN) に発行されます。例: `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE`

次の Amazon SNS トピックは AMI 製品で使用できます。

- [Amazon SNS トピック: aws-mp-subscription-notification](#) - このトピックは、購入者が製品をサブスクライブまたはサブスクライブを解除した時に通知します。この通知は、時間単位、時間単位、年間料金など、時間単位の料金モデルで利用できます。

Amazon SNS トピック: aws-mp-subscription-notification

aws-mp-subscription-notification トピック内の subscribe-success および subscribe-fail アクションのメッセージは、それぞれ次の形式になっています。

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLEX",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXX",
  "offer-identifier": "offer-abcexample123"
}
```

<action-name> は、通知によって異なります。指定できるアクションは以下のとおりです。

- subscribe-success
- subscribe-fail
- unsubscribe-pending
- unsubscribe-success

offer-identifier は、オファーがプライベートオファーの場合にのみ通知に表示されます。

Amazon SNS トピックへ Amazon SQS キューをサブスクライブする

Amazon SQS キューを指定された SNS トピックにサブスクライブすることを推奨します。SQS キューを作成し、そのキューをトピックにサブスクライブする方法の詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[Amazon SNS トピックへ Amazon SQS キューをサブスクライブする](#)」を参照してください。

Note

AWS Marketplace SNS トピックをサブスクライブできるのは、製品の販売 AWS アカウントに使用したからのみです。ただし、メッセージを別のアカウントに転送することはできません。詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[別](#)

[のアカウントの Amazon SQS キューへ Amazon SNS メッセージを送信する](#)」を参照してください。

通知の SQS キューのポーリング

SQS キューを SNS トピックにサブスクライブすると、メッセージは SQS に保存されます。継続的にキューをポーリングし、メッセージを探して必要に応じて処理するサービスを定義する必要があります。

AMI 製品チェックリスト

Amazon マシンイメージ (AMI) 製品のリクエストを に送信する前に AWS Marketplace、このチェックリストを確認してください。この情報を検証すると、送信から公開プロセスにスムーズに進めることができます。

製品の使用:

- AMI は本番稼働に対応する必要があります。
- AMI は、時間などの測定手段によって製品の使用を制限することができません。
- AMI と 1-Click 履行体験に互換性が必要です。
- 製品を使用するために必要なものはすべて、クライアントアプリケーションを含め、ソフトウェアに含まれます。ソフトウェアパッケージまたはクライアントアプリケーションなど、外部依存関係を必要とする製品は、適切な開示を含む [the section called “製品使用ポリシー”](#) に従う必要があります。
- デフォルトのユーザーはランダム化されたパスワードを使用します。あるいは、初期ユーザーの作成では、インスタンス ID などのインスタンスに固有の値を使用してインスタンスを使用する権限が購入者に付与されていることを確認する必要があります。

無料製品にするか、有料製品にするか。

- 製品を使用するために追加のライセンスは必要ありません。
- 購入者は、製品を使用するために個人を特定できる情報 (E メールアドレスなど) を提供する必要はありません。

AMI の準備

- 製品名と説明は、提供する AMI 製品の [説明] フィールドと一致する必要があります。
- ハードウェア仮想マシン (HVM) の仮想化と 64 ビットアーキテクチャを使用します。
- 既知の脆弱性、マルウェア、またはウイルスは含まれていません。
- 購入者には、AMI へのオペレーティングシステムレベルの管理アクセス権があります。
- AMI セルフサービススキャンを使用して AMI を実行します。

Windows AMI の場合:

- Windows Server 2012 以降のオペレーティングシステムを使用する場合は、[EC2Launch v2](#) の最新バージョンを使用してください。
- AMI が Windows Server 2012 と 2012 R2 上の EC2Config で構築されている場合は、最新バージョンの [EC2Config](#) を使用してください。
- EC2Launch v2 を使用している場合は、次の操作が必要です。
 - [Amazon EC2Launch 設定](#) の [管理者アカウントの設定] で [ランダム] を選択します。
 - [Amazon EC2Launch 設定](#) で、[SSM サービスの開始] の下にある [チェックボックス] を選択します。
 - EC2 v2 タスク設定 UserData に を追加します。 [EC2](#)
- EC2Config を使用している場合は、AMI 内で Ec2、Ec2SetPassword、および Ec2WindowsActivate Ec2HandleUserData [の設定ファイル](#) を有効にします。
- ゲストアカウントまたはリモートデスクトップのユーザーが存在しないことを確認します。

Linux AMI:

- ルートログインはロック/無効になっています。
- 承認されたキー、デフォルトのパスワード、またはその他の認証情報が含まれません。

製品ロードフォームまたは [製品] タブ:

- すべての必須フィールドが入力されています。
- すべての値が指定された文字制限内にあります。
- すべての URL がエラーなしで読み込まれます。
- 製品イメージは、少なくとも 110 ピクセル幅で、1:1 ~ 2:1 の比率です。

- 有効なすべてのインスタンスタイプで料金が指定されています (料金モデルが、時間単位、時間ベースの月額、および時間ベースの年額の場合)。
- 月別料金が指定されています (時間ベースの月額および月額モデルの場合)。

AMI ベースの製品要件

AWS Marketplace は、のすべての Amazon マシンイメージ (AMI) 製品およびサービスについて、次のポリシーを維持します AWS Marketplace。このポリシーは、お客様に安全で安心して信頼できるプラットフォームを提供するためのものです。

すべての製品および関連するメタデータは、送信時にレビューされ、現在の AWS Marketplace ポリシーを満たしているか超えているかが確認されます。これらのポリシーは、進化するセキュリティガイドラインを満たすようにレビューおよび調整されます。は、製品 AWS Marketplace を継続的にスキャンして、セキュリティガイドラインの変更を満たしていることを確認します。製品がコンプライアンス違反になった場合 AWS Marketplace は、新しい基準を満たすように AMI 製品を更新するようから連絡があります。同様に、新たに発見された脆弱性が AMI に影響を与えることが判明した場合は、適切な更新を加えた最新の AMI を提供するように求められます。AMI を送信する前に、[セルフサービス AMI スキャンツール](#)を使用する必要があります。このツールにより、AMI は AWS Marketplace ポリシーを確実に満たすことができます。

セキュリティポリシー

すべての AMI は、次のセキュリティポリシーに従う必要があります。

- AMIs には、[セルフサービス AMI スキャンツール](#)または AWS セキュリティによって検出された既知の脆弱性、マルウェア、またはウイルスを含めないでください。
- AMI は、現在サポートされているオペレーティングシステムやその他のソフトウェアパッケージを使用する必要があります。サポート終了 (EoL) オペレーティングシステムまたはその他のソフトウェアパッケージを搭載した AMI のバージョンはすべて AWS Marketplace から削除されます。更新されたパッケージを使用して新しい AMI を構築し、新しいバージョンとしてに公開できます AWS Marketplace。
- すべてのインスタンス認証では、起動時にパスワードが生成、リセット、または定義されていても、パスワードベースの認証は使用せず、キーペアのアクセスを使用する必要があります。AMI には、どのような理由でも、パスワード、認証キー、キーペア、セキュリティキー、またはその他の認証情報が含まれてはなりません。
- AMI は、AWS リソースにアクセスするためにユーザーからのアクセス/シークレットキーを要求または使用してはなりません。AMI アプリケーションがユーザーにアクセスする必要がある場合

は、でインスタンス化された AWS Identity and Access Management (IAM) ロールを使用して実行する必要があります。これにより AWS CloudFormation、インスタンスが作成され、適切なロールが関連付けられます。AWS CloudFormation 配信方法を使用する製品でシングル AMI 起動が有効になっている場合、対応する使用方法には、最小特権の IAM ロールを作成するための明確なガイドランスを含める必要があります。詳細については、「[the section called “を使用した AMI ベースの配信 CloudFormation”](#)」を参照してください。

- Linux ベースの AMI は SSH パスワード認証を許可してはなりません。PasswordAuthentication を NO に設定して、sshd_config ファイルによるパスワード認証を無効にします。

アクセスポリシー

アクセスポリシーには、一般的なポリシー、Linux 固有ポリシー、および Windows 固有ポリシーの 3 つのカテゴリがあります。

一般的なアクセスポリシー

すべての AMI は、次の一般的なポリシーを遵守する必要があります。

- AMI は、コンプライアンス要件、脆弱性の更新、およびログファイルへのアクセスを可能にするために、オペレーティングシステム (OS) レベルの管理機能を許可する必要があります。Linux ベースの AMI は SSH を使用し、Windows ベースの AMI は RDP を使用します。
- 承認されたパスワードまたは承認されたキーを AMI に含めることはできません。
- AMI は管理アクセスに固定パスワードを使用できません。AMI は代わりにランダムなパスワードを使用する必要があります。代替の実装としてインスタンスメタデータを取得し、パスワードとして instance_id を使用します。管理者は、自身の資格情報を設定または変更することを許可される前に、このランダム化されたパスワードの入力を求められる必要があります。インスタンスメタデータの取得については、Amazon EC2 [ユーザーガイドの「インスタンスメタデータとユーザーデータ」](#)を参照してください。
- お客様の実行中のインスタンスにアクセスすることはできません。お客様は、外部アクセスを明示的に有効にする必要があります。AMI に組み込まれているアクセシビリティはデフォルトでは無効にする必要があります。

Linux 固有の (または Unix のような) アクセスポリシー

Linux ベースまたは Unix のような AMIs は、次のアクセスポリシーと一般的なアクセスポリシーに従う必要があります。

- AMIs、[パスワードベースのリモートログインを無効にする](#)必要があります。
- AMIsルートのリモートログインを無効にする必要があります。
- AMIs、ユーザーがルート機能を実行するための管理者コントロールを取得できるようにする必要があります。例えば、Linux ベースの OS sudoへのアクセスを許可します。他のシステムでは、完全な特権レベルのアクセスを許可します。
- AMIs、監査証跡のルートアクティビティを記録する必要があります。
- AMIsOS ユーザー用に承認されたパスワードを含めることはできません。
- AMIs、承認されたキーを含めることはできません。
- AMIs には、空白または null のルートパスワードを含めることはできません。

Windows 固有のアクセスポリシー

Windows ベースの AMI は、以下のポリシーと一般的なアクセスポリシーを遵守する必要があります。

- Windows Server 2016 以降をお使いの場合は、「EC2Launch」を参照してください。
- Windows Server 2012 R2 以前では、最新バージョンの Ec2ConfigService を使用し、Ec2SetPassword、Ec2WindowsActivate、および Ec2HandleUserData を有効にします。
- ゲストアカウントおよびリモートデスクトップユーザーを削除 (許可されていない) します。

カスタマー情報ポリシー

すべての AMI は、次のお客様情報ポリシーに準拠する必要があります。

- ソフトウェアは、BYOL (Bring Your Own License) で要求される場合を除き、お客様の知識と明示的な同意なしにお客様のデータを収集、またはエクスポートしてはなりません。顧客データを収集またはエクスポートするアプリケーションは、以下のガイドラインに従う必要があります。
 - 顧客データの収集は、セルフサービスで、自動化され、安全である必要があります。購入者は、販売者がソフトウェアの導入を承認するのを待つ必要はありません。
 - 顧客データの要件は、リストの説明または使用説明書に明記する必要があります。これには、収集する内容、顧客データの保存場所、および使用方法が含まれます。例えば、この製品はお客様の名前と E メールアドレスを収集します。この情報は、<会社名> に送信され、保存されます。この情報は、<製品名> に関して購入者に連絡するためにのみ使用されます。
 - 支払情報は収集してはなりません。

製品使用ポリシー

すべての AMI は、次の製品使用ポリシーを遵守する必要があります。

- 製品は、製品または製品機能へのアクセスを時間、ユーザーの数または他の制限事項によって制限してはなりません。ベータ版およびプレリリース版の製品、あるいはトライアルまたは評価機能を提供することのみを目的とした製品はサポートされていません。同等の有料バージョンも AWS Marketplace で入手可能である場合、商用ソフトウェアの開発者、コミュニティ、および BYOL エディションがサポートされています。
- すべての AMI は、ウェブサイトからの起動または AWS CloudFormation を介した AMI ベースの配信のいずれかと互換性がある必要があります。ウェブサイトから起動する場合、AMI はインスタンス作成時にお客様データまたはユーザーのデータを正しく機能させることを要求できません。
- AMI とそのソフトウェアはセルフサービス方式でデプロイ可能である必要があります。追加の支払い方法や費用を必要としないものでなければなりません。デプロイ時に外部に依存する必要があるアプリケーションは、以下のガイドラインに従う必要があります。
 - 要件は、リストの説明または使用説明書に明記する必要があります。例えば、この製品を正しくデプロイするにはインターネット接続が必要です。デプロイ時に以下のパッケージがダウンロードされます。<パッケージのリスト>。
- 販売者は、すべての外部依存関係を使用し、その可用性とセキュリティを確保する責任を負います。
- 外部依存関係が利用できなくなった場合は、からも製品を削除する必要があります AWS Marketplace 。
- 外部依存関係によって追加の支払い方法や費用が必要になってはいけません。
- 購入者の直接管理下でない外部リソースへの継続的な接続を必要とする AMIs。例えば、外部 APIs や販売者や第三者が AWS のサービス管理する AMI は、次のガイドラインに従う必要があります。
 - 要件は、リストの説明または使用説明書に明記する必要があります。例えば、この製品には継続的なインターネット接続が必要です。正しく機能するには、以下の継続的な外部サービスが必要です。<リソースのリスト>。
 - 販売者は、すべての外部リソースを使用し、その可用性とセキュリティを確保する責任を負います。
 - 外部リソースが使用できなくなった場合は、製品 AWS Marketplace も から削除する必要があります。
 - 外部リソースは追加の支払い方法や費用を必要とせず、接続の設定を自動化する必要があります。

- 製品ソフトウェアとメタデータには、AWS Marketplaceでは利用できない他のクラウドプラットフォーム、追加の製品、またはアップセルサービスにユーザーをリダイレクトする言葉を含めてはいけません。
- 製品が別の製品または別の ISV 製品のアドオンである場合、製品の説明には、それが他の製品の機能を拡張するものであり、これがないと製品の有用性が非常に限られることを明記する必要があります。例えば、この製品は <製品名> の機能を拡張するものであり、それがなければ、この製品の有用性は非常に限られています。<製品名> は、このリストのすべての機能を利用するには、独自のライセンスが必要な場合がありますのでご注意ください。

アーキテクチャポリシー

すべての AMI は、次のアーキテクチャポリシーを順守する必要があります。

- のソース AMIs は、米国東部 (バージニア北部) リージョンで提供 AWS Marketplace する必要があります。
- AMI は、HVM 仮想化を使用する必要があります。
- AMI は、64 ビットアーキテクチャーまたは 64 ビット ARM アーキテクチャーを使用する必要があります。
- AMI は、Amazon Elastic Block Store (Amazon EBS) によってサポートされる AMI である必要があります。Amazon Simple Storage Service (Amazon S3) にバックアップされた AMI はサポートされません。
- AMI は暗号化された EBS スナップショットを使用してはいけません。
- AMI は暗号化されたファイルシステムを使用してはいけません。
- AMIs は、すべてので実行でき AWS リージョン、リージョンに依存しないように構築する必要があります。リージョンごとに異なる構築をした AMI は許可されません。

AMI 製品使用説明書

AMI 製品の使用説明書を作成する際は、[the section called “AMI とコンテナ製品の使用手順”](#) に記載されている手順とガイダンスに従ってください。

コンテナベースの製品

AWS Marketplace は、Docker コンテナを使用するソフトウェア製品をサポートします。コンテナ製品は、コンテナイメージとデプロイテンプレートのセットである配信オプションで構成されます。製品に少なくとも 1 つの配信オプション (最大 4 つまで) を送信します。各配信オプションについて、コンテナイメージ、使用手順、購入者がその配信オプションを起動するためのデプロイテンプレートへのリンクを提供します。

AWS Marketplace 購入者は、公開されている製品の詳細ページで利用可能な配信オプションを確認できます。製品をサブスクライブして必要な配信方法を選択すると、購入者には製品の発売および使用に関する情報と手順が表示されます。コンテナイメージの配信オプションについては、購入者には利用可能なデプロイテンプレートとコンテナイメージ URL へのリンクが表示されます。また、個々のコンテナイメージを取得する方法についての説明も受け取ります。Helm チャート配信オプションの場合、購入者には Helm を使用して起動する step-by-step 手順が表示されます。

購入エクスペリエンスのチュートリアルについては、[「Amazon ECS クラスターへの AWS Marketplace コンテナのデプロイ」\(3:34\)](#) の動画を参照してください。

任意の環境の任意の Kubernetes クラスター AWS Marketplace で、からサードパーティーの Kubernetes アプリケーションを検索、サブスクライブ、デプロイできます。Amazon EKS Anywhere (EKS Anywhere) を使用して、Amazon Elastic Container Service (Amazon ECS)、Amazon Elastic Kubernetes Service (Amazon EKS) AWS Fargate、およびオンプレミスにサードパーティーの Kubernetes アプリケーションをデプロイできます。Amazon EKS Anywhere オンプレミスまたは Amazon Elastic Compute Cloud (Amazon EC2) の自己管理型 Kubernetes クラスターにデプロイすることもできます。

Docker 互換のランタイムで Free および BYOL モデル (自分のライセンスを使用する) コンテナ製品を実行できます。

トピック

- [ヘルプの利用](#)
- [コンテナ製品の開始方法](#)
- [コンテナベースの製品要件](#)
- [コンテナ製品の料金](#)
- [コンテナ製品の請求、計測、ライセンスの統合](#)
- [コンテナ製品の Amazon SNS 通知](#)

ヘルプの利用

コンテナ製品に関するヘルプについては、AWS Marketplace の事業開発パートナーに連絡するか、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

コンテナ製品の開始方法

このトピックでは、AWS Marketplaceの初めてのコンテナ製品の作成、テスト、公開に関連するすべての手順について説明します。

トピック

- [前提条件](#)
- [コンテナ製品の作成](#)
- [製品のライフサイクル](#)
- [コンテナ製品 \(レガシー\) の更新](#)
- [製品の可視化の更新](#)
- [AWS アカウント ID の許可リストを更新します。](#)
- [料金設定ディメンションの追加](#)
- [ディメンション情報の更新](#)
- [料金条件の更新](#)
- [国別の可用性の更新](#)
- [エンドユーザーライセンス契約の更新](#)
- [製品の返金ポリシーの更新](#)
- [コンテナ製品の製品 ID と製品コードを作成します。](#)
- [初期リストの作成](#)
- [コンテナ製品 \(レガシー\) の料金詳細の作成または更新](#)
- [コンテナ製品 AWS Marketplace Metering Service との統合](#)
- [コンテナ製品 AWS License Manager との統合](#)
- [製品の新しいバージョンを追加する。](#)
- [製品のテストとリリース](#)
- [バージョン情報の更新](#)

- [Amazon EKS アドオンのバージョンを制限する](#)
- [コンテナ製品の製品情報の作成または更新](#)
- [コンテナ製品の公開 \(レガシー\)](#)
- [コンテナ製品はセキュリティ上の問題をスキャンします。](#)

前提条件

開始するには、以下の前提条件を満たす必要があります。

1. [AWS Marketplace 管理ポータル](#) にアクセスして使用する。これは、販売者として登録し、で販売する製品を管理するために使用するツールです AWS Marketplace。詳細については、「[AWS Marketplace 管理ポータル](#)」を参照してください。
2. 販売者として登録し、税金および銀行の情報を提出します。詳細については、「[販売者登録プロセス](#)」を参照してください。
3. Amazon Elastic Container Service (Amazon ECS)、Amazon Elastic Kubernetes Service (Amazon EKS)、または AWS Fargateに少なくとも 1 つのコンテナを作成します。関連するイメージへのリンクがあることを確認します。
4. AWS Marketplaceでコンテナ製品を作成および統合する方法を計画します。

製品を公開する前に、料金、使用権限管理、および計測戦略を十分に計画することをお勧めします。

- コンテナベースの製品の要件については、「[コンテナベースの製品要件](#)」を参照してください。
- 製品の料金設定の詳細については、「[コンテナ製品の料金 pricing](#)」を参照してください。
- 有料のコンテナベース製品のカスタム計測については、「[AWS Marketplace Metering Serviceによる時間単位計測およびカスタム計測](#)」を参照してください。

コンテナ製品の作成

コンテナ製品の作成には、以下のステップを実行します。

1. [製品 ID とコードを作成する](#)。
2. [the section called “初期リストの作成”](#).
3. [製品の初期バージョンを追加する](#)。
4. 有料製品の場合は、[計測を商品に統合する](#)。

5. [the section called “製品の可視化の更新”](#).

Note

製品のライフサイクルの詳細については、「[the section called “製品のライフサイクル”](#)」を参照してください。

製品のライフサイクル

で製品を作成すると AWS Marketplace、最初に可視性が制限されて公開され、製品を作成したアカウントを含め、許可リストのアカウントがその製品を表示できるようになります。準備ができたら、AWS Marketplace カタログに公開して、購入者が製品をサブスクライブして購入できるようにします。

[サーバー製品](#) ページでは、製品のリストを表示できます。ステージに応じて、製品のステータスは、次のいずれかです。

- ステージング - 情報を追加中の、不完全な製品。セルフサービスエクスペリエンスの最初の保存して終了で、変更リクエストが成功すると、送信した完了したステップの情報を含む未公開の製品が作成されます。このステータスから、引き続き商品に情報を追加したり、変更リクエストを通じて送信済みの情報を変更したりできます。
- 制限あり - 製品がシステムに送信され、システム内のすべての検証に合格すると完成します。その後、製品は制限あり状態にリリースされます。この時点で、製品にはアカウントと許可リストに登録したユーザーだけがアクセスできる詳細ページがあります。商品は詳細ページからテストできます。詳細またはヘルプについては、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。
- パブリック - 購入者が商品を見つけてサブスクライブできるように商品を公開する準備ができたら、可視性を更新する変更リクエストを使用します。このリクエストは、AWS Marketplace Seller Operations チームが AWS ポリシーに照らして製品を確認および監査するためのワークフローを開始します。製品が承認され、変更リクエストが処理されると、製品のステータスが制限ありからパブリックに移行します。AWS ガイドラインの詳細については、「[コンテナベースの製品要件](#)」を参照してください。
- 制限付き - 新規ユーザーによる製品のサブスクライブを停止する場合は、可視性の更新変更リクエストを使用して製品を制限できます。制限付きステータスは、既存の許可リストに登録されているユーザーが引き続き製品を使用できることを意味します。ただし、製品は一般には公開されなくなり、新規ユーザーも利用できなくなります。

Note

ステージング、制限あり、またはパブリックのステータスになったら、製品を更新できません。詳細については、「[コンテナ製品の更新](#)」を参照してください。

コンテナ製品 (レガシー) の更新

コンテナ製品の更新には、以下のステップを実行します。

1. 以下を含む[製品の新しいバージョンを追加します](#)。
 - a. コンテナにリポジトリを追加します。
 - b. 最終的なコンテナをリポジトリにアップロードします。
 - c. 最初のコンテナイメージを使用して、製品の最初のバージョンを作成します。
2. [製品バージョン情報を更新します](#)。
3. [購入者に製品を公開します](#)。

製品の可視化の更新

で製品を表示できる購入者を変更するには AWS Marketplace、可視性の更新 を使用できます。

可視性を更新するには

1. AWS Marketplace 管理ポータル で を開き <https://aws.amazon.com/marketplace/management/tour/>、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページの [現在のサーバー製品] タブで、変更するコンテナベースの製品を選択します。
3. [変更をリクエスト] ドロップダウンから [可視性を更新] を選択します。

Note

この変更リクエストを使用して、製品を [制限あり] ステータスから [パブリック] ステータスに移行するようリクエストできます。ただし、変更リクエストを公開に移行するには、AWS Marketplace Seller Operations チームの承認プロセスを経る必要があります。

4. [送信] を選択して、リクエストを送信します。

5. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。

AWS アカウント ID の許可リストを更新します。

製品を限定された状態で表示できる AWS アカウント IDs のリストを変更できます。許可リストに登録されているアカウントには、製品詳細ページの製品バージョンの横に Limited バッジが表示されます。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[許可リストを更新\]](#) を選択します。現在許可リストに登録されているアカウントのリストが表示されます。
4. [\[許可リストに登録された AWS アカウント\]](#) フィールドに AWS アカウント ID を入力し、カンマで区切ります。
5. [\[送信\]](#) を選択して、リクエストを送信します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) または [\[失敗\]](#) に更新されます。

料金設定ディメンションの追加

製品請求に使用する料金モデルに料金設定ディメンションを追加できます。料金モデルの詳細については、「[コンテナ料金モデル](#)」を参照してください。

Note

使用料金 (例: 追加使用料金) の契約に pay-as-you-go 料金ディメンションを追加することは、では使用できません AWS Marketplace 管理ポータル。
契約、使用量、消費料金が設定された契約の間で料金モデルを変更することはできません。
サポートが必要な場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。

2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[料金設定ディメンションを更新\]](#) を選択し、[\[料金設定ディメンションを追加\]](#) を選択します。
4. 料金モデルによっては、[\[API 識別子\]](#)、[\[表示名\]](#)、[\[説明\]](#) の情報を入力して、契約ディメンションや使用量ディメンションを追加できます。
5. [\[次へ\]](#) を選択し、契約ディメンション料金を入力します。
6. [\[送信\]](#) を選択して、リクエストを送信します。
7. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) または [\[失敗\]](#) に更新されます。

ディメンション情報の更新

製品のディメンション情報を変更できます。料金モデルの詳細については、「[コンテナ料金モデル](#)」を参照してください。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[料金設定ディメンションを更新\]](#) を選択し、[\[ディメンション情報を更新\]](#) を選択します。
4. 更新するディメンションの [\[表示名\]](#) と [\[説明\]](#) に情報を入力して、ディメンション情報を追加できます。
5. [\[送信\]](#) を選択して、リクエストを送信します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) または [\[失敗\]](#) に更新されます。

料金条件の更新

製品の料金条件を変更できます。製品を提供する国を変更した場合、料金条件を更新する必要がある場合があります。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。

3. [変更をリクエスト] ドロップダウンリストから [パブリックオファーを更新] を選択し、[料金条件を更新] を選択します。
4. 現在の料金は編集できるように事前に入力されています。現在の料金を削除して、新しい料金を追加できます。レビューのリクエストを送信する前に、リクエストする料金を確認することをお勧めします。
5. [送信] を選択して、リクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] または [失敗] に更新されます。

Note

[料金条件を更新] を使用してディメンションの料金を引き上げると、90 日間は料金を更新できなくなります。値上げは、変更を開始した日から 90 日間、価格をロックします。この料金ロックは料金の値上げにのみ有効で、値下げには適用されません。

国別の可用性の更新

製品を提供およびサブスクライブできる国を変更することができます。詳細については、「[Countries \(国\)](#)」を参照してください。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル でを開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [変更をリクエスト] ドロップダウンリストから [パブリックオファーを更新] を選択し、[国別の可用性を更新] を選択します。
4. 次のオプションのいずれかを選択します。
 - [すべての国] - サポートされているすべての国で利用できます。
 - [除外対象のすべての国] - 選択した国を除くすべてのサポート対象国で利用できます。
 - [許可リストに登録されている国のみ] - 購入者は指定した国のみで購入できます。
5. [送信] を選択して、リクエストを送信します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] または [失敗] に更新されます。

エンドユーザーライセンス契約の更新

エンドユーザーライセンス契約 (EULA) を更新して、[の標準契約 AWS Marketplace](#) またはカスタム EULA を使用できます。EULA の更新は、製品をサブスクライブする新規ユーザーや製品の更新時に有効になります。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[パブリックオファーを更新\]](#) を選択し、[\[EULA を更新\]](#) を選択します。
4. [\[AWS Marketplace の標準契約\]](#) を選択するか、カスタム EULA を送信してください。カスタム EULA の場合は、Amazon S3 バケットからの契約を提供する必要があります。
5. [\[送信\]](#) を選択して、リクエストを送信します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) または [\[失敗\]](#) に更新されます。

製品の返金ポリシーの更新

製品の返金ポリシーを更新できます。返金ポリシーの更新はすべてのユーザーに有効になります。詳細については、「[での製品の返金 AWS Marketplace](#)」を参照してください。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[サーバー製品\]](#) ページから、変更するコンテナ製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[パブリックオファーを更新\]](#) を選択し、[\[返金ポリシーを更新\]](#) を選択します。
4. 現在の返金ポリシーは編集できるように事前に入力されています。現在の返金ポリシーを削除して、新しい返金ポリシーを追加できます。レビューのリクエストを送信する前に、リクエストする返金ポリシーを確認することをお勧めします。リクエストを送信すると、現在の返金ポリシーが上書きされます。
5. [\[送信\]](#) を選択して、リクエストを送信します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) または [\[失敗\]](#) に更新されます。

コンテナ製品の製品 ID と製品コードを作成します。

コンテナ製品の使用を開始するには、で製品 ID と製品コードレコードを作成する必要があります
AWS Marketplace。製品 ID は、製品のライフサイクル全体を通じて製品を追跡するために使用されます。

で新しいコンテナ製品を作成し、製品 ID を生成するには AWS Marketplace 管理ポータル、次の手順に従います。

Note

このプロセスでは、製品とペアリングするコンテナのパブリックキーも作成されます。

コンテナ製品 ID を作成するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. メニューバーから [製品] を選択し、[サーバー] を選択します。
3. [サーバー製品を作成] を選択し、[コンテナ] を選択します。
4. コンテナ製品 ID とコードを生成します。

Note

(オプション) 商品にタグを付けてタグベースの認証を受けることができます。詳細については、「[Tagging your AWS resources](#)」を参照してください。

5. [続行] を選択して製品の作成を続行します。

初期リストの作成

製品 ID、製品コード、パブリックキーを生成したら、ウィザードを使用して初期リストを作成します。

1. 製品リストの製品情報を入力します。
2. 製品の料金モデルを決定します。

Note

コンテナ製品の詳細については、「[コンテナ製品の料金](#)」を参照してください。

Note

有料製品の場合、AWS Marketplace 出品者と Seller Operations チームが高額な費用をかけずに製品をテストできるように、製品の料金は 0.01 USD から開始します。実際の料金は、公開時に提示します。

3. 返金ポリシー、EULA、オファーの提供状況など、その他のオファー情報を提供してください。
4. コンテナ製品の初期リポジトリを追加します。
5. 最後のステップで [送信] を選択し、製品を限定公開に移行します。

Note

コンテナ製品は最初にプレースホルダーバージョンを使用して作成されます。最終バージョンは、製品の公開範囲が限定されている場合に追加します。

コンテナ製品 (レガシー) の料金詳細の作成または更新

コンテナ製品の料金詳細を更新するには、製品ロードフォーム (PLF) を使用する必要があります。製品の PLF は、製品に関する情報を含むスプレッドシートです。次の手順では、PLF を使用して料金の詳細を含む製品に関する情報を更新する方法を説明します。

Note

コンテナ製品の料金モデルの詳細については、「[コンテナ製品の料金](#)」を参照してください。

料金設定と計測は一致させる必要があります。コンテナ製品による計測の詳細については、「[AWS Marketplace Metering Serviceによる時間単位計測およびカスタム計測](#)」を参照してください。

製品ロードフォームを使用してコンテナ製品の料金を更新するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. メニューバーから、[アセット] を展開し、[ファイルのアップロード] を選択します。
3. 右側の [ダウンロード用製品積載フォーム] から、[コンテナ製品積載フォーム] を選択します。
4. コンピュータ上で PLF スプレッドシートを開き、フィールドに入力して商品を定義します。この情報には、コンテナ製品を作成した時に書き留めた製品 ID が含まれます。

Tip

Microsoft Excel で PLF を表示するときは、各フィールドの上にマウスを移動して、各フィールドに入力する方法に関するガイダンスを示すコメントを表示します。

商品の料金モデルに基づいて、料金および計測ディメンションを指定します。詳細については、次を参照してください。

- [カスタム計測用の製品ロードフォーム](#)
- [時間単位の計測用の製品ロードフォーム](#)

Note

必須フィールドには、スプレッドシートに [REQUIRED] という文字が付いた赤いヘッダーが付いています。リクエストの処理が遅れないように、これらのフィールドはすべて入力してください。

5. PLF を保存します。
6. 開いていない場合は、ウェブブラウザを開いて、[AWS Marketplace 管理ポータル](#) にログインします。
7. メニューバーから [Assets (アセット)] を展開し、[File Upload (ファイルのアップロード)] を選択します。
8. [ファイルをアップロード] でコンピュータを開き、このコンテナ製品用に保存した PLF を選択します。
9. アップロードした他の PLF から識別できるように、簡略な説明を入力します。
10. [アップロード] を選択します。アップロードした PLF は、ページ下部の表に表示されます。

料金の詳細は Seller AWS Marketplace Operations チームによって手動で確認および更新されます。通常、更新が完了するまでに数営業日かかります。状態を確認するには、AWS Marketplace 管理ポータルの [アセット] メニューから [コンテナ] を選択します。製品料金の詳細のレビューが完了すると、E メールメッセージが送信されます。

Note

これで、コンテナ製品が限定された状態で作成されました。アカウントでテスト対象の製品を表示したり、変更できます。他のテストアカウントに表示されるようにしたり、一般公開の準備が整ったら、「[コンテナ製品の公開 \(レガシー\)](#)」を参照してください。

コンテナ製品の料金は、製品を公開するまでは同じ手順で編集できます。

製品の料金詳細を作成すると、他の製品詳細の追加、計測の製品への組み込み、製品のソフトウェアバージョンの作成を行えます。

コンテナ製品 AWS Marketplace Metering Service との統合

使用料金を設定したコンテナベースの製品では、製品を使用する使用権限の確認および請求用使用計測の両方で [AWS Marketplace Metering Service](#) を使用します。料金情報を設定するときに作成した料金モデルに合わせて計測する必要があります。詳細については、「[AWS Marketplace Metering Serviceによる時間単位計測およびカスタム計測](#)」を参照してください。

コンテナ製品 AWS License Manager との統合

契約料金のコンテナベースの製品の場合は、AWS License Manager を使用してライセンスを製品に関連付けます。

との統合の詳細については、AWS License Manager 「」を参照してください。[を使用したコンテナ製品の契約料金 AWS License Manager](#)。

製品の新しいバージョンを追加する。

製品には、その使用期間中に複数のバージョンがある場合があります。各バージョンには、そのバージョン固有のコンテナイメージセットがあります。

Note

製品 ID と製品の料金を設定するまで、製品にバージョンを追加できません。これらのステップの詳細については、「[コンテナ製品の製品 ID と製品コードを作成します。](#)」および「[コンテナ製品 \(レガシー\) の料金詳細の作成または更新](#)」を参照してください。

製品のバージョンを作成するには、以下のステップを実行します。

トピック

- [ステップ 1: リポジトリを追加する](#)
- [ステップ 2: コンテナイメージとアーティファクトをリポジトリにアップロードする](#)
- [ステップ 3: コンテナ製品に新しいバージョンを追加する](#)

製品のコンテナイメージやその他のアーティファクトは、AWS Marketplaceのリポジトリに保存されます。通常、必要なアーティファクトごとに1つのリポジトリを作成しますが、リポジトリにはアーティファクトの複数のバージョンを (異なるタグで) 保存できます。

Note

製品デプロイ内のすべてのイメージは、AWS Marketplace リポジトリのイメージを使用する必要があります。

ステップ 1: リポジトリを追加する

次の手順では、に必要なリポジトリを追加する方法について説明します AWS Marketplace。

リポジトリを追加するには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [製品] メニューで [サーバー] を選択します。
3. [サーバー製品] タブで、変更する製品を選択し、[変更をリクエスト] ドロップダウンから [リポジトリを追加] を選択します。
4. 作成するリポジトリの名前を入力します。複数の新しいリポジトリを作成する場合は、追加するリポジトリごとに [新しいリポジトリの追加] を選択し、一意の名前を付けます。

Note

リポジトリは次のような構造になります。<repositoryID>.dkr.ecr.us-east-1.amazonaws.com/<sellerName>/<repositoryName>。リポジトリにアイテムを追加すると (以下の手順で)、アイテムにはタグが付けられ、この構造になります。<repositoryID>.dkr.ecr.us-east-1.amazonaws.com/<sellerName>/<repositoryName>:<tag>。repositoryID は の内部 ID です AWS Marketplace。sellerName は、販売者アカウント用に作成した名前に基づいています。このステップで repositoryName を定義します。tag は、アーティファクトをリポジトリにアップロードするときに設定されます。

5. [Submit] (送信) を選択します。

Note

製品ごとに最大 50 個のリポジトリを持つことができます。

新しいリクエストが作成され、[リクエスト] タブに表示されます。完了すると、数分以内に、作成したリポジトリにコンテナイメージやその他のアーティファクトを追加できるようになります。

ステップ 2: コンテナイメージとアーティファクトをリポジトリにアップロードする

コンテナイメージとアーティファクトをリポジトリにアップロードするには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [製品] メニューの [サーバー] を選択します。
3. [サーバー製品] タブから、変更する製品を選択します。
4. [変更をリクエスト] ドロップダウンから [リポジトリを追加] を選択します。
5. [既存のリポジトリを表示] を選択します。
6. アップロードするリポジトリを選択します。
7. プッシュコマンドを表示 を選択して、DockerコンテナイメージとHelmチャートをそのリポジトリにプッシュするために使用できるコマンドを含む、手順のリストを開きます。

コンテナイメージやその他のアーティファクトをリポジトリにプッシュする方法に関する一般的な情報については、「Amazon Elastic Container Registry ユーザーガイド」の「[イメージのプッシュ](#)」を参照してください。

 Note

`docker pull` または `docker push` を呼び出す時に、次の Amazon Elastic Container Registry (Amazon ECR) API オペレーションを使用できます。

- `DescribeImages` - これを使用して、リポジトリ内のイメージに関するメタデータを確認します。
- `GetAuthorizationToken` - アーティファクトをリポジトリにアップロードする前に認証を行い、`docker pull` または `docker push` コマンドを使用してください。
- `ListImages` - プッシュしたイメージのリストを表示する場合に使用します。

8. リストされているコマンドを使用して、必要なアーティファクトをローカルリポジトリから製品の AWS Marketplace リポジトリにプッシュします。

 Note

`push` コマンドで指定した **tag** は、リポジトリにアップロードするアーティファクトのバージョンを区別するために使用されます。アーティファクトが含まれるバージョンに適したタグを使用してください。

9. 使用しているバージョンで必要なコンテナイメージまたはアーティファクトごとに同じ手順を繰り返します。

 Note

バージョンには、各配信オプションに最大 50 個のコンテナイメージまたはアーティファクトを含めることができます。配信オプションの詳細については、以下の手順を参照してください。

アーティファクトをアップロードしたら、製品のバージョンを作成する準備が整います。

Note

コンテナイメージは自動的にスキャンされ、[コンテナベースの製品要件](#) を満たしているかどうか確認されます。詳細については、「[コンテナ製品はセキュリティ上の問題をスキャンします。](#)」を参照してください。

新しい配信オプションの追加

コンテナ製品の各バージョンには配信オプションが必要です。配信オプションは、購入者が利用できるデプロイオプションを指定します。以下の配信オプションのいずれかに応じて、適切なアーティファクトをリポジトリにアップロードする必要があります。

- コンテナイメージ配信オプションでは、製品のインストールに必要なすべてのコンテナイメージを AWS Marketplace、コンソールで作成された Amazon Elastic Container Registry (Amazon ECR) リポジトリにアップロードします。
- Helm チャート配信オプションの場合は、AWS Marketplace コンソールで作成された Amazon ECR リポジトリに Helm チャートとコンテナイメージをアップロードします。
- Amazon EKS コンソールのアドオン配信オプションの場合は、AWS Marketplace コンソールで作成された Amazon ECR リポジトリに Helm チャートとコンテナイメージをアップロードします。

ステップ 3: コンテナ製品に新しいバージョンを追加する

Note

コンテナに新しいバージョンを追加するときにエラーが発生した場合は、「[リファレンス](#)」の「[新しいバージョンの非同期エラーの追加](#)」表を参照してください。AWS Marketplace Catalog API

コンテナ製品に新しいバージョンを追加するには

1. [AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューの [サーバー] を選択します。
3. [サーバー製品] タブで、バージョンを追加する製品を選択します。次に、[変更をリクエスト] ドロップダウンから [新しいバージョンを追加] を選択します。

4. [新しいバージョンを追加] ページで、使用しているバージョンの [バージョンタイトル] と [リリースノート] を入力します。
5. バージョンの詳細を入力したら、次のステップとして、配信オプションを追加します。配信オプションとは、購入者が製品バージョンからソフトウェアを起動するために使用できる指示と情報のセットです。配信オプションは、購入者へのフルフィルメントオプションと呼ばれます。

 Note

製品は、異なるコンテナイメージ (Kubernetesや Ubuntuデプロイなど) を持つ複数のプラットフォームをサポートできます。顧客が製品をセットアップする方法ごとに1つの配信オプションを作成し、製品のバージョンごとに最大4つの配信オプションを作成できます。

- a. 製品に他のバージョンで既に配信オプションがある場合は、既存のオプションをテンプレートとして使用して、新しいバージョンに配信オプションを追加できます。[配信オプション] で、追加する配信オプションをリストから選択します。オプションは、以降のステップの指示に従って編集できます。
 - b. 新しい配信オプションを追加するには、[新しい配信オプション] を選択します。オプションを追加したら、以下のステップの指示に従って設定します。
6. 配信オプションの配信方法を選択します。配信方法によって、購入者がソフトウェアをどのように起動するかが決まります。
 - [コンテナイメージ] 配信オプションの場合、AWS Marketplace コンソールで作成された Amazon Elastic Container Registry (Amazon ECR) 内のコンテナイメージへのパスを指定します。購入者はコンテナイメージパスを使用してイメージを自分の環境に直接取り込み、ソフトウェアを起動します。
 - Helm チャート配信オプションでは、AWS Marketplace コンソールで作成された Amazon ECR リポジトリ内のHelmチャートへのパスを指定します。購入者は、デプロイ環境にHelmチャートをインストールしてソフトウェアを起動します。
 - Amazon EKS コンソールのアドオン配信オプションでは、AWS Marketplace コンソールで作成された Amazon ECR リポジトリ内のHelmチャートへのパスを指定します。購入者は、Amazon EKS コンソールまたはネイティブ Amazon EKS アドオン APIsし、ソフトウェアを起動します。詳細については、[「Amazon EKS から利用可能な Amazon EKS アドオン」](#)を参照してください。

- a. [コンテナイメージ] 配信オプションを追加するには、以下のステップを実行します。
 - i. [コンテナイメージ] で、製品バージョンのソフトウェアを含むコンテナイメージに Amazon ECR URL を追加します。
 - ii. [配信オプションのタイトル] と [デプロイオプションの説明] に、この配信オプションのタイトルと説明を入力します。
 - iii. [使用説明書] に、購入者がソフトウェアを起動した後に使用するのに役立つ詳細情報を入力します。
 - iv. [サポート対象のサービス] で、購入者がソフトウェアを起動できる環境を選択します。
 - v. [デプロイテンプレート] に、購入者がソフトウェアの起動に使用できるリソースを追加します。各テンプレートのタイトルとリソースの URL を入力します。
- b. Helm チャート配信オプションを追加するには、次のステップを実行します。
 - i. Helm チャート で、購入者がデプロイ環境にインストールしてソフトウェアを起動する Helmチャートに Amazon ECR URL を追加します。
 - ii. [コンテナイメージ] で、製品バージョンのソフトウェアを含むコンテナイメージに Amazon ECR URL を追加します。
 - iii. [配信オプションのタイトル] と [デプロイオプションの説明] に、この配信オプションのタイトルと説明を入力します。
 - iv. [使用説明書] に、購入者がソフトウェアを起動した後に使用するのに役立つ詳細情報を入力します。
 - v. [サポート対象のサービス] で、購入者がソフトウェアを起動できる環境を選択します。
 - vi. オプション - Helmリリース名 に、HelmチャートをインストールするKubernetes名前空間の名前を入力します。
 - vii. オプション - Helmインストール名前空間 に、 `helm install` コマンドで使用される Helmリリースの名前を入力します。
 - viii. オプション - Kubernetesサービスアカウント名 に、 AWS Identity and Access Management (IAM) への接続に使用されるKubernetesサービスアカウントの名前を入力します。Kubernetes サービスアカウントは、ライセンスや計測などのサービスを呼び出し AWS ます。
 - ix. この製品バージョンQuickLaunchで を有効にするには、 を選択します。QuickLaunch は の機能です AWS Marketplace。購入者は、 QuickLaunch を使用して Amazon EKS クラスターをすばやく作成し、 を使用してそのクラスターでソフトウェアを

起動できます AWS CloudFormation。詳細については、[QuickLaunch 「」 の AWS Marketplace 「」](#) を参照してください。

- x. オーバーライドパラメータで、ソフトウェアを起動する Helm CLI コマンドで使用されるパラメータを入力します。これらのパラメータにより、購入者は指定されたデフォルト値を上書きできます。を有効にしている場合は QuickLaunch、CloudFormation フォームのパラメータ名と説明も入力します。AWS Marketplace マネジメントコンソールを使用する場合、パラメータは 15 個に制限されていますが、を使用する場合、制限はありません AWS Marketplace Catalog API。詳細については、「[コンテナベースの製品への新しいバージョンの追加](#)」を参照してください。

 Note

一部の [パラメータの上書き] は必須です。Amazon EKS Anywhere の製品では、ライセンスシークレットに "\${AWSMP_LICENSE_SECRET}" の DefaultValue を持つ [上書きパラメータ] が必要です。有料製品では、"\${AWSMP_SERVICE_ACCOUNT}" の DefaultValue を持つサービスアカウントの設定に対して 1 つの [上書きパラメータ] を指定する必要があります。

- xi. コンソール、コマンドラインツール、および API の機密情報をマスクするには、[パスワードとシークレットを非表示] を選択します。詳細については、「AWS CloudFormation ユーザーガイド」の「[パラメータ](#)」にある NoEcho パラメータのドキュメントを参照してください。
- c. Amazon EKS コンソールのアドオン配信オプションを追加するには、アーティファクトが準拠していることを確認し [Amazon EKS アドオン製品の要件](#)、次のステップを実行します。

 Note

バージョンごとにサポートされる Amazon EKS アドオン配信オプションは 1 つだけです。使用している現在のバージョンが Amazon EKS コンソールで公開されるまで、新しいバージョンを追加することはできません。

- i. Helm チャートで、購入者がデプロイ環境にインストールしてソフトウェアを起動する Helm チャートに Amazon ECR URL を追加します。

- ii. [コンテナイメージ] で、製品バージョンのソフトウェアを含むコンテナイメージに Amazon ECR URL を追加します。Helm チャート内のすべての画像がリストされていることを確認します。
 - iii. [配信オプションのタイトル] と [デプロイオプションの説明] に、この配信オプションのタイトルと説明を入力します。
 - iv. 可視性 では、選択した制限 のデフォルト値のままにします。
 - v. アドオン名 に、このアドオンの一意の名前を入力します。入力したアドオン名には、Amazon EKS コンソールに表示されるときに販売者の名前が付加されます。
 - vi. アドオンバージョン で、このアドオンのインストールまたはアップグレード時に表示されるアドオンのバージョンを入力します。の形式に従います `major.minor.patch`。
 - vii. アドオンタイプ で、ドロップダウンリストからアドオンのカテゴリを選択します。
 - viii. Kubernetes バージョン で、アドオンがサポートするすべての Kubernetes バージョンを選択します。
 - ix. アーキテクチャ で、アドオンがサポートするプラットフォームアーキテクチャを選択します。オプションは AMD64 と ARM64 です。互換性を最大化するために、両方のアーキテクチャをサポートすることをお勧めします。アドオンが ARM64 デバイスをサポートしていない場合は、すべての商用 で製品を公開する前に、サポートを追加する予定日を指定する必要があります AWS リージョン。
 - x. 名前空間 に、アドオンをインストールする一意の Kubernetes 名前空間を入力します。default、kube-system、および kube-public 名前空間は、サードパーティーアドオンのインストールではサポートされていません。
 - xi. Environment Override パラメータ では、Amazon EKS アドオンフレームワークから最大 2 つの環境パラメータを選択できます。values.yaml のパラメータ名を、`${AWS_REGION}` および `default` であるこれらの環境変数にマッピングできます `${AWS_EKS_CLUSTER_NAME}`。
7. 追加の配信オプションを追加するには、新しい配信オプションを選択し、前のステップの手順を繰り返して設定します。
 8. [送信] を選択します。

製品のテストとリリース

このセクションでは、コンテナ製品のバージョンを公開した後の次のステップに関するガイダンスを提供します。製品のテストと一般公開に必要な具体的な手順とプロセスの概要を説明します。

コンテナイメージとHelmチャートの配信オプション

このセクションでは、コンテナイメージとHelmチャートのリリースに関するガイダンスを提供します。

新しいバージョンのリクエストが作成され、数分以内に完了します。[サーバー製品] ページの [リクエスト] タブからリクエストを追跡できます。アドオンのテストまたはリリース時にエラーが発生した場合は、「AWS Marketplace Catalog API リファレンス」の「[新しいバージョンを追加する](#)」の「非同期エラー」表を参照してください。

Note

製品に現在制限付き可用性が設定されている場合、製品が利用可能である購入者のみが製品バージョンにアクセスできます。製品が現在パブリック可用性に設定されている場合は、すべての AWS Marketplace 購入者が製品バージョンにアクセスできます。

これが初めてのバージョンセットだった場合は、これで製品を公開する準備が整いました。製品の公開方法については、「[コンテナ製品の公開 \(レガシー\)](#)」を参照してください。

Amazon EKS アドオン配信オプション

このセクションでは、Amazon EKS アドオンのテストとリリースに関するガイダンスを提供します。

アドオンをテストする

- アドオンを送信すると、はリクエスト AWS Marketplace を処理し、Amazon EKS アドオンカタログで検証できるように制限された状態でアドオンを公開します。のサーバー製品ページのリクエストタブからリクエストを追跡できます AWS Marketplace 管理ポータル。取り込み時間は、処理するリクエストの量に応じて 5~10 営業日です。

リクエストが「レビュー中」ステータスの場合、アドオンは から Amazon EKS アドオンカタログ AWS Marketplace に AWS チームによって発行されています。アドオンが制限付き状態に公開されると、リクエストのステータスが「成功」に変わります。その後、アドオンのテストを開始できます。

- アドオンが使用可能になったら、テスト目的でアジアパシフィック (ソウル) リージョンで確認できます。AWS Marketplace は、ソフトウェアの機能を検証するための専門知識に依存しています。アドオンをテストするには、アドオンが許可リストに登録されている販売者アカウントのA

アジアパシフィック (ソウル) リージョンに Amazon EKS クラスターを作成する必要があります。アドオンをテストするには、[以下の詳細な手順に従ってください](#)。ソフトウェアがサポートする各 Kubernetes バージョンで必ずテストしてください。

- 有料製品を提供する場合は、次の内部へのプライベートオファーを作成します AWS アカウント。これらのアカウントは、すべての商用の Amazon EKS コンソールにソフトウェアを統合するのに役立ちます AWS リージョン。

288092140294, 288092140294, 408202761791

- ガアドオンバージョン AWS Marketplace を承認してパブリックに移動するまで、テストクラスターをアドオンでアクティブのままにします。

Note

AWS Marketplace は、Amazon EKS クラスターでのコンテナ製品のテスト中に発生する AWS インフラストラクチャコストは負担しません。適切なサイジングメカニズムに従って、テスト結果を検証しながら、ノードを最小限の運用コストに抑えることができます。

アドオンをパブリックにリリースする

アドオンとして Amazon EKS クラスター経由でソフトウェアを検証したら、[AWS Marketplace 管理ポータル](#)または を使用して、Amazon EKS アドオンのバージョンをパブリックにリリースするリクエストを送信できます AWS Marketplace Catalog API。

詳細については、「[AWS Marketplace Catalog API リファレンス](#)」の「[Amazon EKS アドオンの可視性を更新する](#)」を参照してください。

のサーバー製品ページのリクエストタブからリクエストを追跡できます AWS Marketplace 管理ポータル。取り込み時間は異なります。

バージョン情報の更新

バージョンが作成されたら、そのバージョンに関連する情報を変更して、更新された情報を購入者に提供すると便利です。例えば、バージョン 1.1 がリリースされた後にバージョン 1.0 を制限する予定であれば、バージョン 1.0 の説明を更新して、購入者をバージョン 1.1 に誘導することができます。バージョン 1.0 が制限される日付を入力してください。バージョン情報は AWS Marketplace 管理ポータルから更新します。

バージョン情報を更新するには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [製品] メニューで [サーバー] を選択します。
3. [サーバー製品] タブで、変更する製品を選択します。
4. [変更をリクエスト] ドロップダウンから [バージョン情報を更新] を選択します。
5. [バージョンを更新] ページで、更新するバージョンを選択します。
6. 選択したバージョンを更新します。更新できるフィールドは、製品バージョンまたは配信オプションのステータスによって異なります。
 - a. すべてのバージョンで、[リリースノート] を更新できます。
 - b. まだ公開されていないバージョンでは、[バージョンタイトル] を更新できます。
 - c. 制限されていない配信オプションについては、以下のフィールドを更新できます。
 - 説明
 - [使用手順]
 - [サポートされるサービス]
 - d. まだ公開されていないバージョンの配信オプションについては、以下のフィールドを更新できます。
 - [配信オプションタイトル]
 - Helm チャート (Helmチャート配信オプションのみ)
 - コンテナイメージ
 - [デプロイリソース]
 - AddOn 名前
 - AddOn バージョン
 - AddOn タイプ
 - Helm チャート URI
 - CompatibleKubernetesバージョン
 - SupportedArchitectures
 - 名前空間
 - EnvironmentOverrideパラメータ

- e. 公開されているバージョンの配信オプションについては、[を更新できません Supported Architectures](#)。
7. [送信] を選択します。
 8. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。

リクエストのステータスは、[サーバー製品](#) ページの [リクエスト] タブからいつでも確認できます。

Amazon EKS アドオンのバージョンを制限する

アドオンとして公開されているコンテナ製品のバージョンを制限するには、[の下部にある AWS Marketplace お問い合わせフォーム](#)を使用して運用チームにお問い合わせください。[AWS Marketplace 管理ポータル](#)。

コンテナ製品の製品情報の作成または更新

製品 ID を作成して料金を設定したら、AWS Marketplace でコンテナ製品について顧客に表示される内容など、製品情報を編集できます。以下の手順では、製品の製品詳細を作成する手順を概説しています。

コンテナ製品の製品詳細を作成または更新するには

1. [AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューで [サーバー] を選択します。
3. [サーバー製品] タブで、変更する製品を選択します。
4. [変更をリクエスト] ドロップダウンから [製品情報を更新] を選択します。
5. 変更する次のフィールドのいずれかを更新します。
 - [製品のタイトル]
 - [SKU]
 - 短い説明
 - [詳しい説明]
 - [製品ロゴイメージ URL]
 - [ハイライト]
 - [製品カテゴリ]

- キーワード
- [製品動画 URL]
- リソース
- [サポート情報]

Note

イメージ URL は、パブリックにアクセス可能な Amazon S3 バケットに含まれている必要があります。ロゴフォーマットの詳細については、「[会社と製品のロゴ要件](#)」を参照してください。

6. [送信] を選択します。
7. リクエストが [リクエスト] タブに [レビュー中] ステータスで表示されていることを確認します。リストにリクエストが表示されるようにするには、ページの更新が必要になる場合があります。

リクエストのステータスは、[サーバー製品](#) ページの [リクエスト] タブからいつでも確認できます。

コンテナ製品の公開 (レガシー)

最初に製品を作成した時点では、利用できるのはアカウントのみに制限されています。製品をテストする準備ができたなら (製品の詳細を入力し、最初のバージョンを作成することを含む)、その製品を他のアカウントでテスト用に使用したり、公開製品としてすべてのアカウントで使用できます。

Note

パブリックに公開する前に、製品が正確であること (イメージリンク、デプロイテンプレート、説明、料金など) を確認する必要があります。パブリックに公開された製品については、料金モデルを変更することはできません。

限定商品を追加のアカウントに公開したり、一般に公開したりするには、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。リクエストには、製品 ID を入力し、加える変更を記述します。

Note

購入者に提供しないバージョンを制限することもできます。公開されている商品にテストバージョンが表示されないように、これを製品の公開リクエストに含めることができます。公開されている製品から公開バージョンがないままになる場合は、そのバージョンを制限することはできません。

コンテナ製品を公開すると、製品をサブスクライブして起動できるすべての AWS 顧客に表示されます。AWS Marketplace Seller Operations チームは、製品情報内のデータと、へのテストコールを確認します AWS Marketplace Metering Service。

コンテナ製品はセキュリティ上の問題をスキャンします。

コンテナ製品に新しいバージョンを追加する変更リクエストを作成すると、その新しいバージョンに含まれるコンテナイメージがスキャンされ、セキュリティの脆弱性がチェックされます。これを行うには、イメージに対して layer-by-layer 静的スキャンを実行します。リモートで悪用可能なリスクベクトルがある重大な脆弱性が見つかった場合は、見つかった問題のリストが表示されます。取り込みと公開プロセスの遅延を避けるには、Clair、Twistlock、Aqua Security、Trend Micro などのコンテナイメージスキャナーを使用して、独自のセキュリティ分析を実行することを強くお勧めします。

コンテナイメージを構築するためのベースイメージの選択は、最終的なイメージのセキュリティプロファイルに大きな影響を与えます。既知の重大な脆弱性があるベースイメージを選択すると、アプリケーションソフトウェアのレイヤーがクリーンであってもベースレイヤーが原因でフラグ付けされます。イメージをビルドして送信する前に、脆弱性のないベースコンテナから開始していることを確認することをお勧めします AWS Marketplace。

コンテナベースの製品要件

AWS Marketplace は、のすべてのコンテナベースの製品およびサービスについて、次の要件を維持します AWS Marketplace。これらの要件は、お客様に安全、安心、および信頼できるカタログを提供するのに役立ちます。また、販売者には、特定の商品のニーズを満たすために、必要に応じて追加の規制やプロトコルの導入を検討するよう奨励しています。

すべての製品および関連するメタデータは、送信時にレビューされ、現在の AWS Marketplace 要件を満たしているか、超えているかが確認されます。進化するセキュリティやその他の使用要件を満たすために、これらのポリシーを見直して調整します。は、既存の製品がこれらの要件に対する変更を引き続き満たしていること AWS Marketplace を継続的に検証します。製品がコンプライアンス違

反になった場合は、AWS Marketplace から製品を更新するよう連絡されます。場合によっては、問題が解決されるまで、新しいサブスクライバーが製品を一時的に利用できないことがあります。

セキュリティ要件

すべてのコンテナベースの製品は、以下の要件を満たしている必要があります。

- Docker コンテナイメージには、既知のマルウェア、ウイルス、または脆弱性が含まれていない必要があります。コンテナ製品に[新しいバージョンを追加する](#)と、そのバージョンに含まれるコンテナイメージがスキャンされます。
- コンテナベースの製品で AWS リソースを管理するためのアクセスが必要な場合は、ユーザーにアクセスキーをリクエストするのではなく、[サービスアカウントの IAM ロール](#) (Amazon Elastic Kubernetes Service (Amazon EKS) を介して実行する場合) または [タスクの IAM ロール](#) (Amazon Elastic Container Service (Amazon ECS) を介して実行する場合) を使用してアクセスする必要があります。
- コンテナベースの製品の実行には最小特権のみが必要です。詳細については、「[ECS セキュリティ](#)」と「[EKS セキュリティ](#)」を参照してください。
- コンテナイメージは、デフォルトで非ルートの権限で実行するように設定する必要があります。

アクセスの要件

すべてのコンテナベースの製品は、以下のアクセス要件を満たしている必要があります。

- コンテナベースの製品では、ランダム化された初期パスワードを使用する必要があります。コンテナベースの製品では、外部管理アクセス (ウェブインターフェイス経由でアプリケーションにログインする場合など) に初期固定パスワードや空白パスワードを使用してはなりません。購入者は、自身の認証情報を設定または変更することを許可される前に、このランダム化されたパスワードの入力を求められる必要があります。
- アプリケーションへの外部からのアクセスは、お客様による明示的な同意と有効化が必要です。

顧客情報の要件

すべてのコンテナベースの製品は、次の顧客情報の要件に従う必要があります。

- ソフトウェアは、BYOL (Bring Your Own License) で要求される場合を除き、お客様の知識と明示的な同意なしにお客様のデータを収集、またはエクスポートしてはなりません。顧客データを収集またはエクスポートするアプリケーションは、以下のガイドラインに従う必要があります。

- 顧客データの収集は、セルフサービスで、自動化され、安全である必要があります。購入者は、販売者がソフトウェアの導入を承認するのを待つ必要はありません。
- 顧客データの要件は、リストの説明または使用説明書に明記する必要があります。これには、収集する内容、顧客データの保存場所、および使用方法が含まれます。例えば、この製品はお客様の名前と E メールアドレスを収集します。この情報は、<会社名> に送信され、保存されます。この情報は、<製品名> に関して購入者に連絡するためにのみ使用されます。
- 支払情報は収集してはなりません。

製品の使用要件

すべてのコンテナベースの製品は、次の製品使用の要件に従う必要があります。

- 販売者は完全に機能する商品のみを出品できます。トライアルまたは評価を目的としたベータ版またはプレリリース版の製品は許可されていません。商用ソフトウェアのデベロッパー、コミュニティ、BYOL エディションは、販売者が無料エディションの提供から 90 AWS Marketplace 日以内に同等の有料バージョンを に提供した場合、サポートされます。
- コンテナベースの製品のすべての使用説明書には、コンテナベースの製品をデプロイするためのすべての手順が記載されている必要があります。使用説明書には、上の対応するコンテナイメージを指すコマンドとデプロイリソースが記載されている必要があります。AWS Marketplace
- コンテナベースの製品には、サブスクライバーがソフトウェアを使用するために必要なすべてのコンテナイメージが含まれている必要があります。さらに、コンテナベースの製品では、ユーザーが外部からのイメージ AWS Marketplace (サードパーティーのリポジトリからのコンテナイメージ など) を使用して製品を起動する必要はありません。
- コンテナとそのソフトウェアはセルフサービス方式でデプロイ可能である必要があります。追加の支払い方法や費用を必要としないものでなければなりません。デプロイ時に外部に依存する必要のあるアプリケーションは、以下のガイドラインに従う必要があります。
 - 要件は、リストの説明または使用説明書に明記する必要があります。例えば、この製品を正しくデプロイするにはインターネット接続が必要です。デプロイ時に以下のパッケージがダウンロードされます。<パッケージのリスト>。
- 販売者は、すべての外部依存関係を使用し、その可用性とセキュリティを確保する責任を負います。
- 外部依存関係が利用できなくなった場合は、からも製品を削除する必要があります AWS Marketplace 。
- 外部依存関係によって追加の支払い方法や費用が必要になってはいけません。

- 購入者の直接の管理下でない外部リソース (外部 API、販売者または第三者が管理する AWS のサービス など) への継続的な接続を必要とするコンテナは、以下のガイドラインに従う必要があります。
- 要件は、リストの説明または使用説明書に明記する必要があります。例えば、この製品には継続的なインターネット接続が必要です。正しく機能するには、以下の継続的な外部サービスが必要です。<リソースのリスト>。
- 販売者は、すべての外部リソースを使用し、その可用性とセキュリティを確保する責任を負います。
- 外部リソースが使用できなくなった場合は、製品 AWS Marketplace も から削除する必要があります。
- 外部リソースは追加の支払い方法や費用を必要とせず、接続の設定を自動化する必要があります。
- 製品ソフトウェアとメタデータには、AWS Marketplaceでは利用できない他のクラウドプラットフォーム、追加の製品、またはアップセルサービスにユーザーをリダイレクトする言葉を含めてはいけません。
- 製品が別の製品または別の ISV 製品のアドオンである場合、製品の説明には、それが他の製品の機能を拡張するものであり、これがないと製品の有用性が非常に限られることを明記する必要があります。例えば、この製品は <製品名> の機能を拡張するものであり、それがなければ、この製品の有用性は非常に限られています。<製品名> は、このリストのすべての機能を利用するには、独自のライセンスが必要な場合がありますのでご注意ください。

アーキテクチャの要件

すべてのコンテナベースの製品は、次のアーキテクチャの要件に従う必要があります。

- のソースコンテナイメージは、 が所有する Amazon Elastic Container Registry (Amazon ECR) リポジトリにプッシュ AWS Marketplace する必要があります AWS Marketplace。これらのリポジトリは、コンテナ製品リストごとにサーバー製品の下に AWS Marketplace 管理ポータル に作成できます。
- コンテナイメージは Linux ベースである必要があります。
- 有料のコンテナベースの製品は [Amazon ECS](#)、[Amazon EKS](#)、または [AWS Fargate](#) にデプロイできる必要があります。
- 契約料金ととの統合を伴う有料コンテナベースの製品は、Amazon EKS、Amazon ECS、Amazon EKS Anywhere AWS Fargate、Amazon ECS Anywhere、Red Hat OpenShift Service on AWS

(ROSA)、オンプレミスのセルフマネージド Kubernetes クラスター、または Amazon Elastic Compute Cloud にデプロイ AWS License Manager する必要があります。

コンテナ製品の使用手順

コンテナ製品の使用説明書を作成するときは、「[the section called “AMI とコンテナ製品の使用手順”](#)」の手順とガイダンスに従ってください。

Amazon EKS アドオン製品の要件

Amazon EKS アドオンは、Kubernetesアプリケーションに運用機能を提供するソフトウェアですが、アプリケーションに固有ではありません。例えば、Amazon EKS アドオンには、クラスターがネットワーク、コンピューティング、ストレージの基盤となる AWS リソースとやり取りできるようにするオブザーバビリティエージェントまたはKubernetesドライバーが含まれています。

コンテナ製品の販売者は、Amazon EKS を含むいくつかのデプロイオプションから選択できます。製品のバージョンを AWS Marketplace アドオンとして Amazon EKS アドオンカタログに公開できます。アドオンは Amazon EKS コンソールの、AWS および他のベンダーが管理するアドオンの横に表示されます。購入者は、他のアドオンと同様に簡単にソフトウェアをアドオンとしてデプロイできます。

詳細については、Amazon EKS ユーザーガイドの [Amazon EKS アドオン](#) を参照してください。

コンテナ製品を AWS Marketplace アドオンとして準備する

コンテナ製品を AWS Marketplace アドオンとして公開するには、次の要件を満たしている必要があります。

- コンテナ製品は で公開する必要があります AWS Marketplace。
- コンテナ製品は AMD64 と ARM64 の両方のアーキテクチャに対応するように構築されている必要があります。
- コンテナ製品で Bring Your Own License (BYOL) [料金モデル](#) を使用してはなりません。

Note

BYOL は Amazon EKS アドオン配信ではサポートされていません。

- すべての [コンテナイメージとチャートをマネージド Amazon ECR リポジトリにプッシュするなど、コンテナベースのすべての製品要件](#) に従う必要があります。Helm AWS Marketplace この要

件には、などのオープンソースイメージが含まれますnginx。イメージとチャートは、[Amazon ECR Public Gallery](#)、Docker Hub、Quay などを含むがこれらに限定されない他の外部リポジトリではホストできません。

- Helm グラフ - ソフトウェアをHelmチャートでデプロイする準備をします。Amazon EKS アドオンフレームワークは、Helmグラフをマニフェストに変換します。一部のHelm機能は Amazon EKS システムではサポートされていません。次のリストは、オンボーディング前に満たす必要がある要件を示しています。このリストでは、すべてのHelmコマンドがHelmバージョン 3.8.1 を使用します。
- は例外として、すべてのCapabilitiesオブジェクトがサポートされています.APIVersions。 .APIVersionsは non-built-inカスタム Kubernetes APIsではサポートされていません。
- Release.Name および Release.Namespace オブジェクトのみがサポートされています。
- Helm フックと lookup関数はサポートされていません。
- すべての依存グラフは、メインHelmチャート内に配置する必要があります (リポジトリパスファイル://... で指定)。
- Helm グラフは、エラーなしでHelmリントとHelmテンプレートを正常に渡す必要があります。コマンドは次のとおりです。
- Helm リント - `helm lint helm-chart`

一般的な問題には、親グラフのメタデータに宣言されていないグラフが含まれます。例えば、次のようになります: `chart metadata is missing these dependencies: chart-base Error: 1 chart(s) linted, 1 chart(s) failed`

- Helm テンプレート - `helm template chart-name chart-location -set k8version=Kubernetes-version -kube-version Kubernetes-version -namespace addon-namespace -include-crds -no-hooks -f any-overridden-values`

オーバーライドされた設定を `-f` フラグで渡します。

- すべてのコンテナバイナリを AWS Marketplace Amazon ECR リポジトリに保存します。マニフェストを作成するには、前述のHelmテンプレートコマンドを使用します。マニフェストで、busyboxや イメージなどの外部gcrイメージリファレンスを検索します。リクエストドロップダウンのリポジトリの追加オプションを使用して作成された AWS Marketplace Amazon ECR リポジトリに、依存関係とともにすべてのコンテナイメージをアップロードします。
- カスタム設定 — デプロイ中にカスタム変数を追加できます。エンドユーザーエクスペリエンスを識別し、ソフトウェアに という名前を付けaws_mp_configuration_schema.json、Helm

チャートでラッパーにパッケージ化する方法については、[「Amazon EKS アドオン: 詳細設定」](#)を参照してください。

[「\\$schema」キーワードに従って](#)、は有効なapplication/schema+jsonリソースを指す URI \$schemaである必要があります。

このファイルは、パスワード、ライセンスキー、証明書などの機密情報を受け付けてはなりません。

シークレットと証明書のインストールを処理するには、アドオン後またはアドオン前のインストール手順をエンドユーザーに提供できます。製品は外部ライセンスに依存しないでください。製品は使用 AWS Marketplace 権限に基づいて機能する必要があります。

の制限の詳細については、aws_mp_configuration_schema.json「」を参照してください。[アドオンプロバイダーのアドオン設定要件とベストプラクティス](#)。

- ソフトウェアをデプロイする名前空間を特定して作成します。製品の最初のリリースでは、テンプレート化された名前空間を追加して、ソフトウェアをデプロイする名前空間を特定する必要があります。
- **serviceAccount** 該当する場合に を作成する – ソフトウェアが の有料ソフトウェアであるか、他の に接続 AWS Marketplace する必要がある場合は AWS のサービス、HelmチャートがserviceAccountデフォルトで を作成することを確認してください。serviceAccount 作成がvalues.yamlファイルのパラメータによって処理される場合は、パラメータ値を に設定しますtrue。例えば serviceAccount.create = true です。これは、お客様が、必要なアクセス許可がすでに付与されている基盤となるノードインスタンスからアクセス許可を継承してアドオンをインストールすることを選択する可能性があるために必要です。Helm チャートが を作成しない場合serviceAccount、アクセス許可を に関連付けることはできませんserviceAccount。
- 追跡可能なデプロイまたはデーモンセット – Helm チャートにデーモンセットまたはデプロイがあることを確認します。Amazon EKS アドオンフレームワークは、それらを使用して Amazon EKS リソースのデプロイを追跡します。追跡可能なデプロイまたはデーモンセットがない場合、アドオンはデプロイエラーに直面することになります。アドオンにデプロイやデーモンセットがない場合、例えば、アドオンが一連のカスタムリソースや追跡不可能な Kubernetes ジョブをデプロイする場合は、ダミーデプロイまたはデーモンセットオブジェクトを追加します。
- AMD および ARM アーキテクチャのサポート — 多くの Amazon EKS のお客様は、現在 ARM64 を使用して AWS Graviton インスタンスを使用しています。サードパーティーソフトウェアは両方のアーキテクチャをサポートしている必要があります。
- のライセンスまたは計測 APIs と統合すると AWS Marketplace、複数の請求モデル AWS Marketplace がサポートされます。詳細については、[「コンテナ製品の請求、計測、ライセンスの](#)

[統合](#)」を参照してください。PAYG メカニズムを使用して製品を販売する場合は、「」を参照してください[AWS Marketplace Metering Service によるコンテナ製品のカスタム計測](#)。前払いモデルまたは契約モデルを使用して製品を販売する場合は、「」を参照してください[を使用したコンテナ製品の契約料金 AWS License Manager](#)。

- ソフトウェアとすべてのアーティファクトと依存関係をアップロードする – Helm チャートは自己完結型でなければならず、などの外部ソースからの依存関係を必要としない必要があります GitHub。ソフトウェアに外部依存関係が必要な場合は、依存関係を同じ AWS Marketplace リストの AWS Marketplace プライベート Amazon ECR リポジトリにプッシュする必要があります。
- ウェブサイトにデプロイ手順を提供する – お客様が [create-addon](#) コマンドを使用してソフトウェアをデプロイする方法を特定するためのデプロイガイドをホストすることをお勧めします。
- IAM ロール – ソフトウェアが機能したり、他のに接続したりするために必要なすべての AWS Identity and Access Management (IAM) ポリシーを一覧表示します AWS のサービス。
- バージョンの更新 – Amazon EKS は、アップストリームリリースから数週間後に新しい Kubernetes バージョンをリリースします。新しい Amazon EKS クラスターバージョンが一般公開されると、ベンダーは新しい Amazon EKS クラスターバージョンリリースと互換性があるソフトウェアを認証または更新するまで 45 日間かかります。アドオンの現在のバージョンが新しい Kubernetes バージョンをサポートしている場合は、バージョン互換性マトリックスを更新できるように、同じバージョンを検証して認証します。新しい Kubernetes バージョンリリースをサポートするために新しいアドオンバージョンが必要な場合は、オンボーディング用の新しいバージョンを送信してください。
- パートナーのソフトウェアは、次のいずれかのタイプに該当するか、Kubernetes または Amazon EKS を強化する運用ソフトウェアである必要があります。Gitops | モニタリング | ログ記録 | 証明書管理 | ポリシー管理 | コスト管理 | 自動スケーリング | ストレージ | kubernetes-管理 | service-mesh | etcd-backup | ingress-service-type || ロードバランサー | ローカルレジストリ | ネットワーク | セキュリティ | バックアップ | イングレスコントローラー | オブザーバビリティ
- ソフトウェアを[コンテナネットワークインターフェイス \(CNI\)](#) にすることはできません。
- ソフトウェアは を通じて販売 AWS Marketplace され、有料製品のライセンスおよび計測 APIs と統合されている必要があります。BYOL 製品は受け付けられません。

アドオンプロバイダーのアドオン設定要件とベストプラクティス

Amazon EKS では、アドオンプロバイダーの [Helm JSON スキーマ](#) 文字列として設定する必要があります。必要な設定を必要とするアドオン、またはオプションの設定を許可するアドオンには、に送信された Helm チャートを含む `aws_mp_configuration_schema.json` ファイルを含める必要があります AWS Marketplace。Amazon EKS は、このスキーマを使用して顧客からの設定入力を検

証し、スキーマに準拠しない入力値を持つ API コールを拒否します。アドオン設定は通常、次の 2 つのカテゴリに分類されます。

- ラベル、許容範囲、nodeSelector などの一般的な Kubernetes プロパティの設定。
- ライセンスキー、機能有効化、URLs など、アドオン固有の設定。

このセクションでは、一般的な Kubernetes プロパティに関連する最初のカテゴリに焦点を当てています。

Amazon EKS では、Amazon EKS アドオンの設定に関するベストプラクティスに従うことをお勧めします。

- [スキーマの要件](#)
- [設定で許可される一般的なパラメータ](#)
- [設定に許可されていない一般的なパラメータ](#)

スキーマの要件

json スキーマを定義するときは、Amazon EKS アドオンでサポートされている jsonschema のバージョンを使用してください。

サポートされているスキーマのリスト：

- <https://json-schema.org/draft-04/schema>
- <https://json-schema.org/draft-06/schema>
- <https://json-schema.org/draft-07/schema>
- <https://json-schema.org/draft/2019-09/schema>

他の json スキーマバージョンを使用すると、Amazon EKS アドオンと互換性がなく、これが修正されるまでアドオンをリリースできなくなります。

Helm スキーマファイルの例

```
{
  "$schema": "http://json-schema.org/schema#",
  "type": "object",
  "properties": {
```

```
"podAnnotations": {
  "description": "Pod Annotations"
  "type": "object"
  },
  "podLabels": {
  "description": "Pod Labels"
  "type": "string"
  },
  "resources": {
  "type": "object"
  "description": "Resources"
  },
  "logLevel": {
  "description": "Logging Level"
  "type": "string",
  "enum": [
    "info",
    "debug"
  ]
  },
  "config": {
  "description": "Custom Configuration"
  "type": "object"
  }
}
```

camelCase

設定パラメータは camelCase にする必要があり、この形式に従わない場合、拒否されます。

説明が必要です

スキーマプロパティのわかりやすい説明を必ず含めてください。この説明は、各設定パラメータの Amazon EKS コンソールでラベル名をレンダリングするために使用されます。

RBAC 定義

アドオンプロバイダーは、最小特権の原則を使用して、アドオンを正常にインストールするために必要な RBAC アクセス許可を定義して提供する必要があります。新しいバージョンのアドオンまたは CVE に対応するための修正について RBAC アクセス許可を変更する必要がある場合、アドオンプロバイダーは Amazon EKS チームに対し、この変更を通知する必要があります。各 Kubernetes リソースに必要なアクセス許可は、オブジェクトのリソース名に制限する必要があります。

```
apiGroups: ["apps"]
resources: ["daemonsets"]
resourceNames: ["ebs-csi-node"]
verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
```

シークレット管理

このセクションは、お客様がアプリケーションキー、API キー、パスワードなどのシークレット情報を設定する必要があるアドオンにのみ適用されます。現在、Amazon EKS APIs、セキュリティ上の影響により、プレーンテキストでの機密情報の受け渡しをサポートしていません。ただし、お客様は設定を使用して、アドオンに必要なキーを保持する Kubernetes シークレットの名前を渡すことができます。お客様は、前提条件のステップと同じ名前空間を持つキーを含む Kubernetes Secret オブジェクトを作成し、アドオンの作成時に設定 BLOB を使用して Secret の名前を渡す必要があります。アドオンプロバイダーは、お客様が誤って実際のキーと間違えないように、スキーマプロパティに名前を付けることをお勧めします。例: appSecretName connectionSecretName など。

要約すると、アドオンプロバイダーはスキーマを活用して、お客様がシークレットの名前を渡すことはできますが、実際にシークレット自体を保持するキーは渡すことはできません。

設定値の例

スキーマに設定例を含めると、お客様がアドオンを設定できるようになります。次の例は、Distro for AWS OpenTelemetry アドオンのスキーマからのものです。

```
"examples": [
  {
    "admissionWebhooks": {
      "namespaceSelector": {},
      "objectSelector": {}
    },
    "affinity": {},
    "collector": {
      "amp": {
        "enabled": true,
        "remoteWriteEndpoint": "https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx/api/v1/remote_write"
      },
      "cloudwatch": {
        "enabled": true
      },
      "mode": "deployment",
```

```
"replicas": 1,
"resources": {
  "limits": {
    "cpu": "256m",
    "memory": "512Mi"
  },
  "requests": {
    "cpu": "64m",
    "memory": "128Mi"
  }
},
"serviceAccount": {
  "annotations": {},
  "create": true,
  "name": "adot-collector"
},
"xray": {
  "enabled": true
}
},
"kubeRBACProxy": {
  "enabled": true,
  "resources": {
    "limits": {
      "cpu": "500m",
      "memory": "128Mi"
    },
    "requests": {
      "cpu": "5m",
      "memory": "64Mi"
    }
  }
},
"manager": {
  "env": {},
  "resources": {
    "limits": {
      "cpu": "100m",
      "memory": "128Mi"
    },
    "requests": {
      "cpu": "100m",
      "memory": "64Mi"
    }
  }
}
```

```

    }
  },
  "nodeSelector": {},
  "replicaCount": 1,
  "tolerations": []
}
]

```

設定で許可される一般的なパラメータ

以下は、顧客向け Helm スキーマファイルで推奨されるパラメータです。

パラメータ	説明	にはデフォルトが必要ですか？
additionalLabels	アドオンによって管理されるすべての Kubernetes オブジェクトに Kubernetes ラベルを追加します。	いいえ
additionalAnnotations	アドオンによって管理されるすべての Kubernetes オブジェクトに Kubernetes 注釈を追加します。	いいえ
podLabels	アドオンによって管理されるポッドに Kubernetes ラベルを追加します。	いいえ
podAnnotations	アドオンによって管理されるポッドに Kubernetes 注釈を追加します。	いいえ
logLevel	アドオンによって管理されるコンポーネントのログレベル。	はい
nodeSelector	ノード選択制約の最も簡単な推奨形式。nodeSelector フィールドをポッド仕様に追加し、ターゲットノードに付けるノードラベルを指定できます。	例えば、Linux ノードのみ

パラメータ	説明	にはデフォルトが必要ですか？
tolerations	許容範囲はポッドに適用されます。許容範囲により、スケジューラはテイントが一致するポッドをスケジューリングできます。許容範囲ではスケジューリングが可能ですが、スケジューリングは保証されません。	おそらく、デーモンセットでより一般的
アフィニティ	アフィニティ機能は 2 種類のアフィニティで構成されます。nodeSelector フィールドのようなノードアフィニティ関数ですが、より表現力が高く、ソフトルールを指定できます。ポッド間アフィニティ/アンチアフィニティを使用すると、他のポッドのラベルに対してポッドを制限できます。	おそらく
トポロジSpreadConstraints	トポロジスプレッド制約を使用して、リージョン、ゾーン、ノード、その他のユーザー定義トポロジドメインなどの障害ドメイン間でポッドをクラスター全体に分散する方法を制御できます。これにより、高可用性と効率的なリソース使用率を実現できます。	おそらく
リソースのリクエスト/制限	各コンテナに必要な cpu/メモリの量を指定します。リクエストを設定することを強くお勧めします。制限はオプションです。	はい
レプリカ	アドオンによって管理されるポッドのレプリカの数。デーモンセットには適用されません。	はい

Note

ワークロードスケジューリング設定パラメータの場合、必要に応じてスキーマの最上位コンポーネントを分離する必要がある場合があります。例えば、Amazon EBS CSI ドライバーには、コントローラーとノードエージェントという 2 つの主要コンポーネントが含まれています。お客様は、コンポーネントごとに異なるノードセレクトタ/許容範囲を必要とします。

Note

JSON スキーマで定義されているデフォルト値は、純粋にユーザードキュメントのみを目的としており、`values.yaml` ファイルで適切なデフォルトを設定する必要を置き換えるものではありません。デフォルトプロパティを使用する場合は、のデフォルトがスキーマのデフォルト`values.yaml`と一致し、Helm チャートが変更されるたびに 2 つのアーティファクト (`values.schema.json` と `values.yaml`) が同期されたままであることを確認してください。

```
"affinity": {
  "default": {
    "affinity": {
      "nodeAffinity": {
        "preferredDuringSchedulingIgnoredDuringExecution": [
          {
            "preference": {
              "matchExpressions": [
                {
                  "key": "eks.amazonaws.com/compute-type",
                  "operator": "NotIn",
                  "values": [
                    "fargate"
                  ]
                }
              ]
            },
            "weight": 1
          }
        ]
      },
      "podAntiAffinity": {
```

```
    "preferredDuringSchedulingIgnoredDuringExecution": [
      {
        "podAffinityTerm": {
          "labelSelector": {
            "matchExpressions": [
              {
                "key": "app",
                "operator": "In",
                "values": [
                  "ebs-csi-controller"
                ]
              }
            ]
          },
          "topologyKey": "kubernetes.io/hostname"
        },
        "weight": 100
      }
    ]
  },
  "description": "Affinity of the controller pod",
  "type": [
    "object",
    "null"
  ]
}
```

設定に許可されていない一般的なパラメータ

clusterName、region、などのクラスターメタデータパラメータはvpcIdaccountId、さまざまなアドオン (Elastic Load Balancing Controller など) で必要になる場合があります。Amazon EKS サービスで既知のこれらに類似するパラメータは、Amazon EKS アドオンによって自動的に挿入され、ユーザーが設定オプションとしてを指定する責任はありません。これらのパラメータには以下が含まれます。

- AWS リージョン
- Amazon EKS クラスター名
- クラスターの VPC ID
- コンテナレジストリ、特にネットワークアドオンで使用される build-prod アカウント用

- DNS クラスター IP、特に coredns アドオン用
- Amazon EKS クラスター API エンドポイント
- クラスターで有効になっている IPv4
- クラスターで有効になっている IPv6
- クラスターで有効になっている IPv6 のプレフィックス委任

アドオンプロバイダーは、該当するパラメータに対してテンプレートが定義されていることを確認する必要があります。上記の各パラメータには、Amazon EKS で定義された事前定義されたparameterType属性があります。リリースメタデータは、parameterTypeとテンプレート内のパラメータの名前/パス間のマッピングを指定します。これにより、お客様が設定を通じてこれらを指定しなくても、Amazon EKS によって値を動的に渡すことができ、アドオンプロバイダーが独自のテンプレート名/パスを定義できる柔軟性も得られます。Amazon EKS が動的に挿入する必要がある上記のようなパラメータは、スキーマファイルから除外する必要があります。

リリースメタデータからのマッピングの例

```
"defaultConfiguration": [
  {
    "key": "image.containerRegistry",
    "parameterType": "CONTAINER_REGISTRY"
  }
]
```

以下は、顧客向け Helm スキーマファイルで設定することを推奨しないパラメータです。パラメータには変更不可能なデフォルトを含めるか、アドオンテンプレートにまったく含めないでください。

パラメータ	説明	にはデフォルトが必要ですか？
画像	Kubernetes クラスターにデプロイされるコンテナイメージ。	いいえ、アドオン定義で管理されます
イメージPullSecrets	シークレットを使用してプライベートレジストリからプルするようにポッドを設定します。	該当なし

パラメータ	説明	にはデフォルト が必要ですか？
livenessProbe	Kubelet プロセスでは、ライブネスプローブを使用して、コンテナを再起動するタイミングを把握します。例えば、ライブネスプローブは、アプリケーションが実行されているが、進行できないデッドロックをキャッチする可能性があります。このような状態のコンテナを再起動すると、バグが発生してもアプリケーションを利用しやすくなる可能性があります。	はい
readinessProbe	コンテナの準備状況プローブを用意することが重要です。これにより、データプレーンで実行されている Kubelet プロセスが、コンテナがトラフィックを処理する準備ができたことを認識します。ポッドは、すべてのコンテナの準備が整うと準備完了と見なされます。このシグナルの1つの用途は、サービスのバックエンドとして使用されるポッドを制御することです。ポッドの準備が整わないと、サービスロードバランサーからポッドが削除されます。	はい

パラメータ	説明	にはデフォルト が必要ですか？
startupProbe	<p>kubelet は、スタートアッププローブを使用して、コンテナアプリケーションがいつ起動したかを把握します。このようなプローブが設定されている場合、成功するまでライブネスチェックと準備状況チェックを無効にし、それらのプローブがアプリケーションの起動を妨げないことを確認します。これは、起動速度の遅いコンテナにライブネスチェックを採用するために使用でき、起動して実行する前にkubeletによって強制終了されるのを防ぐことができます。</p>	<p>オプションです。</p>
ポッドDisruptionBudget	<p>Pod Disruption Budget (PDB) を定義して、ボランティアによる中断中に PODS の最小数を実行し続けます。PDB は、レプリケートされたアプリケーションのポッドのうち、ボランティアによる中断によって同時にダウンするポッドの数を制限します。例えば、クォーラムベースのアプリケーションは、実行中のレプリカの数にクォーラムに必要な数を下回ることがないようにしたいと考えています。ウェブフロントエンドは、ロードを処理するレプリカ数が合計の特定の割合を下回ることがないようにしたい場合があります。</p>	<p>はい。デフォルトで3つ以上のレプリカに設定されている場合</p>
serviceAccount (名前)	<p>サービスアカウントポッドの名前はで実行されます。</p>	<p>はい</p>

パラメータ	説明	にはデフォルト が必要ですか？
serviceAccount (注釈)	サービスアカウントに適用される注釈。通常、サービスアカウントの IAM ロール機能に使用されます。	いいえ。IAM サービスアカウントロール ARN は、最上位の Amazon EKS アドオン API で設定されます。このルールの例外は、アドオンに複数のデプロイ/コントローラー (Flux など) があり、個別の IRSA ロール ARNs。
優先度ClassName	Priority は、他のポッドと比較したポッドの重要性を示します。ポッドをスケジュールできない場合、スケジューラは優先度の低いポッドをプリエンプ (エビクシオン) して、保留中のポッドのスケジューリングを有効にします。	はい。ほとんどのアドオンはクラスター機能にとって重要であり、デフォルトで優先度クラスを設定する必要があります。
ポッドSecurityContext	セキュリティコンテキストは、ポッドまたはコンテナの権限とアクセスコントロールの設定を定義します。通常、fsGroup の設定に使用されます。これは v1.19 以下のクラスターで IRSA に必要でした。	Amazon EKS が Kubernetes v1.19 をサポートしなくなった場合、可能性は低い
securityContext	セキュリティコンテキストは、ポッドまたはコンテナの権限とアクセスコントロールの設定を定義します。	はい

パラメータ	説明	にはデフォルトが必要ですか？
updateStrategy	古いポッドを新しいポッドに置き換えるために使用される戦略を指定します。	はい
nameOverride	ポッドの名前を上書きします。	いいえ
ポッドSecurityPolicy	パラメータに制限を適用します。	いいえ - PSPs非推奨です
Extra VolumeMounts/extraVolumes	Amazon EKS 以外のクラスターの IRSA に使用されます。	いいえ

コンテナ製品の料金

このセクションでは、コンテナ製品で利用できる料金モデルについて説明します。Amazon Elastic Container Service (Amazon ECS)、Amazon Elastic Kubernetes Service (Amazon EKS)、AWS Fargateについて、無料製品、Bring Your Own License model (BYOL) 製品、有料製品を出品できます。1つの製品につき1つの料金設定ができます。

Note

[AWS Marketplace Metering Service](#) を使用して、有料製品の使用権限管理および計測を強化します。タスク単位またはポッド単位の料金については、AWSによって自動的に使用量が計測されます。

コンテナ製品に設定した料金は、すべての AWS リージョンに適用されます。コンテナ製品の料金を下げるたびに、新しい料金が購入者に即時に実装されます。料金を上げる場合、変更が請求に反映される 90 日前に既存の購入者はその変更について通知されます。新規購入者には新しい金額が請求されます。

Note

新規サブスクリイバーの場合、料金変更はすぐに有効になります。既存のサブスクリイバーの場合、料金変更は、料金変更通知が送信された日から始まる 90 日間の期間の翌月の初日

に有効になります。例えば、3月16日に料金変更通知を送信したとします。6月16日は3月16日から約90日後です。料金変更は90日間の期間の翌月の初日に行われるため、変更の発効日は7月1日です。

コンテナ料金モデル

AWS Marketplace には、コンテナ製品の複数の料金モデルがあります。

次の表に、コンテナベースの製品の料金モデルに関する一般的な情報を示します。

コンテナ製品の料金モデル

料金モデル	説明
Bring-Your-Own-License (BYOL)	BYOL は、購入者と維持する外部請求関係 AWS Marketplace を通じての外部で管理されます。
月別	<p>[月額固定料金]</p> <p>ユーザーに翌月中の無制限の製品使用を提供する固定月額料金</p> <p>例: 製品に 1 か月あたり 99 USD の価格を設定します。製品には、Amazon ECS タスク定義を使用してデプロイされた 3 つの異なるコンテナイメージが含まれます。</p> <p>購入者が製品をサブスクライブした後、すぐに 99 USD が課金され、サブスクリプションをキャンセルするまで毎月繰り返されます。また購入者は、製品の無制限の使用を取得します。また、購入者はタスクが実行されているすべてのインフラストラクチャで個別に支払いを行います。サブスクライブ中は、コンテナイメージにアクセスできます。これらのイメージから、任意の設定の Amazon ECS または Amazon EKS で、任意の数のコンテナを起動して実行できます。</p> <p>購入者が 1 か月の途中でサブスクリプションをキャンセルすると、がコンテナイメージ AWS Marketplace を保存する Amazon ECR リポジトリにアクセスできなくなります。購入者が元のイメージを取り出して保存した可能性があります。ただし、で利用できる新しいコンテナイメージバージョンをプルすることはできなくなりました AWS Marketplace。購入者には、</p>

料金モデル	説明
	最終月の未使用分が返金されます。購入者の使用量から合意された AWS Marketplace 料金を引いた金額に基づいてお支払いいただきます。
カスタムメトリクス料金ディメンション	<p>定義されたディメンション (ユーザー、ノード、リポジトリ、または GB など) に基づくカスタム従量制料金 (製品あたり最大 24 ディメンション)。</p> <p>例: ユーザーによる製品課金。管理者ユーザーと通常ユーザーがあり、管理ユーザーの料金を 2 USD、通常ユーザーの料金を 1 USD と定義します。商品を出品する際にそれぞれに異なるディメンションを設定できます。1 日にログインしたユーザーごとに課金され、一日あたりの使用量が計測されます。</p>

料金モデル	説明
<p>タスクあたりまたはポッドあたりの時間単位料金</p>	<p>[Amazon ECS タスクまたは Amazon EKS ポッド]</p> <p>Amazon ECS タスクごとあるいは Amazon EKS ポッドごとに 1 時間あたりの料金設定による秒単位の測定。</p> <p>例: 製品には、コントローラーノード、ワーカーノード、分析ノードの 3 つの異なるコンテナイメージがあります。コントローラーノードがなければ製品は機能しなかったり、役に立たなかったりするため、使用量を課金するイメージを決めます。1 時間あたり 6 USD の料金を設定します。</p> <p>コントローラーノードのコンテナイメージでソフトウェアを変更して、AWS Marketplace Metering Service RegisterUsage API オペレーションと統合します。これにより、アクティブなサブスクリプションがある購入者だけが、そのコンテナイメージを起動して実行することができます。その使用量は実行された時間に基づいて計測されます。</p> <p>購入者は、実行中の各 Amazon EKS コントローラーポッドの使用量について、1 時間あたり 6 USD が課金されます。購入者がコントローラーノードコンテナを含む 5 つの Amazon EKS コントローラポッドを起動すると、1 時間あたり 30 USD (ポッドあたり 6 USD) が課金されます。また、購入者は、ポッドが実行するインフラストラクチャに対して個別に支払いを行います。</p> <p>時間単位の課金の場合、課金は秒単位で、最小は 1 分です。登録者がこのコントローラーコンテナを 20 分 30 秒実行すると、$20 \times (\\$6/60) + 30 \times (\\$6/60/60) = \\$2 + \\$0.05 = \\$2.05$ の料金が課金されます。購入者の使用量から合意された AWS Marketplace 料金を引いた金額に基づいてお支払いいただきます。</p>

料金モデル	説明
時間単位/長期契約による使用量	<p>割引価格による長期契約、前払いまたは定期的な分割払い。長期契約は、カスタム従量制料金が設定されている既存の製品、またはタスクおよびポッドごとに追加できます。購入者が長期契約による購入を超えた消費を行うと、従量制料金を支払います。</p> <p>従量制料金モデルでは、購入者に対して長期契約価格を追加して、前払いのコミットの割引を受けることができます。例えば、通常、消費したユニット数につき 1 USD を請求するとします。1 時間あたり 1 ユニットを使用する購入者は、年間 8760 USD (365 days x 24 hours x \$1 per hour) を支払うこととなります。購入者がその 365 日間、1 時間あたり 1 ユニットを半額 (4380 USD) で使用できるようにする契約を有効にできます。この場合、購入者は 1 年契約の前払いを約束し、料金は 1 ユニットあたり 1 ドルから 0.5 ドルに下がります。また、購入者がこれらの契約を複数購入できるようにすることもできます。計測された数量から、購入者が 1 時間に 10 ユニットを消費したことが示され、2 つの契約を結んでいた場合、2 つの契約には 2 ユニットが含まれます。追加の 8 ユニットには、通常の 1 時間あたり 1 USD、その 1 時間の合計で 8 USD が請求されます。</p> <p>タスク単位またはポッド単位の例では、購入者向けに長期契約料金を追加して、前払いのコミットの割引を受けることができます。通常、ポッドあたり 6 USD を請求する場合は、長期契約期間を 365 日に 13,140 USD (365 days x 24 hours x \$3 per pod per hour) の料金に設定できます。そうすれば、1 つの契約で、その 365 日間、1 時間あたり 1 ポッドをお客様に提供することとなります。お客様は複数の契約を購入することを選択できます。例えば、1 人のお客様が 2 つの契約を購入すると、1 時間に 2 つのポッドを利用できるようになります。お客様が 1 時間あたりに実行するポッドの数が、対象となる契約数よりも多い場合、超過分のポッドには通常の時間単価で請求されます。</p> <p>いずれの場合も、長期契約を購入する購入者は、1 回限りの支払いまたは定期的に予定されるこの先の支払いのどちらに対しても、前払い請求されます。また、購入者は計測率時に契約範囲を超えた追加使用量に対して請求されます。</p>

料金モデル	説明
コンテナ契約料金表	[契約料金の AMI] - 購入者が前払い料金を支払うコンテナベースの製品。

コンテナ製品の契約料金

契約料金のコンテナベースの製品の場合、は、お客様とお客様との契約に基づいて、前払いまたは定義した支払いスケジュールでお客様に AWS Marketplace 請求します。その時点で、それらのリソースの使用権限が付与されます。

料金を設定するには、顧客に提供する 1 つ以上の契約期間を選択します。契約期間ごとに異なる価格を入力できます。契約期間は 1 か月、12 か月、24 か月、36 か月です。プライベートオファーでは、カスタム期間を月単位 (最大 60 か月) で指定できます。

製品の価格を最もよく表すカテゴリを選択します。料金カテゴリは、AWS Marketplace ウェブサイトの顧客に表示されます。[帯域幅] (GB/s、MB/s)、[データ] (GB、MB、TB)、[ホスト]、[リクエスト]、[階層]、[ユーザー] のいずれかを選択できます。事前定義カテゴリのどれもニーズに適合しない場合は、汎用的な [単位] カテゴリを選択できます。

このオファーには最大 24 のディメンションを追加できます。各ディメンションには次のデータが必要です。

- 契約カテゴリ - 製品が契約料金に加えて消費量ベースの計測をサポートしている場合、契約カテゴリは製品の測定または計測に使用されます。消費ベースの料金設定のない契約製品の場合は、契約に含まれるディメンションのカテゴリに最も近いカテゴリを選択できます。契約内のディメンションの単位に似た値がない場合は、Units を選択してください。
- 契約単位 - 製品が消費量ベースの計測をサポートしている場合、契約単位はカテゴリと共に計測に使用されます。選択したカテゴリに基づいて、ディメンションに最も近い単位の値を 1 つ選択します。
- 契約ディメンションは複数購入が可能 - このフィールドは、オファーが階層型の料金設定のオファーなのか、複数のディメンションの購入を許可する非階層型のオファーなのかを示すために使用されます。

階層型オファー - 購入者がオファー内の利用可能なディメンションのうちの一つだけをサブスクライブできるようにします。階層型オファーのディメンションには数量という概念はありません。特定のディメンションで契約を結ぶことは、基本的に、購入者がそのディメンションが示す特定の機能を選択したことを意味します。

非階層型オファー - 顧客は契約の一環として複数のディメンションを調達でき、各ディメンションの複数のユニットを調達できます。

このフィールドの値を true に設定すると、そのオファーは非階層型オファーであることを示します。このフィールドに false を設定すると、そのオファーは階層型オファーであることを示します。

ウィザードを使用してコンテナ製品の契約を作成する場合、料金ディメンションに次のフィールドを定義する必要があります。

- 契約DimensionX API 名 — 購入者の AWS License Manager アカウントで生成されたライセンスに表示される名前。この名前は Checkoutlicense API コールの Entitlement Name の値としても使用されます。
- [契約 DimensionX 表示名] - AWS Marketplace ウェブサイトの製品詳細ページと調達ページに表示される、顧客向けのディメンション名です。わかりやすい名前を付けてください。名前の最大長は 24 文字です。リストが公開されたら、Name の値を変更することはできません。
- [契約 DimensionX 説明] - 特定のディメンションが提供する機能など、製品のディメンションに関する追加情報を提供する、顧客向けのディメンションの説明です。説明の最大長は 70 文字です。
- [契約 DimensionX 数量] - 商品の契約が修正された場合の比例配分の計算に使用されます。このフィールドのこの値は、すべての契約オファーで 1 に設定する必要があります。この値は編集しないでください。
- [契約 DimensionX 1 か月料金] - このディメンションの 1 か月分の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 12 か月料金] - ディメンションに対する 12 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 24 か月料金] - ディメンションに対する 24 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。
- [契約 DimensionX 36 か月料金] - ディメンションに対する 36 か月間の使用権限に対して請求される契約レート。非階層型オファーの場合、このレートは調達されたディメンションの単位ごとに請求されます。このフィールドでは小数点以下 3 桁がサポートされています。

例: データストレージアプリケーション

	1 か月料金	12 か月料金	24 か月料金	P36-month price
暗号化されていないデータ (GB)	1.50 USD/GB	16.00 USD/GB	30.00 USD/GB	60.00 USD/GB
暗号化されているデータ (GB)	1.55 USD/GB	16.60 USD/GB	31.20 USD/GB	61.20 USD/GB

例: ログモニタリング製品

	1 か月料金	12 か月料金	24 か月料金	36 か月料金
Basic (10 ホストのモニタリング、5 コンテナのモニタリング)	100 USD	1,000 USD	2,000 USD	4000 USD
Standard (20 ホストのモニタリング、10 コンテナのモニタリング)	200 USD	2,000 USD	4000 USD	8,000 USD
Pro (40 ホストのモニタリング、20 コンテナのモニタリング)	400 USD	4000 USD	8,000 USD	16,000 USD
1 時間あたりのモニタリング対象追加ホスト	10 USD	100 USD	\$200	400 USD
1 時間あたりのモニタリング対象追加コンテナ	10 USD	100 USD	\$200	400 USD

Note

価格は、1 か月、12 か月、24 か月、36 か月の期間に対して定義できます。製品にこれらのオプションを1 つ以上提供することができます。期間は各ディメンション間で同じである必要があります。

Example

例えば、ReadOnlyUsersと AdminUsersディメンションがある場合、に年間料金を提供する場合は ReadOnlyUsers、 AdminUsersにも年間料金を提供する必要があります。

自動更新

顧客がコンテナ契約 AWS Marketplace を使用して を通じて製品を購入すると、契約条件を自動的に更新することに同意できます。顧客は1年、2年、または3年の期間中に、使用権限に対して毎月支払い続けます。

お客様はいつでも更新の設定を変更できます。詳細については、「AWS Marketplace 購入者ガイド」の「[既存の契約の変更](#)」を参照してください。

コンテナ契約が終了したとき

コンテナ契約製品には、契約の有効期限があります。契約が終了すると、次のイベントが発生します。

1. コンテナ製品は、購入者の使用権限が変更されたことを示す entitlement-updated 通知を受け取り、は空のレスポンス AWS Marketplace Entitlement Service を返します。
2. その顧客の残り使用量を計測するまでに1時間の猶予があります。このステップの後、この顧客の計測レコードを送信することはできません。

コンテナ製品の請求、計測、ライセンスの統合

AWS Marketplace は他の と統合 AWS のサービスして、コンテナ製品の計測と契約ベースの料金の両方を提供します。

AWS Marketplace Metering Serviceによる時間単位計測およびカスタム計測

製品を使用する使用権限および請求用使用計測の権限の両方を確認するには、[AWS Marketplace Metering Service](#) を使用します。独自の料金単位を定義し、その使用量を請求用に計測する場合は、[MeterUsage](#) API オペレーションを使用して統合します。使用したタスクまたはポッドの数に基づいて製品の料金を設定し、その使用量を自動的に AWS 計測する場合は、[RegisterUsage](#) API オペレーションを使用して統合します。どちらのタイプの料金設定でも、AWS Marketplace Metering Serviceとの統合を変更する必要なしで長期契約料金を追加できます。

で新しいコンテナ製品を作成すると AWS Marketplace 管理ポータル、製品を と統合するために使用される一連の製品識別子 (製品コードとパブリックキー) が提供されます AWS Marketplace Metering Service。

エンタイトルメント

と統合 AWS Marketplace Metering Service することで、有料ソフトウェアを実行している顧客が製品をサブスクライブしていることを検証し AWS Marketplace、コンテナの起動時の不正使用を防ぐことができます。使用権限を確認するには、料金モデルに応じて、[MeterUsage](#) または [RegisterUsage](#) API オペレーションを使用します。時間単位および固定月単位の料金モデルの場合は、[RegisterUsage](#) API オペレーションを使用します。カスタム計量料金モデルの場合は、[MeterUsage](#) API オペレーションを使用します。

購入者に商品の購入資格がない場合、これらの API オペレーションは `CustomerNotEntitledException` 例外を返します。

Note

購入者が製品を実行中に登録を解除した場合、購入者にはその製品を実行し続ける権利が与えられます。ただし、製品の追加のコンテナを起動することはできません。

統合ガイドライン

コンテナ製品を作成して公開し、使用権限と計測に [MeterUsage](#) API オペレーションあるいは [RegisterUsage](#) API オペレーションを使用する際は、以下のガイドラインに留意してください。

- ソフトウェアまたは Docker コンテナイメージ内で AWS 認証情報を設定しないでください。購入者の AWS 認証情報は、コンテナイメージが Amazon ECS タスクまたは Amazon EKS ポッド内で実行されているときに、実行時に自動的に取得されます。
- Amazon EKS から MeterUsage または RegisterUsage API オペレーションを呼び出すには、[サポートされている AWS SDK を使用する必要があります](#)。Amazon EKS の MeterUsage あるいは RegisterUsage 統合をテストするには、Kubernetes 1.13.x 以降が搭載されている Amazon EKS クラスターを実行する必要があります。ポッドサポートの AWS Identity and Access Management (IAM) ロールには Kubernetes 1.13 が必要です。実行中のポッドが Amazon EKS でこれらのアクションを呼び出すために必要な AWS 認証情報を取得するには、IAM ロールが必要です。
- ローカルな開発を実行できますが、PlatformNotSupportedException 例外が発生します。この例外は、コンテナサービス (Amazon ECS、Amazon EKS、Fargate) で AWS コンテナを起動しても発生しません。

サポート対象 AWS リージョン

AWS Marketplace サポートされているすべての のリストについては AWS リージョン、グローバルインフラストラクチャウェブサイトの「[リージョン表](#)」を参照してください。

計測 AWS リージョン 用の の取得

計測用のコンテナを MeterUsage または RegisterUsage API オペレーションと統合する場合、特定の を使用するよう AWS SDK を設定しないでください AWS リージョン。リージョンは実行時に動的に取得する必要があります。

Example

例えば、顧客が Amazon ECS タスクまたは Amazon EKS ポッドを起動したとします。RegisterUsage API オペレーションは、Amazon ECS タスクまたは Amazon EKS ポッドが起動されたリージョンとは異なるリージョンで呼び出されます。そのため、RegisterUsage API オペレーションでは InvalidRegionException エラーが発生します。

AWS SDK 言語は、一貫した方法で AWS_REGION を決定しません。SDK が自動的に AWS_REGION に対応しない場合は、ソフトウェアを手動で作成して AWS_Region を判断する必要があります。例えば、環境変数やその他の設定が存在しない場合、AWS SDK for Java は自動的に [Amazon EC2 インスタンスメタデータ](#) (具体的には ec2InstanceMetadata) を使用してリージョンを取得します。このインスタンスでは、AWS_REGION 環境変数が存在しない場合にのみ ec2InstanceMetadata を呼び出します。

ランタイム AWS リージョン に を動的に取得する方法については、プログラミング言語の [AWS SDK デベロッパーガイド](#)を参照してください。

計測変更の防止

RegisterUsage または MeterUsage への呼び出しを変更あるいは上書きする手段を購入者に提供すると、望ましくない請求および支払い問題が発生することがあります。計測および使用権限ロジックを統合することを強くお勧めします。

計測の変更を防ぐように製品を構築する場合は、次の点に留意してください。

- 購入者が CMD または ENTRYPOINT 説明を含む新しいイメージレイヤーを挿入できる場合、コンテナイメージを介して購入者が実行しているソフトウェアに RegisterUsage または MeterUsage を直接統合します。そうしないと、ベースイメージからの CMD または ENTRYPOINT を介した RegisterUsage または MeterUsage への呼び出しが購入者によって上書きされる可能性が高まります。
- ソフトウェアが入力として使用する AWS Marketplace 製品コード、RegisterUsage または購入者が変更できない MeterUsage 方法で管理することをお勧めします。ただし、Helm チャート、Kubernetes AWS CloudFormation マニフェストなど、顧客が上書きできる方法で製品が製品コードを管理する場合は、信頼できる AWS Marketplace 製品コードのリストを維持する必要があります。これは、ソフトウェアが入力として RegisterUsage または MeterUsage に渡される製品コードが有効であることを確認するためです。
- 信頼される製品コードのいずれかが無料製品用の場合、その製品コードが有料製品のコードの代わりに使用できないように確認します。

との契約料金 AWS License Manager

契約料金のコンテナベースの製品の場合、を使用してライセンス AWS License Manager を製品に関連付けます。

AWS License Manager は、お客様が購入したライセンス (エンタイトルメントとも呼ばれます) をアプリケーションが追跡および更新できるようにするライセンス管理ツールです。このセクションでは、製品についての情報を提供します。このセクションでは、製品を AWS License Manager と統合する方法について説明します。統合が完了したら、AWS Marketplace に製品リストを公開できます。

の詳細については AWS License Manager、[AWS License Manager 「ユーザーガイド」](#) および AWS CLI 「コマンドリファレンス」の [AWS License Manager 「」](#) セクションを参照してください。

Note

- 契約の有効期限を過ぎると、顧客はコンテナの新しいインスタンスを起動できなくなります。ただし、契約期間中は、インスタンスをいくつでも起動できます。これらのライセンスは特定のノードやインスタンスには適用されません。AWS 認証情報が割り当てられている限り、どのノードのどのコンテナで実行されているソフトウェアでもライセンスをチェックアウトできます。
- [プライベートオファーの作成] - 販売者は、AWS Marketplace 管理ポータルプライベートオファー作成ツールを使用して、製品のプライベートオファーを作成できます。
- [レポート] - AWS Marketplace 管理ポータルの [レポート] セクションで Amazon S3 バケットを設定して、データフィードを設定できます。詳細については、「[出品者レポート、データフィード、ダッシュボード](#)」を参照してください。

ワークフロー統合

以下のステップは、コンテナ製品を AWS License Manager と統合するワークフローを示しています。

1. 販売者は、AWS License Manager 統合を使用して製品を作成します。
2. 販売者は に製品を一覧表示します AWS Marketplace。
3. 購入者は で製品を見つけ AWS Marketplace で購入します。
4. ライセンスは、AWS アカウントの購入者に送付されます。
5. 購入者は Amazon EC2 インスタンス、Amazon ECS タスク、Amazon EKS ポッドソフトウェアを起動してソフトウェアを使用します。顧客は IAM ロールを使用してデプロイします。
6. ソフトウェアは購入者の AWS License Manager アカウントでライセンスを読み取り、購入した使用権限を検出し、それに応じて機能をプロビジョニングします。

Note

License Manager は追跡や更新を行いません。これは販売者のアプリケーションによって行われます。

AWS Marketplace Metering Service による時間単位の計測

コンテナ製品でカスタム従量制料金ディメンションの代わりに時間単位、タスク単位またはポッド単位の料金を使用する場合、カスタム計測ディメンションを定義する必要はありません。

RegisterUsage API オペレーションは、Amazon Elastic Container Service (Amazon ECS) タスクまたは Amazon Elastic Kubernetes Service (Amazon EKS) ポッドごとに、1 時間あたりのソフトウェア使用量を測定します。使用量は、秒単位で比例配分されます。存続期間の短いタスクやポッドには、最低 1 分の使用量が適用されます。ソフトウェア使用のための継続的な計測は、によって自動的に処理されます AWS Marketplace Metering Control Plane。ソフトウェアでは、開始に使用するソフトウェアの計測のために RegisterUsage を 1 回呼び出すこと以外に、計測に固有のアクションを実行する必要はありません。

RegisterUsage はコンテナを起動したらすぐに呼び出す必要があります。コンテナの起動から最初の 6 時間以内にコンテナを登録しない場合、AWS Marketplace Metering Service は前月分の計測保証を提供しません。ただし、計測は当月以降、コンテナが終了するまで継続されます。

は AWS Marketplace Metering Control Plane、顧客のサブスクリプション状態に関係なく、Amazon ECS タスクと Amazon EKS ポッドの実行に対して引き続き顧客に請求します。これにより、タスクまたはポッドが最初に正常に起動した後に、ソフトウェアが使用権限チェックを行う必要がなくなります。

時間単位の計測の前提条件

製品を公開する前に、次を行う必要があります。

1. で新しいコンテナ製品を作成し AWS Marketplace 管理ポータル、その製品コードを書き留めます。

詳細については、「[コンテナ製品の作成](#)」を参照してください。

2. 必要な時間単位の料金情報を製品積載フォーム (PLF) に記入し、処理のために当社に返送してください。

詳細については、「[コンテナ製品 \(レガシー\) の料金詳細の作成または更新](#)」を参照してください。

3. を呼び出すために必要な IAM アクセス許可を持つアプリケーションを実行しているタスクまたはポッドに AWS Identity and Access Management (IAM) ロールを使用します RegisterUsage。IAM マネージドポリシー `AWSMarketplaceMeteringRegisterUsage` には、これらのアクセス許可があります。

4. (オプション) ログ記録を表示する場合は、タスクまたはポッド定義で AWS CloudTrail ログ記録を有効にすることをお勧めします。
5. 定義したすべての料金ディメンションのレコードを使用して、RegisterUsage API オペレーションをテストコールします。

時間単位の計測用の製品ロードフォーム

時間単位の計測のために製品積載フォームに入力する際、製品を定義するその他の必須フィールドとオプションのフィールドに加えて、製品に関する次のフィールドに入力します。

- [時間料金] - 製品の 1 時間あたりの料金。
- [ディメンション長期レート] - 購入者が前払いした場合の、長期契約におけるソフトウェア料金の合計です。
- [長期期間 (日数)] - 長期契約の期間 (日単位)。

RegisterUsage の統合とプレビューモードのテスト

RegisterUsage API オペレーションを使用して、イメージを公開 AWS Marketplace のために送信する前に統合をテストします。

プレビューモードは稼働モードと同じように動作しますが、プレビューモードでは製品を使用する権限を確認しません。プレビューモードで RegisterUsage を呼び出すには、Amazon ECS または Amazon EKS で製品を実行して、コンテナイメージから RegisterUsage を呼び出します。AWS アカウントで製品を一覧表示するために使用しているを使用します AWS Marketplace。計測統合では、ハードコーディングするのではなく AWS リージョン、を動的に設定する必要があります。ただし、テストするときは、米国東部 (バージニア北部) リージョンで、有料コンテナを含む Amazon ECS タスクまたは Amazon EKS ポッドを少なくとも 1 つ起動してください。これにより、AWS Marketplace 運用チームはそのリージョンのログで作業を確認できます。

Note

製品が Amazon ECS と Amazon EKS の両方をサポートしている場合は、Amazon EKS を起動するだけで、統合が検証されます。

製品および必要なすべてのメタデータと料金情報が公開されるまでは、統合を完全にテストすることはできません。リクエストされた場合、AWS Marketplace カタログオペレーションチームは、レビューモードで計測レコードの受信を確認できます。

RegisterUsage でのエラー処理

コンテナイメージがと統合 AWS Marketplace Metering Service され、コンテナの起動 `ThrottlingException` 時以外の例外が発生した場合は、不正使用を防ぐためにコンテナを終了する必要があります。

`ThrottlingException` 以外の例外は、`RegisterUsage` API オペレーションへの最初の呼び出しでのみスローされます。同じ Amazon ECS タスクまたは Amazon EKS ポッドの以降の呼び出しでは、タスクまたはポッドの実行中に顧客がサブスクリプションを解除した場合でも、`CustomerNotSubscribedException` はスローされません。この顧客は、サブスクリプションを解除した後で使用状況が追跡されると、実行中のコンテナに対して課金されます。

次の表は、`RegisterUsage` API オペレーションがスローする可能性があるエラーを説明しています。各 AWS SDK プログラミング言語には、追加情報を参照できる一連のエラー処理ガイドラインがあります。

エラー	説明
<code>InternalServiceErrorException</code>	<code>RegisterUsage</code> は使用できません。
<code>CustomerNotEntitledException</code>	製品の有効なサブスクリプションが顧客にありません。
<code>InvalidProductCodeException</code>	リクエストの一部として渡された <code>ProductCode</code> 値が存在しません。
<code>InvalidPublicKeyException</code>	リクエストの一部として渡された <code>PublicKeyVersion</code> 値が存在しません。
<code>PlatformNotSupportedException</code>	AWS Marketplace は、基盤となるプラットフォームからの計測使用量をサポートしていません。Amazon ECS、Amazon EKS、およびのみがサポート AWS Fargate されています。

エラー	説明
ThrottlingException	RegisterUsage の呼び出しが抑制されます。
InvalidRegionException	RegisterUsage は、Amazon ECS タスクまたは Amazon EKS ポッドが起動された AWS リージョン のと同じ で呼び出される必要があります。これにより、RegisterUsage を呼び出す際にコンテナがリージョン (withRegion("us-east-1") など) を選択することを防ぎます。

を使用してコンテナ製品を AWS Marketplace Metering Service と統合する AWS SDK for Java

次の手順では、を使用して [AWS Marketplace Metering Service](#) の RegisterUsage アクションと統合 AWS SDK for Java する実装例の概要を示します。完全なソースコードについては、「[RegisterUsage Java の例](#)」を参照してください。これらの手順の多くは、言語にかかわらず適用されます。

AWS Marketplace Metering Service 統合の手順の例

1. [AWS Marketplace 管理ポータル](#) にサインインします。
2. [Assets (アセット)] から [コンテナ] を選択して、新しいコンテナ製品の作成を開始します。製品を作成すると、製品とコンテナイメージを統合するための製品コードが生成されます。公開の詳細については、「[コンテナ製品の公開 \(レガシー\)](#)」を参照してください。IAM アクセス許可の設定については、「[the section called “AWS Marketplace 計測と使用権限 API のアクセス許可”](#)」を参照してください。
3. 公開 [AWS Java SDK](#) をダウンロードします。

Important

Amazon EKS から計測 APIs を呼び出すには、[サポートされている AWS SDK を使用し](#)、Kubernetes 1.13 以降を実行している Amazon EKS クラスターで [を実行する必要があります](#)。

4. (オプション) RegisterUsageアクションと統合していて、デジタル署名検証を実行する場合は、アプリケーションクラスパスで [BouncyCastle](#) 署名検証ライブラリを設定する必要があります。

JSON ウェブトークン (JWT) を使用する場合は、アプリケーションのクラスパスに [JWT Java](#) ライブラリも含める必要があります。JWT を使用すると、署名検証のアプローチが簡素化されますが、必須ではなく、BouncyCastle 代わりにスタンドアロンを使用できます。JWT とのどちらを使用する場合でも BouncyCastle、Maven などのビルドシステムを使用して、アプリケーションクラスパスに BouncyCastle または JWT の推移的な依存関係を含める必要があります。

```
// Required for signature verification using code sample
<dependency>
  <groupId>org.bouncycastle</groupId>
  <artifactId>bcpkix-jdk15on</artifactId>
  <version>1.60</version>
</dependency>

// This one is only required for JWT
<dependency>
  <groupId>com.nimbusds</groupId>
  <artifactId>nimbus-jose-jwt</artifactId>
  <version>6.0</version>
</dependency>
```

5. 製品提供の各有料コンテナイメージから RegisterUsage を呼び出します。ProductCode と PublicKeyVersion は必須パラメータとなり、その他の入力はすべてオプションです。RegisterUsage のペイロード例を次に示します。

```
{
  "ProductCode" : "string", // (required)
  "PublicKeyVersion": 1,    // (required)
  "Nonce": "string",       // (optional) to scope down the registration
                           // to a specific running software
                           // instance and guard against
                           // replay attacks
}
```

Note

AWS Marketplace 計測サービスへの接続に一時的な問題が発生する可能性があります。AWS Marketplace では、短期的な停止やネットワークの問題を避けるため、指数関数的なバックオフを含めて最大 30 分間の再試行を実施することを強くお勧めします。

6. RegisterUsage は、リクエストの信頼性の検証に使用できる SHA-256 を使用して、RSA-PSS デジタル署名を生成します。署名には、ProductCode、PublicKeyVersion、Nonce のフィールドが含まれています。デジタル署名を検証するには、リクエストからこれらのフィールドを保持する必要があります。次のコードは、RegisterUsage 呼び出しに対するレスポンスの例です。

```
{
  "Signature": "<<JWT Token>>"
}

// Where the JWT Token is composed of 3 dot-separated,
// base-64 URL Encoded sections.
// e.g. eyJhbGcVCj9.eyJzdWIMzkwMjJ9.rr09Qw0SXRWTe

// Section 1: Header/Algorithm
{
  "alg": "PS256",
  "typ": "JWT"
}

// Section 2: Payload
{
  "ProductCode" : "string",
  "PublicKeyVersion": 1,
  "Nonce": "string",
  "iat": date // JWT issued at claim
}

// Section 3: RSA-PSS SHA256 signature
"rr09Q4FEi3gweH3X4lrt2okf5zwIatUUwERlw016wTy_21Nv8S..."
```

7. RegisterUsage 呼び出しを含むコンテナイメージを再構築し、コンテナにタグ付けして、Amazon ECR または Amazon ECR Public など、Amazon ECS または Amazon EKS と互換性がある任意のコンテナレジストリにプッシュします。Amazon ECR を使用している場合

は、Amazon ECS タスクまたは Amazon EKS ポッドを起動するアカウントに、Amazon ECR リポジトリへのアクセス許可があることを確認します。そうしないと、起動は失敗します。

- 次のコードで定義されているように、コンテナが RegisterUsage を呼び出すアクセス許可を付与する [IAM](#) ロールを作成します。Amazon ECS タスクまたは Amazon EKS ポッド定義の [タスクロール](#) パラメータでこの IAM ロールを指定する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:RegisterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

- と統合されたコンテナを参照 AWS Marketplace し、ステップ 7 で作成した IAM ロールを参照する Amazon ECS タスクまたは Amazon EKS ポッド定義を作成します。AWS CloudTrail ログ記録を表示するには、タスク定義でログ記録を有効にする必要があります。
- Amazon ECS または Amazon EKS クラスターを作成して、タスクまたはポッドを実行します。Amazon ECS クラスターの作成の詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「[クラスターの作成](#)」を参照してください。Amazon EKS クラスターの作成 (Kubernetes バージョン 1.1.3.x 以降を使用) の詳細については、「[Amazon EKS クラスターの作成](#)」を参照してください。
- Amazon ECS または Amazon EKS クラスターを設定し、us-east-1 で作成した Amazon ECS タスク定義または Amazon EKS ポッドを起動します AWS リージョン。製品が本番稼働になる前のこのテストプロセス中にのみ、このリージョンを使用する必要があります。
- RegisterUsage から有効なレスポンスを取得すると、コンテナ製品の作成を開始することができます。ご質問がある場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

RegisterUsage Java の例

次の例では、AWS SDK for Java および AWS Marketplace Metering Service を使用して RegisterUsage オペレーションを呼び出します。署名の検証はオプションですが、署名の検証を実

行する場合は、必要なデジタル署名検証ライブラリを含める必要があります。この例は、例示のみを目的としています。

```
import com.amazonaws.auth.PEM;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMetering;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMeteringClientBuilder;
import com.amazonaws.services.marketplacemetering.model.RegisterUsageRequest;
import com.amazonaws.services.marketplacemetering.model.RegisterUsageResult;
import com.amazonaws.util.json.Jackson;
import com.fasterxml.jackson.databind.JsonNode;
import com.nimbusds.jose.JWSObject;
import com.nimbusds.jose.JWSVerifier;
import com.nimbusds.jose.crypto.RSASSAVerifier;
import java.io.ByteArrayInputStream;
import java.nio.charset.StandardCharsets;
import java.security.PublicKey;
import java.security.Security;
import java.security.Signature;
import java.security.interfaces.RSAPublicKey;
import java.util.Base64;
import java.util.Optional;
import java.util.UUID;
import org.bouncycastle.jce.provider.BouncyCastleProvider;

/**
 * Class for making calls out to AWS Marketplace Metering Service.
 */
class RegisterUsage {

    private static final String PRODUCT_CODE = ".....";

    private final AWSMarketplaceMetering registerUsageClient;
    private final SignatureVerifier signatureVerifier;
    private final int publicKeyVersion;

    public RegisterUsage(final SignatureVerifier signatureVerifier) {
        this.signatureVerifier = signatureVerifier;
        this.publicKeyVersion = PublicKeyProvider.PUBLIC_KEY_VERSION;
        this.registerUsageClient =
AWSMarketplaceMeteringClientBuilder.standard().build();
    }

    /**
```

```
* Shows how to call RegisterUsage client and verify digital signature.
*/
public void callRegisterUsage() {
    RegisterUsageRequest request = new RegisterUsageRequest()
        .withProductCode(PRODUCT_CODE)
        .withPublicKeyVersion(publicKeyVersion)
        .withNonce(UUID.randomUUID().toString());

    // Execute call to RegisterUsage (only need to call once at container startup)
    RegisterUsageResult result = this.registerUsageClient.registerUsage(request);

    // Verify Digital Signature w/o JWT
    boolean isSignatureValid = this.signatureVerifier.verify(request, result);
    if (!isSignatureValid) {
        throw new RuntimeException("Revoke entitlement, digital signature
invalid.");
    }
}

/**
 * Signature verification class with both a JWT-library based verification
 * and a non-library based implementation.
 */
class SignatureVerifier {
    private static BouncyCastleProvider BC = new BouncyCastleProvider();

    private static final String SIGNATURE_ALGORITHM = "SHA256withRSA/PSS";

    private final PublicKey publicKey;

    public SignatureVerifier(PublicKeyProvider publicKeyProvider) {
        this.publicKey = publicKeyProvider.getPublicKey().orElse(null);
        Security.addProvider(BC);
    }

    /**
     * Example signature verification using the NimbusJOSEJWT library to verify the JWT
     Token.
     *
     * @param request RegisterUsage Request.
     * @param result RegisterUsage Result.
     * @return true if the token matches.
     */
}
```

```
public boolean verifyUsingNimbusJOSEJWT(final RegisterUsageRequest request, final
RegisterUsageResult result) {
    if (!getPublicKey().isPresent()) {
        return false;
    }

    try {
        JWSVerifier verifier = new RSASSAVerifier((RSAPublicKey)
getPublicKey().get());
        JWSObject jwsObject = JWSObject.parse(result.getSignature());
        return jwsObject.verify(verifier) &&
validatePayload(jwsObject.getPayload().toString(), request, result);
    } catch (Exception e) {
        // log error
        return false;
    }
}

/**
 * Example signature verification without any JWT library support.
 *
 * @param request RegisterUsage Request.
 * @param result RegisterUsage Result.
 * @return true if the token matches.
 */
public boolean verify(final RegisterUsageRequest request, final RegisterUsageResult
result) {
    if (!getPublicKey().isPresent()) {
        return false;
    }
    try {
        String[] jwtParts = result.getSignature().split("\\.");
        String header = jwtParts[0];
        String payload = jwtParts[1];
        String payloadSignature = jwtParts[2];

        Signature signature = Signature.getInstance(SIGNATURE_ALGORITHM, BC);
        signature.initVerify(getPublicKey().get());
        signature.update(String.format("%s.%s", header,
payload).getBytes(StandardCharsets.UTF_8));
        boolean verified = signature.verify(Base64.getUrlDecoder()
                .decode(payloadSignature.getBytes(StandardCharsets.UTF_8)));

        String decodedPayload = new String(Base64.getUrlDecoder().decode(payload));
    }
}
```

```
        return verified && validatePayload(decodedPayload, request, result);
    } catch (Exception e) {
        // log error
        return false;
    }
}

/**
 * Validate each value in the returned payload matches values originally
 * supplied in the request to RegisterUsage. TimeToLiveInMillis and
 * PublicKeyExpirationTimestamp will have the values in the payload compared
 * to values in the signature
 */
private boolean validatePayload(final String payload, final RegisterUsageRequest
request,
                                final RegisterUsageResult result) {
    try {
        JsonNode payloadJson = Jackson.getObjectMapper().readTree(payload);
        boolean matches = payloadJson.get("productCode")
            .asText()
            .equals(request.getProductCode());
        matches = matches && payloadJson.get("nonce")
            .asText()
            .equals(request.getNonce());
        return matches = matches && payloadJson.get("publicKeyVersion")
            .asText()
            .equals(String.valueOf(request.getPublicKeyVersion()));
    } catch (Exception ex) {
        // log error
        return false;
    }
}

private Optional<PublicKey> getPublicKey() {
    return Optional.ofNullable(this.publicKey);
}
}

/**
 * Public key provider taking advantage of the AWS PEM Utility.
 */
class PublicKeyProvider {
    // Replace with your public key. Ensure there are new-lines ("\n") in the
```

```
// string after "-----BEGIN PUBLIC KEY-----\n" and before "\n-----END PUBLIC
KEY-----".
private static final String PUBLIC_KEY =
    "-----BEGIN PUBLIC KEY-----\n"
    + "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDdlatRjRjogo3WojgGHFHYLugd
\n"
    + "UWAY9iR3fy4arWNA1KoS8kVw33cJibXr8bvWUAUparCwlvdbH6dvE0fou0/gCFQs
\n"
    + "HUfQrSDv+MuSUMAe8jzKE4qW+jK+xQU9a03GUnKHkk1e+Q0pX/g6jXZ7r1/xAK5D
\n"
    + "o2kQ+X5xK9cipRgEKwIDAQAB\n"
    + "-----END PUBLIC KEY-----";

public static final int PUBLIC_KEY_VERSION = 1;

public Optional<PublicKey> getPublicKey() {
    try {
        return Optional.of(PEM.readPublicKey(new ByteArrayInputStream(
            PUBLIC_KEY.getBytes(StandardCharsets.UTF_8))));
    } catch (Exception e) {
        // log error
        return Optional.empty();
    }
}
}
```

AWS Marketplace Metering Service によるコンテナ製品のカスタム計測

AWS Marketplace コンテナ製品では、製品ごとに最大 24 の異なる料金ディメンションでカスタム計測を行うことができます。各ディメンションには、長期契約料金を関連付けることができます。カスタム計測を有効にするには、コンテナ製品を AWS Marketplace Metering Service と統合してください。[MeterUsage](#) API オペレーションを使用して、請求の への使用量 AWS に対して独自の料金単位とカスタム計測を定義できます。

料金ディメンションは、製品積載フォームおよび MeterUsage オペレーションを介した 2 つの場所で定義されます。この 2 つの要素方法により、後続のオファーが一般公開される前に意図したとおりに作動することを確実にします。

カスタム計測を設定するには、使用カテゴリ、単位タイプ、料金ディメンションを選択する必要があります。

- 使用カテゴリ - 使用カテゴリは購入者が製品の内容とその使用方法を理解するために役立ちます。

- **単位タイプ** - 単位タイプは請求の測定単位を定義します。例えば、GBps または MBps で測定された帯域幅、ホスト数、MB、GB、TB で測定されたデータなど。
- **料金ディメンション** - 料金ディメンションは、個数ごとの料金を設定した機能またはサービス (ユーザー、スキャン、vCPU、デプロイされたエージェントなど) を表します。料金設定ディメンションは公開されています。ただし、公開されている製品については、引き続きプライベートオファーと Bring Your Own License (BYOL) を定義できます。計測レコードに料金設定を送信しないでください。単位の数量を計測すると、製品を作成するときに定義した料金と共にこれを使用して、購入者の請求書を計算します。

製品の料金設定が事前定義されたカテゴリまたは単位タイプに適合しない場合は、汎用 [単位] カテゴリを選択できます。次に、ディメンションの説明を使用して単位が何であるかを説明します。

オプションで、追跡するプロパティごとに使用量を配分することもできます。割り当ては購入者にはタグとして表示されます。これらのタグにより、購入者はコストをタグ値別に使用量に分けて表示できます。例えば、ユーザーごとに課金し、ユーザーに「Department」プロパティがある場合、「Department」のキーを持つタグを使用して使用量の割り当てを作成し、値ごとに1つの割り当てを行うことができます。これによってレポートする料金、サイズ、総使用量は変更されませんが、顧客は製品に適したカテゴリ別にコストを確認できます。

1 時間ごとに計測レコードを送信することをお勧めします。ただし、日単位または月単位の使用量を集計することもできます。停止が発生した場合は、購入者ソフトウェアの使用を集約し、次の時間帯に送信できます。1 時間に複数のレコードを送信することはできません。

Important

無料トライアルと前払いの使用権限は、時間単位で追跡されます。その結果、これらのレコードを別々に送信すると、購入者に過大請求が発生する可能性があります。

カスタム計測の前提条件

製品を公開する前に、次を行う必要があります。

1. 新しいコンテナ製品を作成し AWS Marketplace 管理ポータル、その製品コードを書き留めます。
2. 製品積載フォームに必要なディメンション情報を入力し、処理のために当社に返送してください。

3. を呼び出すために必要な IAM アクセス許可を持つアプリケーションを実行しているタスクまたはポッドに AWS Identity and Access Management (IAM) ロールを使用します MeterUsage。IAM マネージドポリシー AWSMarketplaceMeteringRegisterUsage には、これらのアクセス許可があります。
4. (オプション) AWS CloudTrail ログ記録を表示するには、タスクまたはポッド定義でログ記録を有効にすることをお勧めします。
5. 定義したすべての料金ディメンションのレコードを使用して、MeterUsage API オペレーションをテストコールします。

カスタム計測用の製品ロードフォーム

カスタム測定用の製品積載フォームに記入する場合、各製品には最大 24 のディメンションを設定できます。ディメンションは次のフィールドで定義されています。

- [ディメンション名] - コンテナアプリケーションが AWS Marketplace Metering Service に計測レコードを送信するときに使用される名前。この名前は、購入者が使用するディメンションを示します。この名前は請求レポートに記載されます。この名前を設定した後に変更することはできません。
- [ディメンションの説明] 購入者向けのディメンションの説明。説明は最大 70 文字です。製品が購入者に公開された後は、このフィールドを変更することはできません。
- [ディメンション料金] - 購入者が使用ごとに支払う場合のこの製品の単位あたりのソフトウェア料金。このフィールドでは小数点以下 3 桁がサポートされています。
- [ディメンション長期レート] - 購入者が前払いした場合の、長期契約におけるソフトウェア料金の合計です。
- [長期期間 (日数)] - 長期契約の期間 (日単位)。

MeterUsage の統合とプレビューモードのテスト

公開 AWS Marketplace のためにイメージを に送信する前に、MeterUsage オペレーションを使用して統合をテストします。

プレビューモードは稼働モードと同じように動作しますが、プレビューモードでは製品を使用する権限を確認しません。MeterUsage プレビューモードで を呼び出すには、Amazon Elastic Container Service (Amazon ECS) または Amazon Elastic Kubernetes Service (Amazon EKS) で製品を実行し、で製品を一覧表示するために AWS アカウント 使用している でコンテナイメージ MeterUsage が

ら を呼び出します AWS Marketplace。計測統合では、ハードコーディングではなく AWS リージョン、 を動的に設定する必要があります。ただし、テスト時には、米国東部 (バージニア北部) リージョンで有料コンテナを含む Amazon ECS タスクまたは Amazon EKS ポッドを少なくとも 1 つ起動して、AWS Marketplace 運用チームがそのリージョンのログで作業を検証できるようにします。

Note

- 製品が Amazon ECS と Amazon EKS の両方をサポートしている場合は、Amazon EKS を起動するだけで、統合が検証されます。
- 製品を一般公開する前、および新しいディメンションを追加した後に、すべてのディメンションをテストします。コンテナ製品に関連付けられた各ディメンションの計測レコードを送信しない場合、リクエストが失敗してエラーが発生します。

製品および必要なすべてのメタデータと料金情報が公開されるまでは、統合を完全にテストすることはできません。リクエストされた場合、AWS Marketplace カタログオペレーションチームは、レビューモードで計測レコードの受信を確認できます。

MeterUsage でのエラー処理

コンテナ起動時にコンテナイメージが MeterUsage オペレーションを統合し、ThrottlingException 以外の例外を受け取る場合、コンテナイメージを終了して不正使用を防止する必要があります。

ThrottlingException 以外の例外は、MeterUsage への最初の呼び出しでのみスローされます。同じ Amazon ECS タスクまたは Amazon EKS ポッド呼び出しからの以降の呼び出しでは、CustomerNotSubscribedException はスローされません。タスクまたはポッドが実行されている間に顧客がサブスクリプションを解除した場合でも同様です。この顧客は、サブスクリプションを解除したあとで使用状況が追跡されると、実行中のコンテナに対して課金されます。

[MeterUsage](#) の一般的なエラーの詳細については、AWS Marketplace Metering Service 「API リファレンス」の「」を参照してくださいMeterUsage。各 AWS SDK プログラミング言語には、追加情報を参照できる一連のエラー処理ガイドラインがあります。

ベンダー計測タグ付け (オプション)

ベンダー計測タグ付けは、独立系ソフトウェアベンダー (ISV) がソフトウェアの使用状況についてより詳細な情報を購入者に提供し、コスト配分を行うのに役立ちます。

購入者が使用したソフトウェア使用状況をタグ付けするには、さまざまな方法があります。1つの方法は、コスト配分に何を期待するかを購入者にたずねます。次に、購入者のアカウントで追跡するプロパティ全体で用量を分割できます。プロパティの例には、AccountId、Business Unit、Cost Centers、および商品に関連する他のメタデータが含まれます。これらのプロパティはタグとして購入者に公開されます。タグを使用すると、購入者は AWS 請求コンソール (<https://console.aws.amazon.com/billing/>) のタグ値別にコストを用量に分割して表示できます。ベンダー計測タグ付けでは、報告する価格、ディメンション、合計用量は変わりません。お客様は商品に適したカテゴリ別にコストを確認できます。

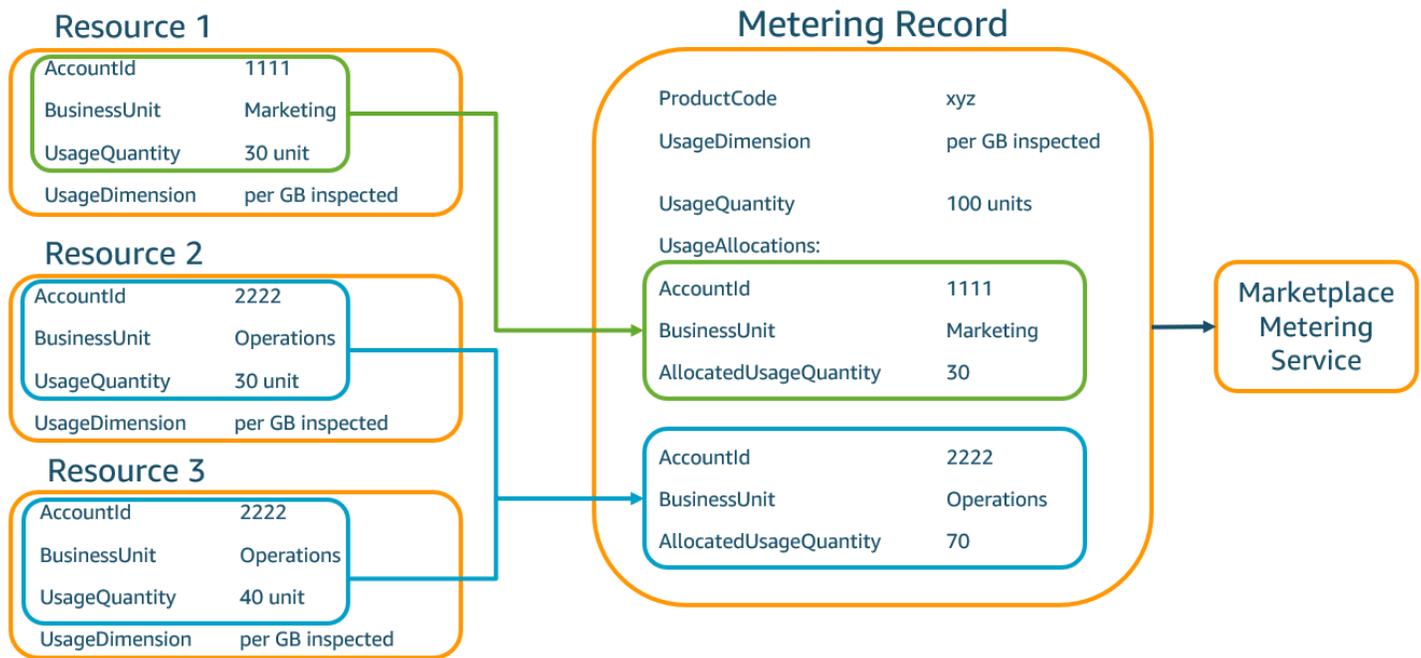
一般的なユースケースでは、購入者が1つの AWS アカウントを使って商品を購入します。また、購入者には同じ製品サブスクリプションに関連するユーザーが多数います。AccountId のキーを持つタグを使用して用量の割り当てを作成し、各ユーザーに用量を割り当てることができます。この場合、購入者は Billing and Cost Management コンソールで AccountId タグを有効にして、個々のユーザーの使用状況を分析できます。

販売者のエクスペリエンス

販売者は、すべてのリソースの用量を集計する代わりに、同じタグセットを使用してリソースの計測レコードを集計できます。例えば、販売者は UsageAllocations の異なるバケットを含む計測レコードを作成できます。各バケットは、AccountId や BusinessUnit などの、タグのセットの UsageQuantity を表します。

以下の図では、[リソース 1] には固有の AccountId と BusinessUnit タグセットがあり、[計測レコード] に1つのエントリとして表示されます。

[リソース 2] と [リソース 3] にはどちらも同じ AccountId タグ、2222、および同じ BusinessUnit タグ Operations があります。その結果、これらは [計測レコード] の1つの UsageAllocations エントリにまとめられます。



また、販売者は、タグのないリソースを割り当てられた使用量の単一の UsageAllocation にまとめて、UsageAllocations のエントリの 1 つとして送信することもできます。

制限には以下が含まれます。

- タグの数 - 5
- UsageAllocations のサイズ (カーディナリティ) - 2,500

検証には次の項目が含まれます。

- タグのキーと値に使用できる文字 - a-zA-Z0-9+ =._:\@
- UsageAllocation リスト全体の最大タグ数 - 5
- 2 つの UsageAllocations タグに同じタグ (つまり、同じタグキーと値の組み合わせ) を持つことはできません。その場合は、同じ UsageAllocation を使用する必要があります。
- UsageAllocation の AllocatedUsageQuantity の合計は、総使用量である UsageQuantity と等しくなる必要があります。

購入者エクスペリエンス

次の表は、購入者が AccountId および BusinessUnit ベンダータグを有効化した後の購入者エクスペリエンスの例を示しています。

この例では、購入者は [コスト使用状況レポート] で割り当てられた使用量を確認できます。ベンダー計測タグにはプレフィックス “aws:marketplace:isv” が使用されます。購入者は、請求情報とコスト管理の、[コスト配分タグ]、[AWS生成コスト配分タグ] でそれらを有効化できます。

[コスト使用状況レポート] の最初と最後の行は、出品者が Metering Service に送信する内容に関係します ([販売者のエクスペリエンス](#) 例を参照)。

コスト使用状況レポート (簡略版)

ProductCode	購入者	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	70	2222	オペレーショ ン
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	30	3333	財務
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	20	4444	IT
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	20	5555	マーケティング
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	30	1111	マーケティング

コード例については、[使用量割り当てタグ付きの MeterUsage コード例 \(オプション\)](#) を参照してください。

コード例

次のコード例は、コンテナ製品を製品の公開と保守に必要な AWS Marketplace APIs と統合するのに役立ちます。

使用量割り当てタグ付きの **MeterUsage** コード例 (オプション)

次のコード例は、消費料金モデルを持つコンテナ製品に関連しています。Python の例では、適切な使用量割り当てタグを含む計測レコードを に送信 AWS Marketplace して、顧客に料金を請求します pay-as-you-go 。

```
# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
#       AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited

# Import AWS Python SDK
import boto3
import time

usageRecord = [
    {
        "AllocatedUsageQuantity": 2,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "IT" },
                { "Key": "AccountId", "Value": "123456789" },
            ]
    },
    {
        "AllocatedUsageQuantity": 1,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "Finance" },
                { "Key": "AccountId", "Value": "987654321" },
            ]
    }
]
```

```
marketplaceClient = boto3.client("meteringmarketplace")

response = marketplaceClient.meter_usage(
    ProductCode="testProduct",
    Timestamp=int(time.time()),
    UsageDimension="Dimension1",
    UsageQuantity=3,
    DryRun=False,
    UsageAllocations=usageRecord
)
```

の詳細についてはMeterUsage、AWS Marketplace Metering Service 「API リファレンス[MeterUsage](#)」の「」を参照してください。

レスポンスの例

```
{ "MeteringRecordId": "string" }
```

AWS SDK for Java AWS Marketplace Metering Service を使用したコンテナ製品のとの統合

次の例では、を使用して [AWS Marketplace Metering Service](#) MeterUsageオペレーションと AWS SDK for Java 統合する実装の概要を示します。詳細については、「[MeterUsage Java の例](#)」を参照してください。次の手順の多くは、言語にかかわらず適用されます。

例：AWS Marketplace Metering Service の統合

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [アセット] から [コンテナ] を選択して、新しいコンテナ製品の作成を開始します。製品を作成すると、製品とコンテナイメージを統合するための製品コードが生成されます。公開の詳細については、「[コンテナ製品の公開 \(レガシー\)](#)」を参照してください。AWS Identity and Access Management (IAM) アクセス許可の設定については、「」を参照してください[the section called “AWS Marketplace 計測と使用権限 API のアクセス許可”](#)。
3. 公開 [AWS Java SDK](#) をダウンロードします。

⚠ Important

Amazon Elastic Kubernetes Service (Amazon EKS) から計測 API オペレーションを呼び出すには、[サポートされている AWS SDK を使用](#)し、Kubernetes 1.13 以降を実行している Amazon EKS クラスターで実行する必要があります。

4. 各ディメンションの使用状況について、1 時間に 1 回、タスクまたはポッドから MeterUsage オペレーションを呼び出します。API オペレーションは、Dimension、Resource、Hour の一意の組み合わせに対して 1 つの計測レコードを受け入れます。リソースは、Amazon Elastic Container Service (Amazon ECS) タスクまたは Amazon Elastic Container Service (Amazon ECS) ポッドのいずれかです。

```
{
  "ProductCode" : "string", // (required)
  "UsageDimension" : "string", // (required)
  "UsageQuantity": int, // (optional) Default is 0. Acceptable value from [0,
2147483647 (INT_MAX)]
  "Timestamp": Date, // (required) Timestamp in UTC. Value can be one hour in the
past.
  "UsageAllocations": List<UsageAllocation> // (optional) UsageAllocations across
1 or more tags.
}
```

ℹ Note

への接続時に一時的な問題が発生する可能性があります AWS Marketplace Metering Service。AWS Marketplace では、短期的な停止やネットワークの問題を回避するために、エクスポネンシャルバックオフで最大 30 分間の再試行を実装することを強くお勧めします。

5. MeterUsage 呼び出しを含むコンテナイメージを再構築し、コンテナにタグ付けして、Amazon Elastic Container Registry (Amazon ECR) など、Amazon ECS または Amazon EKS と互換性がある任意の Docker レジストリにプッシュします。Amazon ECR を使用している場合は、Amazon ECS タスクまたは Amazon EKS ポッドを起動するアカウントに、Amazon ECR リポジトリへのアクセス許可があることを確認します。それ以外の場合は、このオペレーションは失敗します。

6. 次のコード例で定義されているように、コンテナが MeterUsage を呼び出すアクセス許可を付与する [IAM](#) ロールを作成します。この AWS Identity and Access Management (IAM) ロールは、Amazon ECS [タスクまたは Amazon EKS ポッド定義のタスクロール](#) パラメータで指定する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:MeterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

7. と統合されたコンテナを参照 AWS Marketplace し、ステップ 6 で作成した IAM ロールを参照する Amazon ECS タスクまたは Amazon EKS ポッド定義を作成します。ログ記録を表示するには、タスク定義で AWS CloudTrail ログ記録を有効にします。
8. Amazon ECS または Amazon EKS クラスターを作成して、タスクまたはポッドを実行します。Amazon ECS クラスターの作成の詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「[クラスターの作成](#)」を参照してください。Amazon EKS クラスターの作成 (Kubernetes バージョン 1.1.3.x 以降を使用) の詳細については、「[Amazon EKS クラスターの作成](#)」を参照してください。
9. Amazon ECS または Amazon EKS クラスターを設定し、us-east-1 AWS リージョンのステップ 8 で作成した Amazon ECS タスク定義または Amazon EKS ポッドを起動します。製品が本番稼働になる前のこのテストプロセス中にも、このリージョンを使用する必要があります。
10. 製品に対して発行されるディメンションごとに MeterUsage から有効な応答が返されると、コンテナ製品の作成を開始できます。ご質問がある場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

MeterUsage Java の例

次のコード例では、AWS SDK for Java および AWS Marketplace Metering Service を使用して MeterUsage オペレーションを呼び出します。

次のコード例は、UsageAllocations なしで MeterUsage オペレーションを呼び出します。

```
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMetering;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMeteringClientBuilder;
import com.amazonaws.services.marketplacemetering.model.MeterUsageRequest;
import com.amazonaws.services.marketplacemetering.model.MeterUsageResult;

import java.util.Date;

public class MeterUsage {
    private static final String PRODUCT_CODE = ".....";
    private final AWSMarketplaceMetering awsMarketplaceMetering;

    public MeterUsage() {
        awsMarketplaceMetering =
AWSMarketplaceMeteringClientBuilder.standard().build();
    }

    /**
     * Submits metering record for a FCP Dimension. The API accepts 1 metering record
     per dimension
     * for a given buyer's resource for a given timestamp hour. Ex. If a buyer is
     running 10 tasks,
     * the API will accepts 1 call to MeterUsage in an hour for a given dimension for
     each running task.
     *
     * @param dimension - FCP dimension name provided during the publishing of the
     product.
     * @param quantity - FCP dimension consumption value for the hour.
     * @param timestamp - Timestamp, in UTC, for which the usage is being reported.
     *
     * Timestamp cant be more than 1 hour in the past.
     *
     * Make sure the timestamp value is not before the start of the
     software usage.
     */
    public void callMeterUsage(String dimension, int quantity, Date timestamp) {
        MeterUsageRequest meterUsageRequest = new MeterUsageRequest()
            .withProductCode(PRODUCT_CODE)
            .withUsageDimension(dimension)
            .withUsageQuantity(quantity)
            .withTimestamp(timestamp);
        MeterUsageResult meterUsageResult =
awsMarketplaceMetering.meterUsage(meterUsageRequest);
    }
}
```

次のコード例は、UsageAllocations で MeterUsage オペレーションを呼び出します。

```
private static String callMeterUsageWithAllocationsByTag(AWSMarketplaceMetering
marketplaceMetering) {
    // Tag Keys for the product
    String tagKey1 = "Key1";
    String tagKey2 = "Key2";
    String tagKey3 = "Key3";

    // 1st Usage Allocation bucket which has two Tags [{Key1, Key1Value1},{Key2,
Key2Value1}]
    List<Tag> tagsForUsageAllocation1 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value1"),
        new Tag().withKey(tagKey2).withValue("Key2Value1"));
    UsageAllocation usageAllocation1 = new UsageAllocation()
        .withTags(tagsForUsageAllocation1)
        .withAllocatedUsageQuantity(20);

    // 2nd Usage Allocation bucket which has two Tags [{Key1, Key1Value2},{Key2,
Key2Value1}]
    List<Tag> tagsForUsageAllocation2 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value2"),
        new Tag().withKey(tagKey2).withValue("Key2Value1"));
    UsageAllocation usageAllocation2 = new UsageAllocation()
        .withTags(tagsForUsageAllocation2)
        .withAllocatedUsageQuantity(20);

    // 3rd Usage Allocation bucket which has two Tags [{Key1, Key1Value2},{Key2,
Key2Value2},{Key3, Key3Value1}]
    List<Tag> tagsForUsageAllocation3 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value2"),
        new Tag().withKey(tagKey2).withValue("Key2Value2"),
        new Tag().withKey(tagKey3).withValue("Key3Value1"));
    UsageAllocation usageAllocation3 = new UsageAllocation()
        .withTags(tagsForUsageAllocation3)
        .withAllocatedUsageQuantity(15);

    // 4th Usage Allocation bucket with no tags
    UsageAllocation usageAllocation4 = new UsageAllocation()
        .withAllocatedUsageQuantity(15);

    List<UsageAllocation> usageAllocationList = Arrays.asList(usageAllocation1,
        usageAllocation2,
        usageAllocation3,
```

```
        usageAllocation4);

    MeterUsageRequest meterUsageRequest = new MeterUsageRequest()
        .withProductCode("TestProductCode")
        .withUsageDimension("Dimension1")
        .withTimestamp(new Date())
        //UsageQuantity value must match with sum of all
    AllocatedUsageQuantity
        .withUsageQuantity(70)
        .withUsageAllocations(usageAllocationList);

    MeterUsageResult meterUsageResult;
    try {
        meterUsageResult = marketplaceMetering.meterUsage(meterUsageRequest);
    } catch (Exception e) {
        // Log Error
        throw e;
    }

    return meterUsageResult.getMeteringRecordId();
}
```

を使用したコンテナ製品の契約料金 AWS License Manager

契約料金のコンテナベースの製品の場合は、を使用してライセンス AWS License Manager を製品に関連付けます。

AWS License Manager は、お客様が購入したライセンス (エンタイトルメントとも呼ばれます) をアプリケーションが追跡および更新できるようにするライセンス管理ツールです。このセクションでは、製品についての情報を提供します。このセクションでは、製品を AWS License Manager と統合する方法について説明します。統合が完了したら、AWS Marketplace に製品リストを公開できます。

License Manager を Amazon EKS Anywhere、Amazon ECS Anywhere、Amazon Elastic Compute Cloud (Amazon EC2)、またはオンプレミスインフラストラクチャ向けの Containers Anywhere 製品の AWS Marketplace と統合する場合は、[AWS Marketplace for Containers Anywhere 製品を License Manager と統合する](#) に記載されている手順に従ってください。

の詳細については AWS License Manager、[AWS License Manager 「ユーザーガイド」](#) および AWS CLI 「コマンドリファレンス」の [AWS License Manager 「](#)」セクションを参照してください。

ライセンスモデル

AWS Marketplace との統合では、次の 2 つのライセンスモデル AWS License Manager がサポートされています。

- [設定可能なライセンスモデル](#)
- [階層型ライセンスモデル](#)

設定可能なライセンスモデル

設定可能なライセンスモデル (定量化可能なライセンスモデルとも呼ばれます) では、購入者がライセンスを調達した後に、特定の量のリソースを購入者に付与できます。

料金設定ディメンションと単価を設定します。次に、購入者は購入するリソースの数量を選択できます。

Example 料金ディメンションと単価の

料金設定ディメンション (データバックアップなど) と単価 (1 ユニットあたり 30 USD など) を設定できます。

購入者は 5 単位、10 単位、または 20 単位の購入を選択できます。

製品は使用状況を追跡して測定し、消費されたリソースの量を測定します。

設定モデルでは、使用権限は次の 2 つのいずれかにカウントされます。

- [ドローダウンライセンス](#)
- [フローティングライセンス](#)

ドローダウンライセンス

ライセンスは、使用時に許可されたライセンス数のプールから引き出されます。その使用権限は永久にチェックアウトされ、ライセンスプールに戻すことはできません。

Example 限られた量のデータを処理する

ユーザーには 500 GB のデータを処理する権限があります。ユーザーがデータを処理し続けると、500 GB のライセンスがすべて消費されるまで 500 GB のプールからデータが引き出されます。

ドローダウンライセンスの場合は、CheckoutLicense API オペレーションを使用して、消費されたライセンスユニット (使用権限) を確認できます。

Example S3 へのバックアップの年間単位数

Amazon Simple Storage Service (Amazon S3) に、1 年間最大 1,024 単位のデータをバックアップできるストレージ製品があります。アプリケーションは、複数の Amazon EC2 インスタンスを使用して起動できます。アプリケーションには、データを追跡して集約するメカニズムがあります。ソフトウェアは、バックアップのたびに、または一定の間隔で、製品 ID を使用して CheckoutLicense API オペレーションを呼び出し、消費量を更新します。

この例では、ソフトウェアが CheckoutLicense API オペレーションを呼び出して 10 単位のデータをチェックアウトします。合計容量が顧客が購入したバックアップ制限に達すると、API コールは失敗します。

リクエスト

```
linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PERPETUAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \
--entitlements "Name=DataConsumption, Value=10, Unit=Count" \
--client-token "AKIAIOSFODNN7EXAMPLE"
```

レスポンス

```
{"CheckoutType": "PERPETUAL",
"EntitlementsAllowed": [{
  "Name": "IntermediateTier",
  "Units": "None"
}],
"Expiration": "2021-04-22T19:02:36",
"IssuedAt": "2021-04-22T18:02:36",
"LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
"LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

フローティングライセンス

ライセンスは、使用後に許可されたライセンス数のプールに返却されます。

フローティングライセンスの場合、アプリケーションはリソースが使用されているときに CheckoutLicense API オペレーションを使用して使用権限プールから使用権限をチェックアウトします。CheckoutLicense API オペレーションのレスポンスには、チェックアウト時の一意の識別子であるライセンス消費トークンが含まれます。ライセンス消費トークンを使用して、チェックアウトした使用権限をライセンスに戻したり、チェックアウトを延長したりするなど、追加のアクションを実行できます。

使用権限をプールに戻すには、リソースが使用されなくなった時に CheckInLicense API オペレーションを使用します。

```
aws license-manager check-in-license --license-consumption-token
"f1603b3c1f574b7284db84..."
```

使用権限のチェックインに失敗した場合 (アプリケーションがクラッシュした場合)、使用権限は 60 分後に自動的にプールにチェックインされます。リソースが 60 分以上使用されている場合は、リソースが使用されている限り ExtendLicenseConsumption API オペレーションを使用して使用権限をプールからチェックアウトしたままにしておくことがベストプラクティスです。

```
aws license-manager extend-license-consumption --license-consumption-token
"f1603b3c1f574b7284..."
```

Example 固定された上限を超えるユーザー数

1 人のユーザーには、アプリケーションで 500 人の同時ユーザーを割り当てることができます。ユーザーがログインしてログアウトすると、そのユーザーは引き出され、500 人のユーザープールに戻ります。ただし、500 人の同時ユーザーが固定の上限であるため、アプリケーションはプールから 500 人を超えるユーザーを引き出すことはできません。

フローティング使用権限の場合は、CheckInLicense API オペレーションを使用してライセンス単位を使用権限プールに戻すことができます。

Example 1 年間の同時ユーザー数

製品の料金は、同時接続ユーザー数に基づいて決定されます。顧客は 10 ユーザー分のライセンスを 1 年間購入します。顧客は AWS Identity and Access Management (IAM) アクセス許可を与えてソフトウェアを起動します。ユーザーがログインすると、アプリケーションが CheckoutLicense API オペレーションを呼び出して数を 1 つ減らします。ユーザーがログアウトすると、アプリケーションは CheckInLicense API オペレーションを呼び出してそのライセンスをプールに返しま

す。CheckInLicense を呼び出さない場合、ライセンス単位は 1 時間後に自動的にチェックインされます。

Note

次のリクエストでは、key-fingerprint はプレースホルダー値ではなく、すべてのライセンスが公開されるフィンガープリントの実際の値です。

リクエスト

```
aws license-manager checkout-license\  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PROVISIONAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  
--entitlements "Name=ReadOnlyUSers, Value=10, Unit=Count" \  
--client-token "AKIAIOSFODNN7EXAMPLE"
```

レスポンス

```
{  
  "CheckoutType": "PROVISIONAL",  
  "EntitlementsAllowed": [  
    {  
      "Name": "ReadOnlyUsers",  
      "Count": 10,  
      "Units": "Count",  
      "Value": "Enabled"  
    }  
  ],  
  "Expiration": "2021-04-22T19:02:36",  
  "IssuedAt": "2021-04-22T18:02:36",  
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",  
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"  
}
```

階層型ライセンスモデル

階層型ライセンスモデルでは、購入者がライセンスを調達した後に、特定のレベル (階層) のアプリケーション機能を利用できるようになります。

製品には、ベーシック、中級、プレミアムなどの階層を作成します。次に、購入者はあらかじめ定義されている階層の1つを選択します。

アプリケーションはアプリケーションの使用状況を追跡したり、測定する必要はありません。

階層型ライセンスモデルでは、使用権限はカウントされず、代わりに顧客が購入したサービスの階層が示されます。

バンドルされた機能を一緒に提供する場合は、階層化するのが望ましいです。

Example ベーシック、中級、プレミアムの各レベル

お客様は、ソフトウェアの3つのレベル(ベーシック、中級、プレミアム)のいずれかの契約に署名できます。これらの階層にはそれぞれ独自の料金設定があります。CheckoutLicense API オペレーションを呼び出し、リクエストで可能なすべての階層を指定して、ソフトウェアが顧客がサインアップした階層を特定できます。

リクエストのレスポンスには、顧客が調達した階層に対応する使用権限が含まれています。この情報に基づいて、ソフトウェアは適切なカスタマーエクスペリエンスを提供できます。

リクエスト

```
linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \
--entitlements "Name=BasicTier, Unit=None" "Name=IntermediateTier, Unit=None" \
"Name=PremiumTier, Unit=None"
```

レスポンス

```
{
  "CheckoutType": "PROVISIONAL",
  "EntitlementsAllowed": [
    {
      "Name": "IntermediateTier",
      "Units": "None"
    }
  ],
  "Expiration": "2021-04-22T19:02:36",
  "IssuedAt": "2021-04-22T18:02:36",
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b..."
```

```
"LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"  
}
```

AWS License Manager 統合の前提条件

製品を公開する前に、次を行う必要があります。

1. で新しいコンテナ製品を作成し AWS Marketplace 管理ポータル、その製品コードを書き留めます。

詳細については、「[コンテナ製品の作成](#)」を参照してください。

2. 必要な料金情報を製品積載フォーム (PLF) に記入し、処理のために当社に返送してください。

詳細については、「[コンテナ製品 \(レガシー\) の料金詳細の作成または更新](#)」を参照してください。

3. CheckoutLicense、ExtendLicenseConsumption、CheckInLicense API オペレーションを呼び出すために必要な IAM アクセス許可を持つアプリケーションを実行するタスクまたはポッドの IAM ロールを使用します。

必要な IAM アクセス許可の詳細は、次の IAM ポリシーに記載されています。

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "VisualEditor0",  
      "Effect": "Allow",  
      "Action": [  
        "license-manager:CheckoutLicense",  
        "license-manager:GetLicense",  
        "license-manager:CheckInLicense",  
        "license-manager:ExtendLicenseConsumption",  
        "license-manager:ListReceivedLicenses"  
      ],  
      "Resource": "*"   
    }   
  ]   
}
```

4. 定義したすべての料金ディメンションのレコードを使用して、RegisterUsage API オペレーションをテストコールします。

コンテナ製品と License Manager の統合

コンテナベースの製品を License Manager と統合するには

1. License Manager を呼び出すための IAM アクセス許可を設定します。詳細については、「[AWS License Manager 統合の前提条件](#)」を参照してください。
2. AWS SDK をダウンロードします。

Note

Software. AWS credentials 内で AWS 認証情報を設定しないでください。購入者の認証情報は、コンテナが Amazon EC2 インスタンス、Amazon ECS タスク、または Amazon EKS ポッド内で実行されているときに、実行時に自動的に取得されます。

3. ライセンスチェックを製品に追加します。

ライセンスチェックを実行する必要がある場所ならどこでも、製品が CheckoutLicense API オペレーションを呼び出すことができます。ライセンスを確認するには、製品が以下を認識している必要があります。

1. ライセンスの信頼できる発行者 (AWS Marketplace)
2. アプリケーションの製品 SKU (製品 ID)
3. このアプリケーションを確認する使用権限。

API コールは、設定した料金ライセンスの種類によって異なります。

4. で製品リストを公開します AWS Marketplace。

License Manager API オペレーション

顧客のライセンスマネージャーアカウントに保存されているライセンスを管理するために、ソフトウェアは次の API オペレーションを使用できます。

- GetLicense - ソフトウェアがクエリできる API。購入したライセンスのステータス (有効期限が切れた、または間もなく期限切れになる) を取得し、ステータス通知を顧客に送信します。
- CheckoutLicense - ユーザーが購入したライセンスを検出します。ユーザーがいくらかのライセンスを消費した時に、CheckoutLicense API オペレーションを使用してライセンス数を更新することもできます。CheckoutLicense を使用すると、顧客が使用したライセンス数を常に確

認することができます。顧客がすべてのライセンスを使い果たすと、この呼び出しはエラーを返します。CheckoutLicense を実行するための推奨ケイデンスについては、「[the section called “ライセンスの更新とアップグレード”](#)」を参照してください。

- ExtendLicenseConsumption - フローティングディメンションの場合、ソフトウェアがライセンスをチェックアウトすると、60 分後にライセンスは自動的にプールに戻ります。ライセンスがチェックアウトされたままの期間を延長する場合は、ExtendLicenseConsumption API オペレーションを使用してライセンスをさらに 60 分間延長します。
- CheckInLicense - フローティングディメンションの場合、ライセンスを使用権限プールに戻す場合は、CheckInLicense API オペレーションを使用してください。
- ListReceivedLicenses API - 購入者が購入したライセンスを一覧表示します。

ライセンスの更新とアップグレード

顧客は AWS Marketplace 管理ポータルでライセンスを更新またはアップグレードできます。追加購入を行うと、は新しい使用権限を反映する新しいバージョンのライセンス AWS Marketplace を生成します。ソフトウェアは同じ API オペレーションを使用して新しい使用権限を読み取ります。License Manager の統合に関しては、更新やアップグレードを処理するために特別なことをする必要はありません。

ライセンスの更新、アップグレード、キャンセルなどが発生するため、製品の使用中は定期的に CheckoutLicense API オペレーションを呼び出すことをお勧めします。CheckoutLicense API オペレーションを定期的に使用して、製品はアップグレードや有効期限などの使用権限の変更を検出できます。

CheckoutLicense API コールは 15 分ごとに実行することをお勧めします。

AWS Marketplace for Containers Anywhere 製品を License Manager と統合する

Amazon EKS Anywhere AWS Marketplace 、 Amazon ECS Anywhere 、 Amazon EC2、またはオンプレミスインフラストラクチャ用の for Containers Anywhere 製品 AWS License Manager と統合するには、次の手順に従います。 Amazon EKS Anywhere Amazon EC2

使用可能なライセンスモデルを含む AWS Marketplace、License Manager との統合に関する一般的な情報については、「」を参照してください [を使用したコンテナ製品の契約料金 AWS License Manager](#)。AWS License Managerの詳細については、「[AWS License Manager ユーザーガイド](#)」および「AWS CLI コマンドリファレンス」の「[AWS License Manager セクション](#)」を参照してください。

AWS Marketplace for Containers Anywhere 製品を License Manager と統合する

以下の手順を使用して、AWS Marketplace for Containers Anywhere 製品を と統合します AWS License Manager。

AWS Marketplace for Containers Anywhere 製品を License Manager と統合するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. 以下のステップを実行して、コンテナ製品の製品 ID を作成します。この ID をコンテナイメージで使用して、後のステップでライセンスチェックを行います。
 - a. メニューバーから [Assets (アセット)] を展開し、[コンテナ] を選択します。
 - b. 商品に購入者向けの名前を入力し、[作成] を選択します。この名前は後で変更できます。
 - c. [プロジェクト ID] を書き留めます。製品料金の詳細を作成または更新するときに使用します。

Tip

製品 ID を紛失した場合は、アセットメニューからコンテナ AWS Marketplace 管理ポータル を選択して、 で確認できます。[コンテナ] ページには、製品のリストとそれに関連する製品 ID が表示されます。

3. 最新のパブリック AWS SDK をダウンロードし、コンテナアプリケーションにインストールします。お好みの AWS SDK のインストール手順については、[「AWS で構築するツール」](#) を参照してください。

Note

Amazon EKS Anywhere または によって提供されていない Kubernetes クラスターから License Manager API オペレーションを呼び出すには AWS、サポートされている AWS SDK を使用する必要があります。サポートされている AWS SDKs [「サポートされている AWS SDK の使用」](#) を参照してください。

4. カスタム認証情報プロバイダーを使用して AWS License Manager クライアントを作成し、AWS およびオンプレミスにデプロイされたコンテナアプリケーションに認証情報を提供できるようにします。カスタム認証情報プロバイダー (LicenseCredentialProvider) の完全なソースコードについては、以下のセクションを参照してください。

- [LicenseManagerCredentialsProvider - Java 実装](#)

- [LicenseManagerCredentialsProvider - Golang 実装](#)

LicenseCredentialsProvider は、AWS を追加することで、SDK のデフォルトの認証情報プロバイダーチェーンをオンプレミス用に拡張します。LicenseManagerTokenCredentialsProvider。これにより、オンプレミス環境で License Manager OIDC が発行した ID トークンを使用して認証情報が提供されます。LicenseCredentialsProvider のソースコードをアプリケーションのクラスパスに含める必要があります。

Note

を拡張DefaultCredentialsProviderすると、同じコンテナアプリケーションが実行されているとき AWS、およびオンプレミス環境で実行されているときに認証情報を取得できるようになります。コンテナアプリケーションが既にデフォルトではなくカスタム認証情報プロバイダーチェーンを使用している場合は、LicenseManagerTokenCredentialsProvider をカスタムチェーンに追加して拡張することもできます。

次のコードスニペットは、Java を使用して AWS License Manager クライアントを作成する例です。

```
LicenseManagerClientBuilder clientBuilder =  
    LicenseManagerClient.builder().credentialsProvider(LicenseCredentialsProvider.create());
```

5. 製品オファー内の各有料コンテナイメージから `aws license-manager checkout-license` コマンドを使用して、CheckoutLicense API オペレーションを呼び出します。これにより、購入者がアプリケーションのライセンスを使用する使用権限があるかどうかを確認されます。購入者にアプリケーションに対する使用権限がある場合は、CheckoutLicense は、申請を承諾し、リクエストされた使用権限とその値を返します。購入者にアプリケーションに対する使用権限がない場合は、CheckoutLicense は、例外をスローします。

CheckoutLicense API オペレーションを呼び出す際には、以下のパラメータが必要です。

- CheckoutType - 有効な値は PROVISIONAL または PERPETUAL です。
 - チェックアウトされた使用権限の数がプールからなくなる場合に、PERPETUAL を使用します。

例: 購入者には 500 GB のデータを処理する権限があります。データの処理を続けると、500 GB のプールからその量が引き出され、使い果たされます。

- 使用権がプールからチェックアウトされ、使用後に返却されるフローティングライセンス使用権に PROVISIONAL を使用します。

例: ユーザーには、アプリケーション内で同時に 500 人のユーザーを使用する権利があります。ユーザーがログインまたはログアウトすると、そのユーザーは引き出されるか、500 人のユーザープールに戻ります。フローティングライセンス使用権限の詳細については、「[License Manager を使用したフローティングライセンス使用権限](#)」を参照してください。

- ClientToken - 大文字と小文字が区別される一意の識別子。固有のリクエストごとにランダムな UUID を使用することをお勧めします。
- Entitlements - チェックアウトする使用権限のリスト。
- 機能の使用権限の場合は、Name プロパティと Unit プロパティを次のように指定します。

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "None"
}
```

- カウント対象使用権限については、以下のように、Name プロパティ、Unit プロパティ、および Count プロパティを指定します。

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "<Entitlement_Unit>",
  "Value": <Desired_Count>
}
```

- KeyFingerprint - AWS Marketplace が発行するライセンスのキーフィンガープリントは `aws:294406891311:AWS/Marketplace:issuer-fingerprint` です。このキーフィンガープリントを使用すると、ライセンスは信頼できないエンティティによって発行 AWS Marketplace され、によって発行されないようになります。
- ProductSKU - 前のステップ AWS Marketplace 管理ポータル で生成された製品 ID。

次のスニペットは、AWS CLIを使用した CheckoutLicense API オペレーションによる呼び出しの例です。

```
aws license-manager checkout-license \  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PROVISIONAL" \  
--client-token "79464194dca9429698cc774587a603a1" \  
--entitlements "Name=AWS::Marketplace::Usage/Drawdown/DataConsumption, Value=10, Unit=Gigabytes" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

Note

ライセンスを確認するには、コンテナアプリケーションが License Manager を使用するのためのアウトバウンドネットワークアクセスが必要です。オンプレミスでデプロイされたアプリケーションでは、アウトバウンドネットワークアクセスの信頼性が低くなったり、速度が低下する可能性があります。これらのアプリケーションでは、License Manager を呼び出す際に適切に再試行する必要があります。詳細については、「[オンプレミスデプロイでの License Manager との統合に関するベストプラクティス](#)」を参照してください。

- CheckoutLicense API オペレーションを定期的呼び出して、AWS Marketplaceで行われた更新、アップグレード、またはキャンセルによって顧客のライセンスが変更されたかどうかを確認してください。ケイデンスはアプリケーションによって異なります。購入者の介入なしに変更が自動的に反映されるように、1日1回ライセンスを確認することをお勧めします。

オンプレミスにデプロイされたアプリケーションでは、定期的にライセンスをチェックするためのアウトバウンドネットワークアクセスが信頼できない場合があります。このような場合、アプリケーションは十分な耐障害性を保つためにキャッシュされたライセンスを使用する必要があります。詳細については、「[オンプレミスデプロイでの License Manager との統合に関するベストプラクティス](#)」を参照してください。

- CheckoutLicense 呼び出しをコンテナアプリケーションと統合したら、変更を加えた新しいバージョンの Docker コンテナイメージを構築します。
- アプリケーションの Helm チャートを更新して、License Manager API を使用してライセンスにアクセスするための設定を含む Kubernetes シークレットをオプション入力として受け入れます。設定シークレットには、License Manager によって発行された ID トークンと、コンテナアプリケーションがオンプレミスにデプロイされたときに License Manager APIs を呼び出すための AWS 認証情報を取得するために前述のカスタム認証情報プロバイダーによって使用される AWS Identity and Access Management ロールが含まれます。また、デフォルト値 us-east-1 を使用して AWS リージョン を入力として追加します。

コンテナアプリケーションをオンプレミスにデプロイする購入者は、コンテナ製品の AWS Marketplace 購入者エクスペリエンスを通じて Kubernetes シークレットを作成できます。Kubernetes シークレット名を `helm install` コマンドへの入力として指定します。設定シークレットは次の形式で構成されます。

```
apiVersion: v1
kind: Secret
metadata:
  name: aws-marketplace-license-config
type: Opaque
stringData:
  license_token: <token_value> // License Manager issued JWT token
  iam_role: <role_arn> // AWS Identity and Access Management role to assume with
  license token
```

9. と統合されたコンテナイメージの Helm チャートのアプリケーションデプロイテンプレートを更新 AWS License Manager して、以下を含めます。

- ポッドのサービスアカウント - Amazon EKS に Helm をデプロイするにはサービスアカウントが必要です。コンテナイメージのサービスアカウントの IAM ロールを設定して、License Manager API オペレーションを呼び出す権限を取得するために使用されます。サービスアカウントの IAM ロールの詳細については、「[サービスアカウントの IAM ロール](#)」を参照してください。
- オンプレミスデプロイのライセンスアクセス - ライセンス設定シークレットは、オンプレミス環境での Helm デプロイの License Manager API オペレーションを呼び出すための認証情報と適切なアクセス許可を提供するために必要です。購入者は、AWS Marketplace 購入者エクスペリエンスからライセンスシークレットを生成して Helm に提供します。

以下のコードスニペットは、サービスアカウント、ライセンス設定、イメージプルシークレットを含むデプロイ仕様のサンプルです。

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: example-app
spec:
  replicas: 1
  selector:
    matchLabels:
```

```
    app: example-app
  template:
    metadata:
      labels:
        app: example-app
  spec:
    // Service account for pod
    serviceAccountName: {{ .Values.serviceAccountName }}
    containers:
      - name: example-app
        image: example-app
        ports:
          - containerPort: 8001
// Add the following conditional attributes
{{ - if .Values.awsmpl.licenseConfigSecretName }}
  //Mount the license volume to the container image
  volumeMounts:
    - name: awsmpl-product-license
      mountPath: "/var/run/secrets/product-license"
  //Add following environment variable to container for credential
  provider
  env:
    - name: AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE
      value: "/var/run/secrets/product-license/license_token"
    - name: AWS_ROLE_ARN
      valueFrom:
        secretKeyRef:
          name: {{ .Values.aws.licenseConfigSecretName }}
          key: iam_role
  //Mount the license secret as a volume to the pod
  volumes:
    - name: awsmpl-product-license
      secret:
        secretName: {{ .Values.aws.licenseConfigSecretName }}
        optional: true
{{ - end }}
```

Note

ライセンス設定シークレットはオプションです。購入者はこの値をオンプレミスデプロイにのみ使用します。AWS デプロイの場合、デプロイ仕様には License Manager 統合イメージのサービスアカウントを含める必要があります。

10. 以下のセクションのステップを実行して、License Manager 統合をローカルと Amazon EKS でテストします。
 - a. [License Manager 統合のローカルテスト](#)
 - b. [Amazon EKS での License Manager 統合のテスト](#)
11. License Manager の統合をオンプレミス AWS とオンプレミスの両方で正常に検証したら、「」の手順に従ってコンテナ製品リストを作成できます [コンテナ製品の作成](#)。

License Manager 統合のローカルテスト

minikube やその他のセットアップを使用して、ローカルで任意の Kubernetes クラスターで License Manager の統合をテストできます。Kubernetes クラスターに、License Manager API オペレーションを呼び出すためのアウトバウンドインターネットアクセスがあることを確認します。

License Manager 統合をローカルでテストするには

1. テスト用販売者アカウントで、希望する使用権限を持つテストライセンスを作成します。テストライセンスを設定するには、AWS License Manager API リファレンスの [CreateLicense](#) 「」を参照してください。または、次のスクリプトを使用してテストライセンスを作成し、テスト購入者アカウントにライセンスを付与してライセンスを使用します。次のスクリプトでは、テスト用販売者アカウントの認証情報を使用しています。

```
read -p 'AWS Account for test buyer: ' TEST_BUYER_ACCOUNT_ID
read -p 'License entitlements: ' ENTITLEMENTS

# TEST_SELLER_ACCOUNT_ID="109876543210"
# ENTITLEMENTS="{\"Name\": \"ByData\", \"MaxCount\": 1000, \"Overage\": true, \"Unit\": \"Gigabits\", \"AllowCheckIn\": true}"

# Create License

NOW=$(date +%Y-%m-%dT00:00:00+00:00)

PRODUCT_NAME="My awesome product"
PRODUCT_SKU="c97b7825-44c4-4f42-b025-12baa4c171e0"

LICENSE_BENEFICIARY=" arn:aws:iam::$TEST_BUYER_ACCOUNT_ID:root "
LICENSE_ISSUER_NAME="test-seller"
LICENSE_NAME="test-seller-license"
```

```

CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"
CONSUMPTION_TTL=180
CONSUMPTION_RENEW_TYPE="None"

HOME_REGION="us-east-1"

LICENSE_ARN=$(aws license-manager create-license --license-name
"$LICENSE_NAME" --product-name "$PRODUCT_NAME" --product-sku
"$PRODUCT_SKU" --issuer Name="$LICENSE_ISSUER_NAME" --home-region
"$HOME_REGION" --validity Begin="$NOW" --entitlements "$ENTITLEMENTS"
--beneficiary "$LICENSE_BENEFICIARY" --consumption-configuration
RenewType="$CONSUMPTION_RENEW_TYPE",ProvisionalConfiguration={MaxTimeToLiveInMinutes=
$CONSUMPTION_TTL} --client-token "$CLIENT_TOKEN" | jq -r ".LicenseArn" )

echo "License arn: $LICENSE_ARN"

# Create Grant

GRANT_TOKEN="e9a14140-4fca-4219-8230-57511a6ea6"
GRANT_NAME="test-grant"

GRANT_ARN=$(aws license-manager create-grant --grant-name "$GRANT_NAME"
--license-arn "$LICENSE_ARN" --principals "$LICENSE_BENEFICIARY" --home-
region "$HOME_REGION" --client-token "$GRANT_TOKEN" --allowed-operations
"CheckoutLicense" "CheckInLicense" "ExtendConsumptionLicense" "CreateToken" | jq -
r ".GrantArn")

echo "Grant arn: $GRANT_ARN"

```

2. 以前に定義したシークレット形式を使用して、ライセンストークンと IAM ロールを使用して Kubernetes シークレットを作成します。License Manager CreateToken API オペレーションを使用してライセンストークンを生成します。次に、IAM CreateRole API オペレーションを使用して、アクセス許可と信頼ポリシーを含む IAM ロールを作成します。このスクリプトの例を参照してください。次のスクリプトでは、テスト用購入者アカウントの認証情報を使用しています。

```

read -p 'AWS Account for test license: ' TEST_ACCOUNT_ID
read -p 'License Arn' LICENSE_ARN
# Create IAM Role
ROLE_NAME="AWSLicenseManagerConsumptionTestRole"
ROLE_DESCRIPTION="Role to test AWS License Manager integration on-prem"
ROLE_POLICY_ARN="arn:aws:iam::aws:policy/service-role/
AWSLicenseManagerConsumptionPolicy"

```

```

ROLE_TRUST_POLICY="{\"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect\":
\"Allow\", \"Principal\": { \"Federated\": \"openid-license-manager.amazonaws.com
\" }, \"Action\": \"sts:AssumeRoleWithWebIdentity\", \"Condition\":
{ \"ForAnyValue:StringLike\": { \"openid-license-manager.amazonaws.com:amr\":
\"aws:license-manager:token-issuer-account-id:${TEST_ACCOUNT_ID}\" } } ]}"
ROLE_SESSION_DURATION=3600

ROLE_ARN=$(aws iam create-role --role-name "$ROLE_NAME" --description
"$ROLE_DESCRIPTION" --assume-role-policy-document "$ROLE_TRUST_POLICY" --max-
session-duration $ROLE_SESSION_DURATION | jq ".Role" | jq -r ".Arn")

aws iam attach-role-policy --role-name "$ROLE_NAME" --policy-arn "$ROLE_POLICY_ARN"

echo "Role arn: $ROLE_ARN"

# Create Token
CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"

TOKEN=$(aws license-manager create-token --license-arn $LICENSE_ARN --role-arns
$ROLE_ARN --client-token $CLIENT_TOKEN | jq '.Token')

echo "License access token: $TOKEN"

```

3. の外部でホストされている Kubernetes クラスターをセットアップします AWS。これを使用して、コンテナアプリケーションが 以外の環境から AWS License Manager API に接続できること AWS、およびカスタム認証情報プロバイダーがアプリケーションに適切に統合されていることをテストします。
4. 以前に生成したライセンストークンと IAM ロールをローカル Kubernetes クラスターにデプロイします。

```

kubectl create secret generic "awsmp-license-access-config" \
--from-literal=license_token=${TOKEN} \
--from-literal=iam_role=${ROLE_ARN}

```

5. シークレット名を入力としてアプリケーションを Helm 経由でデプロイし、アプリケーションが License Manager API オペレーションを呼び出して使用権限チェックを実行できることを確認します。Helm とデプロイ仕様の変更については、[AWS Marketplace for Containers Anywhere 製品を License Manager と統合する](#) の「ステップ 9」を参照してください。

Amazon EKS での License Manager 統合のテスト

Amazon EKS で License Manager の統合をテストすることもできます。ライセンス設定シークレットがなくても、アプリケーションが License Manager API オペレーションを呼び出せることをテストして確認します。また、サービスアカウントを使用してサービスアカウント (IRSA) の IAM ロールを設定し、関連する認証情報をアプリケーションに提供することも確認してください。

Amazon EKS で License Manager 統合をテストするには

1. テスト用販売者アカウントで、希望する使用権限を持つテストライセンスを作成します。[CreateLicense 「API リファレンス」](#) を参照してテストライセンスをセットアップするか、次のスクリプトを使用してテストライセンスを作成し、テスト購入者アカウントへのライセンス付与を作成してライセンスを使用します。次のスクリプトでは、テスト用販売者アカウントの認証情報を使用しています。

```
read -p 'AWS Account for test buyer: ' TEST_BUYER_ACCOUNT_ID
read -p 'License entitlements: ' ENTITLEMENTS

# TEST_SELLER_ACCOUNT_ID="109876543210"
# ENTITLEMENTS="{\"Name\": \"ByData\", \"MaxCount\": 1000, \"Overage\": true, \"Unit\": \"Gigabits\", \"AllowCheckIn\": true}"

# Create License

NOW=$(date +%Y-%m-%dT00:00:00+00:00)

PRODUCT_NAME="My awesome product"
PRODUCT_SKU="c97b7825-44c4-4f42-b025-12baa4c171e0"

LICENSE_BENEFICIARY=" arn:aws:iam::$TEST_BUYER_ACCOUNT_ID:root "
LICENSE_ISSUER_NAME="test-seller"
LICENSE_NAME="test-seller-license"

CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"
CONSUMPTION_TTL=180
CONSUMPTION_RENEW_TYPE="None"

HOME_REGION="us-east-1"

LICENSE_ARN=$(aws license-manager create-license --license-name
"$LICENSE_NAME" --product-name "$PRODUCT_NAME" --product-sku
"$PRODUCT_SKU" --issuer Name="$LICENSE_ISSUER_NAME" --home-region
```

```
"$HOME_REGION" --validity Begin="$NOW" --entitlements "$ENTITLEMENTS"
--beneficiary "$LICENSE_BENEFICIARY" --consumption-configuration
RenewType="$CONSUMPTION_RENEW_TYPE",ProvisionalConfiguration={MaxTimeToLiveInMinutes=$
$CONSUMPTION_TTL} --client-token "$CLIENT_TOKEN" | jq -r ".LicenseArn" )

echo "License arn: $LICENSE_ARN"

# Create Grant

GRANT_TOKEN="e9a14140-4fca-4219-8230-57511a6ea6"
GRANT_NAME="test-grant"

GRANT_ARN=$(aws license-manager create-grant --grant-name "$GRANT_NAME"
--license-arn "$LICENSE_ARN" --principals "$LICENSE_BENEFICIARY" --home-
region "$HOME_REGION" --client-token "$GRANT_TOKEN" --allowed-operations
"CheckoutLicense" "CheckInLicense" "ExtendConsumptionLicense" "CreateToken" | jq -
r ".GrantArn")

echo "Grant arn: $GRANT_ARN"
```

2. 必要な設定のテスト Amazon EKS クラスターを作成するか、以下のコマンドを実行してデフォルト設定を使用します。

```
aws ec2 create-key-pair --region us-west-2 --key-name eks-key-pair
```

```
eksctl create cluster \
--name awsmvp-eks-test-example \
--region us-west-2 \
--with-oidc \
--ssh-access \
--ssh-public-key eks-key-pair
```

3. 既存のクラスターのサービスアカウントを作成し、それを IAM ロールに関連付けます。次のコマンドは、AWSLicenseManagerConsumptionPolicy を使用して IAM ロールを作成します。次に、コマンドは、License Manager 統合イメージをデプロイする必要がある Amazon EKS クラスターの test_sa サービスアカウントにアタッチします。その結果、サービスアカウントは、License Manager API オペレーションを呼び出すための適切な認証情報を取得できます。

```
eksctl create iamserviceaccount \
--name test_sa \
```

```
--namespace test_namespace \  
--cluster awsm-p-eks-test-example \  
--attach-policy-arn "arn:aws:iam::aws:policy/service-role/  
AWSLicenseManagerConsumptionPolicy" \  
--approve \  
--override-existing-serviceaccounts
```

4. Helm を通じて、前のコマンドで IAM ロールが関連付けられているサービスアカウントにアプリケーションをデプロイします。アプリケーションが License Manager API オペレーションを呼び出して使用権限チェックを実行できることを確認します。

License Manager を使用したフローティングライセンス使用権限

フローティングライセンスでは、ユーザーがアプリケーションにログインすると、使用可能なライセンスのプールからライセンスが引き出されます。ユーザーがログアウトすると、ライセンスは使用可能なライセンスのプールに再び追加されます。

フローティングライセンスの場合、アプリケーションは CheckoutLicense API オペレーションを使用して、リソースが使用されているときに使用権限プールから使用権限をチェックアウトします。CheckoutLicense API オペレーションのレスポンスには、チェックアウト時の一意の識別子であるライセンス消費トークンが含まれます。ライセンス消費トークンは、チェックアウトした使用権限をライセンスプールに戻したり、チェックアウトを延長したりするなど、追加のアクションを実行できます。

リソースが使用されなくなると、アプリケーションは CheckInLicense API オペレーションを使用して使用権限をプールにチェックインし直します。

```
aws license-manager check-in-license \  
--license-consumption-token "f1603b3c1f574b7284db84a9e771ee12"
```

オペレーション中にアプリケーションがクラッシュした場合など、ライセンスをプールにチェックインし直せなかった場合、60 分後に使用権限が自動的にプールにチェックインされます。このため、リソースが 60 分以上使用されている場合は、使用権限をプールからチェックアウトしたままにしておくことがベストプラクティスです。そのためには、リソースが使用されている限り ExtendLicenseConsumption API オペレーションを使用してください。

```
aws license-manager extend-license-consumption \  
--license-consumption-token "f1603b3c1f574b7284db84a9e771ee12"
```

オンプレミスデプロイでの License Manager との統合に関するベストプラクティス

オンプレミス環境にコンテナアプリケーションをデプロイすると、信頼性の低いアウトバウンドネットワークアクセスが発生する可能性があります。以下のベストプラクティスを活用して回復性を高め、インターネット接続の不備による潜在的な問題による購入者のサービスの中断を回避してください。

- 適切な再試行 — 一時的なネットワークの問題により、アプリケーションがに接続できなくなる可能性があります AWS License Manager。最大で 30 分間の再試行をし、指数関数的にバックオフを行います。これにより、短期的な停止やネットワークの問題を回避できます。
- ハードリミットの回避 - 接続されたクラスターにデプロイされたアプリケーションは、定期的にライセンスをチェックして、アップグレードや更新による変更を特定できます。アウトバウンドアクセスの信頼性が低いと、アプリケーションはそれらの変更を識別できない可能性があります。可能な限り、アプリケーションは、License Manager を通じてライセンスを確認できないことによる購入者へのサービスの中断を避ける必要があります。ライセンスの有効期限が切れてライセンスが有効かどうかを確認できなくなると、アプリケーションは無料試用エクスペリエンスまたはオープンソースエクスペリエンスに頼ることができます。
- 顧客への通知 - キャッシュ型ライセンスを使用する場合、ライセンスへの変更 (更新やアップグレードを含む) は、実行中のワークロードに自動的に反映されません。アプリケーションがキャッシュされたライセンスを更新できるように、アプリケーションへのアウトバウンドアクセスを一時的に再度許可する必要があることを顧客に通知します。例えば、アプリケーション自体またはドキュメントを通じて顧客に通知します。同様に、下位の機能に頼る場合は、その使用権限がなくなったか、ライセンスの有効期限が切れていることを顧客に通知します。その後、顧客はアップグレードまたは更新のどちらかを選択できます。

LicenseManagerCredentialsProvider - Java 実装

LicenseCredentialsProvider は、AWS を追加することで、SDK のデフォルトの認証情報プロバイダーチェーンをオンプレミス用に拡張します LicenseManagerTokenCredentialsProvider。

LicenseCredentialsProvider

```
package com.amazon.awssmp.license;

import software.amazon.awssdk.auth.credentials.AwsCredentials;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProviderChain;
```

```
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.auth.credentials.internal.LazyAwsCredentialsProvider;
import software.amazon.awssdk.utils.SdkAutoCloseable;

public class LicenseCredentialsProvider implements AwsCredentialsProvider,
    SdkAutoCloseable {
    private static final LicenseCredentialsProvider CREDENTIALS_PROVIDER = new
    LicenseCredentialsProvider();
    private final LazyAwsCredentialsProvider providerChain;

    private LicenseCredentialsProvider() {
        this.providerChain = createChain();
    }

    public static LicenseCredentialsProvider create() {
        return CREDENTIALS_PROVIDER;
    }

    @Override
    public AwsCredentials resolveCredentials() {
        return this.providerChain.resolveCredentials();
    }

    @Override
    public void close() {
        this.providerChain.close();
    }

    private LazyAwsCredentialsProvider createChain() {
        return LazyAwsCredentialsProvider.create(() -> {
            AwsCredentialsProvider[] credentialsProviders = new
            AwsCredentialsProvider[]{
                DefaultCredentialsProvider.create(),
                LicenseManagerTokenCredentialsProvider.create()};

            return AwsCredentialsProviderChain.builder().reuseLastProviderEnabled(true)
                .credentialsProviders(credentialsProviders).build();
        });
    }
}
```

LicenseManagerTokenCredentialsProvider

LicenseManagerTokenCredentialsProvider により、オンプレミス環境で License Manager OIDC が発行した ID トークンを使用して認証情報が提供されます。LicenseCredentialsProvider のソースコードをアプリケーションのクラスパスに含める必要があります。

```
package com.amazon.awsmp.license;

import software.amazon.awssdk.auth.credentials.AnonymousCredentialsProvider;
import software.amazon.awssdk.auth.credentials.AwsCredentials;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.core.SdkSystemSetting;
import software.amazon.awssdk.core.client.config.ClientOverrideConfiguration;
import software.amazon.awssdk.core.retry.RetryPolicyContext;
import software.amazon.awssdk.core.retry.conditions.OrRetryCondition;
import software.amazon.awssdk.core.retry.conditions.RetryCondition;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.regions.providers.DefaultAwsRegionProviderChain;
import software.amazon.awssdk.services.licensemanager.LicenseManagerClient;
import software.amazon.awssdk.services.licensemanager.model.GetAccessTokenRequest;
import software.amazon.awssdk.services.licensemanager.model.GetAccessTokenResponse;
import software.amazon.awssdk.services.sts.StsClient;
import
    software.amazon.awssdk.services.sts.auth.StsAssumeRoleWithWebIdentityCredentialsProvider;
import software.amazon.awssdk.services.sts.model.AssumeRoleWithWebIdentityRequest;
import software.amazon.awssdk.services.sts.model.IdpCommunicationErrorException;
import software.amazon.awssdk.utils.IoUtils;
import software.amazon.awssdk.utils.SdkAutoCloseable;
import software.amazon.awssdk.utils.StringUtils;
import software.amazon.awssdk.utils.SystemSetting;

import java.io.IOException;
import java.io.InputStream;
import java.io.UncheckedIOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Duration;
import java.util.function.Supplier;

public class LicenseManagerTokenCredentialsProvider implements AwsCredentialsProvider,
    SdkAutoCloseable {

    private final StsAssumeRoleWithWebIdentityCredentialsProvider credentialsProvider;
```

```
private final RuntimeException loadException;

private Path licenseAccessTokenFile;
private String roleArn;
private String roleSessionName;
private StsClient stsClient;
private LicenseManagerClient lmClient;

public static LicenseManagerTokenCredentialsProvider create() {
    return new Builder().build();
}

@Override
public AwsCredentials resolveCredentials() {
    if (this.loadException != null) {
        throw this.loadException;
    }
    return this.credentialsProvider.resolveCredentials();
}

@Override
public void close() {
    IoUtils.closeQuietly(this.credentialsProvider, null);
    IoUtils.closeQuietly(this.stsClient, null);
    IoUtils.closeIfCloseable(this.lmClient, null);
}

private LicenseManagerTokenCredentialsProvider(Builder builder) {
    StsAssumeRoleWithWebIdentityCredentialsProvider credentialsProvider = null;
    RuntimeException loadException = null;

    try {
        this.licenseAccessTokenFile =
Paths.get(StringUtils.trim(LicenseSystemSetting.AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE.getStringValue()));
        this.roleArn = SdkSystemSetting.AWS_ROLE_ARN.getStringValueOrThrow();
        this.roleSessionName =
SdkSystemSetting.AWS_ROLE_SESSION_NAME.getStringValue().orElse("aws-sdk-java-" +
System.currentTimeMillis());
        this.stsClient = builder.stsClient != null ? builder.stsClient :
StsClientFactory.create();
        this.lmClient = builder.lmClient != null ? builder.lmClient :
LicenseManagerClientFactory.create();
    }
}
```

```
        AssumeRoleWithWebIdentityRequest request =
AssumeRoleWithWebIdentityRequest.builder()

        .roleArn(this.roleArn).roleSessionName(this.roleSessionName).build();

        Supplier<AssumeRoleWithWebIdentityRequest> supplier = new
AssumeRoleRequestSupplier(request,
                this.licenseAccessTokenFile, this.lmClient);

        credentialsProvider =
StsAssumeRoleWithWebIdentityCredentialsProvider.builder()
                .stsClient(this.stsClient).refreshRequest(supplier).build();
    } catch (RuntimeException ex) {
        loadException = ex;
    }

    this.credentialsProvider = credentialsProvider;
    this.loadException = loadException;
}

public static final class Builder {
    private Path licenseAccessTokenFile;
    private String roleArn;
    private String roleSessionName;
    private StsClient stsClient;
    private LicenseManagerClient lmClient;

    public LicenseManagerTokenCredentialsProvider build() {
        return new LicenseManagerTokenCredentialsProvider(this);
    }

    public LicenseManagerTokenCredentialsProvider.Builder
licenseAccessTokenFile(Path licenseAccessTokenFile) {
        this.licenseAccessTokenFile = licenseAccessTokenFile;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder roleArn(String roleArn) {
        this.roleArn = roleArn;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder roleSessionName(String
roleSessionName) {
```

```
        this.roleSessionName = roleSessionName;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder stsClient(StsClient
stsClient) {
        this.stsClient = stsClient;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder
lmClient(LicenseManagerClient lmClient) {
        this.lmClient = lmClient;
        return this;
    }
}

private static final class AssumeRoleRequestSupplier implements Supplier {
    private final LicenseManagerClient lmClient;
    private final AssumeRoleWithWebIdentityRequest request;
    private final Path webIdentityRefreshTokenFile;

    AssumeRoleRequestSupplier(final AssumeRoleWithWebIdentityRequest request,
final Path
webIdentityRefreshTokenFile,
final LicenseManagerClient lmClient) {

        this.lmClient = lmClient;
        this.request = request;
        this.webIdentityRefreshTokenFile = webIdentityRefreshTokenFile;
    }

    public AssumeRoleWithWebIdentityRequest get() {
        return this.request.toBuilder()
            .webIdentityToken(getIdentityToken())
            .build();
    }

    private String getIdentityToken() {
        return refreshIdToken(readRefreshToken(this.webIdentityRefreshTokenFile));
    }

    private String readRefreshToken(Path file) {
        try (InputStream webIdentityRefreshTokenStream =
Files.newInputStream(file)) {
```

```
        return IoUtils.toUtf8String(webIdentityRefreshTokenStream);
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }
}

private String refreshIdToken(String licenseRefreshToken) {
    final GetAccessTokenRequest request = GetAccessTokenRequest.builder()
        .token(licenseRefreshToken)
        .build();

    GetAccessTokenResponse response = this.lmClient.getAccessToken(request);
    return response.accessToken();
}

}

private static final class LicenseManagerClientFactory {
    private static final Duration DEFAULT_API_TIMEOUT = Duration.ofSeconds(30);
    private static final Duration DEFAULT_API_ATTEMPT_TIMEOUT =
Duration.ofSeconds(10);

    public static LicenseManagerClient create() {
        return getLicenseManagerClient();
    }

    private static LicenseManagerClient getLicenseManagerClient() {
        ClientOverrideConfiguration configuration =
ClientOverrideConfiguration.builder()
            .apiCallTimeout(DEFAULT_API_TIMEOUT)
            .apiCallAttemptTimeout(DEFAULT_API_ATTEMPT_TIMEOUT)
            .build();

        LicenseManagerClient client = LicenseManagerClient.builder()
            .region(configureLicenseManagerRegion())
            .credentialsProvider(AnonymousCredentialsProvider.create())
            .overrideConfiguration(configuration).build();
        return client;
    }

    private static Region configureLicenseManagerRegion() {
        Region defaultRegion = Region.US_EAST_1;

        Region region;
        try {
```

```
        region = (new DefaultAwsRegionProviderChain()).getRegion();
    } catch (RuntimeException ex) {
        region = defaultRegion;
    }
    return region;
}
}

private static final class StsClientFactory {
    private static final Duration DEFAULT_API_TIMEOUT = Duration.ofSeconds(30);
    private static final Duration DEFAULT_API_ATTEMPT_TIMEOUT =
Duration.ofSeconds(10);

    public static StsClient create() {
        return getStsClient();
    }

    private static StsClient getStsClient() {
        OrRetryCondition retryCondition = OrRetryCondition.create(new
StsRetryCondition(),
            RetryCondition.defaultRetryCondition());

        ClientOverrideConfiguration configuration =
ClientOverrideConfiguration.builder()
            .apiCallTimeout(DEFAULT_API_TIMEOUT)
            .apiCallAttemptTimeout(DEFAULT_API_ATTEMPT_TIMEOUT)
            .retryPolicy(r -> r.retryCondition(retryCondition))
            .build();

        return StsClient.builder()
            .region(configureStsRegion())
            .credentialsProvider(AnonymousCredentialsProvider.create())
            .overrideConfiguration(configuration).build();
    }

    private static Region configureStsRegion() {
        Region defaultRegion = Region.US_EAST_1;
        Region stsRegion;
        try {
            stsRegion = (new DefaultAwsRegionProviderChain()).getRegion();
        } catch (RuntimeException ex) {
            stsRegion = defaultRegion;
        }
        return stsRegion;
    }
}
```

```
    }

    private static final class StsRetryCondition implements RetryCondition {
        public boolean shouldRetry(RetryPolicyContext context) {
            return context.exception() instanceof IdpCommunicationErrorException;
        }
    }
}

private enum LicenseSystemSetting implements SystemSetting {
    AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE("aws.webIdentityRefreshTokenFile");

    private String systemProperty;
    private String defaultValue = null;

    LicenseSystemSetting(String systemProperty) {
        this.systemProperty = systemProperty;
    }

    @Override
    public String property() {
        return this.systemProperty;
    }

    @Override
    public String environmentVariable() {
        return this.name();
    }

    @Override
    public String defaultValue() {
        return this.defaultValue;
    }
}
}
```

LicenseManagerCredentialsProvider - Golang 実装

LicenseCredentialsProvider

LicenseCredentialsProvider は、AWS を追加することで、SDK のデフォルトの認証情報プロバイダーチェーンをオンプレミス用に拡張しますLicenseManagerTokenCredentialsProvider。

```
package lib

import (
    "context"
    "fmt"
    "sync"

    "github.com/aws/aws-sdk-go-v2/aws"
    "github.com/aws/aws-sdk-go-v2/config"
)

// LicenseCredentialsProvider is the custom credential provider that can retrieve valid
// temporary aws credentials
type LicenseCredentialsProvider struct {
    fallbackProvider aws.CredentialsProvider
    mux              sync.RWMutex
    licenseCredentials aws.Credentials
    err              error
}

// NewLicenseCredentialsProvider method will create a LicenseCredentialProvider Object
// which contains valid temporary aws credentials
func NewLicenseCredentialsProvider() (*LicenseCredentialsProvider, error) {
    licenseCredentialProvider := &LicenseCredentialsProvider{}
    fallbackProvider, err := createCredentialProvider()
    if err != nil {
        return licenseCredentialProvider, fmt.Errorf("failed to create
LicenseCredentialsProvider, %w", err)
    }
    licenseCredentialProvider.fallbackProvider = fallbackProvider
    return licenseCredentialProvider, nil
}

// Retrieve method will retrieve temporary aws credentials from the credential provider
func (l *LicenseCredentialsProvider) Retrieve(ctx context.Context) (aws.Credentials,
error) {
    l.mux.RLock()
    defer l.mux.RUnlock()
    l.licenseCredentials, l.err = l.fallbackProvider.Retrieve(ctx)
    return l.licenseCredentials, l.err
}

func createCredentialProvider() (aws.CredentialsProvider, error) {
```

```
// LoadDefaultConfig will examine all "default" credential providers
ctx := context.TODO()
cfg, err := config.LoadDefaultConfig(ctx)
if err != nil {
    return nil, fmt.Errorf("failed to create FallBackProvider, %w", err)
}

var useFallbackProvider bool
if cfg.Credentials != nil {
    if _, err := cfg.Credentials.Retrieve(ctx); err != nil {
        // If the "default" credentials provider cannot retrieve credentials, enable
        fallback to customCredentialsProvider.
        useFallbackProvider = true
    }
} else {
    useFallbackProvider = true
}

if useFallbackProvider {
    customProvider, err := newLicenseManagerTokenCredentialsProvider()
    if err != nil {
        return cfg.Credentials, fmt.Errorf("failed to create fallBackProvider, %w", err)
    }
    // wrap up customProvider with CredentialsCache to enable caching
    cfg.Credentials = aws.NewCredentialsCache(customProvider)
}
return cfg.Credentials, nil
}
```

LicenseManagerTokenCredentialsProvider

LicenseManagerTokenCredentialsProvider により、オンプレミス環境で License Manager OIDC が発行した ID トークンを使用して認証情報が提供されます。LicenseCredentialsProvider のソースコードをアプリケーションのクラスパスに含める必要があります。

```
package lib

import (
    "context"
    "fmt"
    "io/ioutil"
    "os"
```

```
"sync"
"time"

"github.com/aws/aws-sdk-go-v2/aws"
"github.com/aws/aws-sdk-go-v2/config"
"github.com/aws/aws-sdk-go-v2/service/sts"
)

const awsRefreshTokenFilePathEnvVar = "AWS_LICENSE_ACCESS_FILE"

// licenseManagerTokenCredentialsProvider defines and contains
  StsAssumeRoleWithWebIdentityProvider
type licenseManagerTokenCredentialsProvider struct {
  stsCredentialProvider *stsAssumeRoleWithWebIdentityProvider
  mux                    sync.RWMutex
  licenseCredentials    aws.Credentials
  err                   error
}

// Retrieve method will retrieve credentials from credential provider.
// Make this method public to make this provider satisfies CredentialProvider interface
func (a *licenseManagerTokenCredentialsProvider) Retrieve(ctx context.Context)
  (aws.Credentials, error) {
  a.mux.RLock()
  defer a.mux.RUnlock()
  a.licenseCredentials, a.err = a.stsCredentialProvider.Retrieve(ctx)
  return a.licenseCredentials, a.err
}

// newLicenseManagerTokenCredentialsProvider will create and return
  a LicenseManagerTokenCredentialsProvider Object which wraps up
  stsAssumeRoleWithWebIdentityProvider
func newLicenseManagerTokenCredentialsProvider()
  (*licenseManagerTokenCredentialsProvider, error) {
  // 1. Retrieve variables From yaml environment
  envConfig, err := config.NewEnvConfig()
  if err != nil {
    return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
    LicenseManagerTokenCredentialsProvider, %w", err)
  }
  roleArn := envConfig.RoleARN
  var roleSessionName string
  if envConfig.RoleSessionName == "" {
    roleSessionName = fmt.Sprintf("aws-sdk-go-v2-%v", time.Now().UnixNano())
  }
}
```

```
} else {
    roleSessionName = envConfig.RoleSessionName
}
tokenFilePath := os.Getenv(awsRefreshTokenFilePathEnvVar)
b, err := ioutil.ReadFile(tokenFilePath)
if err != nil {
    return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
LicenseManagerTokenCredentialsProvider, %w", err)
}
refreshToken := aws.String(string(b))

// 2. Create stsClient
cfg, err := config.LoadDefaultConfig(context.TODO())
if err != nil {
    return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
LicenseManagerTokenCredentialsProvider, %w", err)
}
stsClient := sts.NewFromConfig(cfg, func(o *sts.Options) {
    o.Region = configureStsClientRegion(cfg.Region)
    o.Credentials = aws.AnonymousCredentials{}
})

// 3. Configure StsAssumeRoleWithWebIdentityProvider
stsCredentialProvider := newStsAssumeRoleWithWebIdentityProvider(stsClient, roleArn,
roleSessionName, refreshToken)

// 4. Build and return
return &licenseManagerTokenCredentialsProvider{
    stsCredentialProvider: stsCredentialProvider,
}, nil
}

func configureStsClientRegion(configRegion string) string {
    defaultRegion := "us-east-1"
    if configRegion == "" {
        return defaultRegion
    } else {
        return configRegion
    }
}
```

コンテナ製品の Amazon SNS 通知

通知を受け取るには、製品の作成時に提供される AWS Marketplace Amazon Simple Notification Service (Amazon SNS) トピックをサブスクライブします。このトピックでは、製品の顧客のサブスクリプションの変更に関する通知が提供されます。例えば、これを使用して、顧客がプライベートオファーをいつ受け入れたかを知ることができます。

Note

製品の作成プロセス中に SNS トピックの実際の Amazon リソースネーム (ARN) に発行されます。例: `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE`

次の Amazon SNS トピックはコンテナ製品で使用できます。

- [Amazon SNS トピック: `aws-mp-subscription-notification`](#) - このトピックは、購入者が製品をサブスクライブまたはサブスクライブを解除した時に通知します。これは、時間単位や長期の時間単位を含む時間単位の料金モデルで利用できます。

Amazon SNS トピック: `aws-mp-subscription-notification`

`aws-mp-subscription-notification` トピック内の各メッセージの形式は次のとおりです。

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLEX",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXX",
  "offer-identifier": "offer-abcexample123",
  "isFreeTrialTermPresent": "true"
}
```

`<action-name>` は、通知によって異なります。指定できるアクションは以下のとおりです。

- `subscribe-success`
- `subscribe-fail`
- `unsubscribe-pending`
- `unsubscribe-success`

offer-identifier は、オファーがプライベートオファーの場合にのみ通知に表示されます。

Amazon SNS トピックへ Amazon SQS キューをサブスクライブする

Amazon SQS キューを指定された SNS トピックにサブスクライブすることを推奨します。SQS キューを作成し、そのキューをトピックにサブスクライブする方法の詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[Amazon SNS トピックへ Amazon SQS キューをサブスクライブする](#)」を参照してください。

Note

AWS Marketplace SNS トピックをサブスクライブできるのは、製品の販売 AWS アカウントに使用したからのみです。ただし、メッセージを別のアカウントに転送することはできません。詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[別のアカウントの Amazon SQS キューへ Amazon SNS メッセージを送信する](#)」を参照してください。

通知の SQS キューのポーリング

SQS キューを SNS トピックにサブスクライブすると、メッセージは SQS に保存されます。継続的にキューをポーリングし、メッセージを探して必要に応じて処理するサービスを定義する必要があります。

Machine Learning 製品

販売者は、AWS Marketplace を使用して、購入者が にデプロイできる機械学習 (ML) アルゴリズムとモデルを作成できます AWS。に記載されている Amazon SageMaker 製品には、次の 2 種類があります AWS Marketplace。

モデルパッケージ

購入者による追加のトレーニングを必要としない、予測を行うための事前トレーニング済みモデル。

アルゴリズム

予測を行う前に、購入者がトレーニングデータを提供する必要があるモデル。トレーニングアルゴリズムは含まれています。

これらの製品は、Amazon SageMaker コンソールまたは を通じて購入者が利用できます AWS Marketplace。購入者は、製品説明、ドキュメント、カスタマーレビュー、料金表、サポート情報を確認できます。モデルパッケージ製品またはアルゴリズム製品をサブスクライブすると、SageMaker コンソールの製品リストに追加されます。購入者は AWS SDKs、AWS Command Line Interface (AWS CLI)、または SageMaker コンソールを使用して、フルマネージド REST 推論エンドポイントを作成したり、データのバッチに対して推論を実行したりすることもできます。

Amazon で機械学習製品を作成するサポートについては SageMaker、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。

機械学習製品の使用を開始する

AWS Marketplace は、Amazon を使用する 2 つの機械学習製品タイプをサポートしています SageMaker。モデルパッケージ製品とアルゴリズム製品のどちらのタイプでも、予測を行うためのデプロイ可能な推論モデルが生成されます。

SageMaker モデルパッケージ

[Amazon SageMaker モデルパッケージ](#)製品には、事前トレーニング済みのモデルが含まれています。トレーニング済みのモデルは にデプロイ SageMaker して、リアルタイムまたはバッチで推論または予測を行うことができます。本製品には、モデルアーティファクト (存在する場合) を含むトレーニング済みの推論コンポーネントが含まれています。販売者は、 を使用してモデルをトレーニング SageMaker したり、独自のモデルを導入したりできます。

SageMaker アルゴリズム

購入者は [SageMaker アルゴリズム](#) 製品を使用して、完全な機械学習ワークロードを実行できます。アルゴリズム製品には、トレーニングと推論という 2 つの論理コンポーネントがあります。では SageMaker、購入者は独自のデータセットを使用して、トレーニングコンポーネントでトレーニングジョブを作成します。トレーニングコンポーネントのアルゴリズムが完了すると、機械学習モデルのモデルアーティファクトが生成されます。は、購入者の Amazon Simple Storage Service (Amazon S3) バケットにモデルアーティファクト SageMaker を保存します。では SageMaker、購入者は、生成されたモデルアーティファクトとともに推論コンポーネントをデプロイして、リアルタイムまたはバッチで推論 (または予測) を実行できます。

推論モデルのデプロイ

推論モデルがモデルパッケージから作成されたものであれ、アルゴリズムから作成されたものであれ、それらをデプロイするには次の 2 つの方法があります。

- エンドポイント — このメソッドは SageMaker、を使用してモデルをデプロイし、API エンドポイントを作成します。購入者はこのエンドポイントをバックエンドサービスの一部として使用してアプリケーションを強化できます。データがエンドポイントに送信されると、はそれをモデルコンテナに SageMaker 渡し、API レスポンスで結果を返します。エンドポイントとコンテナは、購入者が停止するまで実行され続けます。

Note

では AWS Marketplace、エンドポイントメソッドはリアルタイム推論 と呼ばれ、SageMaker ドキュメントではホスティングサービス と呼ばれます。詳細については、[「Amazon にモデルをデプロイ SageMaker する」](#)を参照してください。

- バッチ変換ジョブ - この方法では、購入者は推論用データセットを Amazon S3 に保存します。バッチ変換ジョブが開始されると、はモデルを SageMaker デプロイし、S3 バケットからモデルのコンテナにデータを渡し、その結果を S3 バケットに返します。ジョブが完了すると、はジョブを SageMaker 停止します。詳細については、[「バッチ変換を使用してデータセット全体の推論を取得する」](#)を参照してください。

Note

はデータをモデルに渡し、結果を購入者に返すため SageMaker、どちらの方法もモデルに対して透過的です。

セキュリティと知的財産

Amazon は、 から取得したモデルとアルゴリズムの知的財産と購入者データの両方 SageMaker を保護します AWS Marketplace。

知的財産の保護

製品を作成すると、コードは Docker コンテナイメージにパッケージ化されます。詳細については、このガイドで後述する「[で製品を準備する SageMaker](#)」を参照してください。コンテナイメージをアップロードすると、イメージとアーティファクトは転送中も保存時も暗号化されます。また、イメージは公開前にスキャンされ、脆弱性がないことを確認されます。

知的財産を保護するために、 は購入者のみが AWS のサービス エンドポイントを介して製品にアクセス SageMaker できるようにします。購入者は、コンテナイメージやモデルアーティファクトに直接アクセスしたり、コンテナイメージやモデルアーティファクトを取得したりすることはできません。また、基盤となるインフラストラクチャにアクセスすることはできません。

ネットワークアクセスなし

購入者が作成する SageMaker モデルやアルゴリズムとは異なり、購入者が から製品を起動すると AWS Marketplace、モデルやアルゴリズムはネットワークにアクセスせずにデプロイされます。SageMaker は、ネットワークや AWS のサービス エンドポイントにアクセスできない環境にイメージをデプロイします。例えば、コンテナイメージは、インターネット、[VPC エンドポイント](#)、またはその他の AWS のサービスへのアウトバウンド API 呼び出しを行うことができません。

顧客データのセキュリティ

製品は購入者の SageMaker 内で実行されます AWS アカウント。そのため、購入者が製品を使用してデータ推論を行っても、販売者はデータにアクセスできません。

アルゴリズム製品の場合、トレーニングジョブのたびにトレーニングイメージからモデルアーティファクトが出力されます。モデルアーティファクトは購入者のアカウントに保存されます。トレーニングジョブのモデルアーティファクトは、購入者が顧客の推論イメージを使用してモデルをデプロイするときに使用されます。モデルアーティファクトに含まれる可能性のある知的財産を保護するには、公開前に暗号化してください。

Important

このセキュリティモデルにより、実行時にコードがインターネットにアクセスするのを防ぎます。したがって、コードはインターネット上のリソースやライブラリを使用できない

め、依存関係は Docker コンテナイメージにパッケージ化してください。トレーニングジョブから出力されたアーティファクトを暗号化する場合、これは特に重要です。ランタイムには、アーティファクトを暗号化および復号化するキーにインターネット経由でアクセスすることはできません。これらは、イメージとパッケージ化する必要があります。

詳細については、「[Amazon のセキュリティ SageMaker](#)」を参照してください。

機械学習製品の料金

Amazon SageMaker 製品で利用可能な複数の料金モデルから選択できます。製品をサブスクライブする購入者は、自分の SageMaker 内で製品を実行します AWS アカウント。購入者に提示される料金は、購入者の AWS アカウント で稼働するリソースのインフラストラクチャコストと、設定した製品価格の組み合わせです。

インフラストラクチャの料金

購入者は、製品の使用 SageMaker 中の のインフラストラクチャコストをすべて負担します。これらのコストは によって設定 AWS され、[Amazon の SageMaker 料金](#) ページで確認できます。

ソフトウェアの料金

製品の使用に対して購入者に AWS Marketplace 請求するソフトウェア料金を決定します。機械学習製品を AWS Marketplace に追加するときに、料金と条件を設定します。

インスタンスタイプごとのすべてのインフラストラクチャとソフトウェアの料金は、購入者がサブスクライブ AWS Marketplace する前に、 の製品リストページに表示されます。

料金無料

製品を無料で提供することもできます。この場合、購入者はインフラストラクチャの料金のみを支払います。

時間単位の料金

で実行されているソフトウェアのインスタンスごとに、1 時間あたりの料金で製品を提供できます SageMaker。ソフトウェアが実行するインスタンスタイプごとに異なる時間単位の料金を請求できます。購入者がソフトウェアを実行している間、 は使用状況 AWS Marketplace を追跡し、それに従って購入者に請求します。使用量は分単位で計算されます。

モデルパッケージ製品の場合、購入者はソフトウェアを2つの方法で実行できます。エンドポイントを継続的にホストしてリアルタイムの推論を実行する方法と、データセットに対してバッチ変換ジョブを実行する方法です。購入者がソフトウェアを実行する2つの方法にそれぞれ異なる料金を設定できます。

アルゴリズム製品では、前述のように推論の実行に対して料金を決定するだけでなく、トレーニングジョブに対しても時間単位の料金を決定します。トレーニングイメージがサポートするインスタンスタイプごとに異なる時間単位の料金を請求できます。

時間単位料金での年間契約

契約オプションで固定の前払い料金を指定でき、サブスクリプションの時点で契約の全額が顧客に請求されます。年間契約の終了時に、引き続き実行するインスタンスには設定した時間単位の料金が発生します。

推論の料金

購入者がエンドポイントをホストしてソフトウェアを実行し、リアルタイムの推論を継続的に実行する場合、推論ごとに料金を設定することができます。

Note

バッチ変換プロセスでは、常に時間単位の料金が適用されます。アルゴリズム製品のトレーニングジョブでも、常に時間単位の料金が適用されます。これらの料金は、推論の料金と無関係に、また相互にも無関係に設定できます。

デフォルトでは、推論料金では、はエンドポイントの呼び出しごとに購入者に AWS Marketplace 課金します。ただし、ソフトウェアが1回の呼び出しで推論のバッチを処理する場合もあります(ミニバッチとも呼ばれます)。エンドポイントのデプロイでは、1回の呼び出しに対して AWS Marketplace が購入者に請求する推論の数をカスタムで指定できます。これを行うには、次の例のように、呼び出しの HTTP レスポンスヘッダーにカスタムメータリングヘッダーを含めます。この例では、3件の推論を購入者に請求する呼び出しを示しています。

```
X-Amzn-Inference-Metering: {"Dimension": "inference.count", "ConsumedUnits": 3}
```

Note

推論料金では、HTTP レスポンスコードが 2XX であるリクエストに対して AWS Marketplace のみ購入者に課金されます。

無料トライアル

オプションで、製品の無料トライアルを作成し、無料トライアルの日数を定義できます。無料トライアル期間は 5～120 日間です。無料トライアル期間中、購入者はソフトウェアを好きなだけ実行でき、ソフトウェア料金は発生しません。無料トライアル中、インフラストラクチャの料金は発生しません。トライアル期間が終了すると、通常のソフトウェア料金とインフラストラクチャの料金が請求されます。

Note

無料トライアルを作成できるのは、時間単位で請求されるオファーのみです。推論の料金が適用される製品の無料トライアルは作成できません。

購入者が無料トライアルのある製品をサブスクライブすると、ウェルカムメールメッセージが届きます。メッセージには、無料トライアルの期間、計算された有効期限、サブスクリプションの解除に関する詳細が含まれます。有効期限が切れる 3 日前に通知メールが送信されます。

で製品の無料トライアルを提供する場合 AWS Marketplace、無料トライアルの特定の[返金ポリシー](#)に同意したものと見なされます。

Note

機械学習のプライベートオファーについては、「[プライベートオファー](#)」をご覧ください。

料金の変更

販売者は、[AWS Marketplace セラーオペレーションチーム](#)に連絡して、機械学習製品の料金を変更できます。製品 ID と新しい料金の詳細を入力します。新しい料金は 90 日後に有効になります。また、2 回目の料金変更を行うには 90 日待つ必要があります。この制限は、サポートされているイ

インスタンスの既存のリストに新しいインスタンスタイプを追加する場合にも適用されます。例として、2023年11月1日に機械学習製品の料金を引き上げた場合、2024年1月30日以降に新しいインスタンスタイプの追加や、2回目の料金変更が可能です。

で製品を準備する SageMaker

で製品を公開する前に AWS Marketplace、Amazon で製品を準備する必要があります SageMaker。製品の準備では、次の3つのステップに従います。

1. [コードをイメージにパッケージ化する](#) - モデルパッケージまたはアルゴリズム製品を準備するには、製品用の Docker コンテナイメージを作成する必要があります。
2. [イメージのアップロード](#) - コードをコンテナイメージにパッケージ化してローカルでテストしたら、イメージをアップロードしてスキャンし、既知の脆弱性を持っていないかを確認します。続行する前に、すべての脆弱性を修正してください。
3. [Amazon SageMaker リソースの作成](#) - イメージが正常にスキャンされたら、イメージを使用してモデルパッケージまたはアルゴリズムリソースを作成できます SageMaker。

コードをイメージにパッケージ化する

の機械学習製品は、Amazon AWS Marketplace を使用して SageMaker、購入者に提供する機械学習ロジックを作成して実行します。は、ロジックを含む Docker コンテナイメージ SageMaker を実行します。SageMaker は、安全でスケーラブルなインフラストラクチャでこれらのコンテナを実行します。詳細については、「[セキュリティと知的財産](#)」を参照してください。

トピック

- [どのタイプのコンテナイメージを作成すればよいですか？](#)
- [モデルパッケージイメージ](#)
- [アルゴリズムイメージ](#)

どのタイプのコンテナイメージを作成すればよいですか？

コンテナイメージには、推論イメージとトレーニングイメージの2つのタイプがあります。

モデルパッケージ製品を作成するには、推論イメージのみが必要です。詳細な手順については、「[モデルパッケージイメージ](#)」を参照してください。

アルゴリズム製品を作成するには、トレーニングイメージと推論イメージの両方が必要です。詳細な手順については、「[アルゴリズムイメージ](#)」を参照してください。

コードをコンテナイメージに適切にパッケージ化するには、コンテナがファイル構造に従う必要があります SageMaker。サービスがコンテナとデータをやり取りできるように、コンテナは正しいエンドポイントを公開する必要があります。以下のセクションでは、このプロセスを詳しく説明します。

Important

セキュリティ上の理由から、コンテナ化された製品を購入者がサブスクライブすると、Docker コンテナはインターネット接続のない隔離された環境で実行されます。コンテナの作成時には、インターネット経由での呼び出しに依存しないでください。その呼び出しは失敗します。への呼び出しも失敗 AWS のサービスします。詳細については、「[セキュリティと知的財産](#)」セクションを参照してください。

必要に応じて、推論イメージとトレーニングイメージを作成するときに、開始点として、[使用可能な深層学習コンテナイメージ](#)のコンテナを使用します。イメージは既にさまざまな機械学習フレームワークで適切にパッケージ化されています。

モデルパッケージイメージ

Amazon SageMaker モデルパッケージは、予測を行う事前トレーニング済みのモデルであり、購入者による追加のトレーニングは必要ありません。

モデルパッケージには、次のコンポーネントが含まれています。

- [Amazon Elastic Container Registry](#) (Amazon ECR) に保存された推論イメージ
- (オプション) [Amazon S3](#) に個別に保存されているモデルアーティファクト

Note

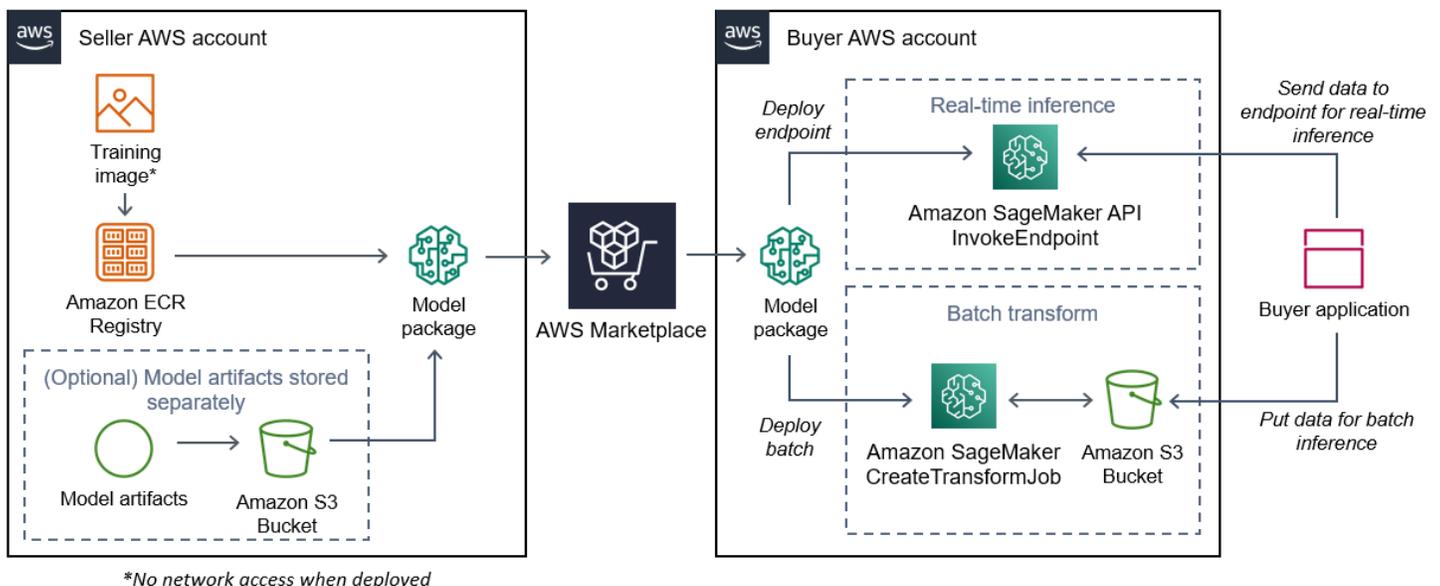
モデルアーティファクトは、モデルが予測を行うために使用するファイルであり、通常は独自のトレーニングプロセスの結果です。アーティファクトは、モデルが必要とするどのようなファイルタイプでも構いませんが、.tar.gz 圧縮を使用する必要があります。モデルパッケージの場合、推論イメージ内にバンドルすることも、Amazon に個別に保存することもできます SageMaker。Amazon S3 に保存されているモデルアーティファクトは、実行時に推論コンテナにロードされます。モデルパッケージを公開すると、これらのアーティファクト

は公開され、購入者が直接アクセスできない AWS Marketplace 所有の Amazon S3 バケットに保存されます。

Tip

推論モデルが Gluon、Keras、MXNet、TensorFlow Lite、PyTorch、TensorFlow、ONNX などの深層学習フレームワークで構築されている場合は、Amazon SageMaker Neo の使用を検討してください。Neo は、m1.c4、m1.p2 などの、クラウドインスタンスタイプの特定のファミリーにデプロイする推論モデルを自動的に最適化できます。詳細については、「Amazon SageMaker デベロッパーガイド」の「[Neo を使用してモデルのパフォーマンスを最適化する](#)」を参照してください。

次の図は、モデルパッケージ製品を公開して使用するワークフローを示しています。



1. 販売者は推論イメージ (デプロイ時のネットワークアクセス不可) を作成し、Amazon ECR レジストリにプッシュします。

モデルアーティファクトは推論イメージにバンドルすることも S3 に個別に保存することもできます。

2. 次に、販売者は Amazon でモデルパッケージリソース SageMaker を作成し、ML 製品を に公開します AWS Marketplace。
3. 購入者は ML 製品をサブスクライブしてモデルをデプロイします。

Note

このモデルは、リアルタイムの推論を行うエンドポイントとしてデプロイすることも、データセット全体の予測を一度に取得するバッチジョブとしてデプロイすることもできます。詳細については、「[推論のためのモデルをデプロイする](#)」を参照してください。

4. SageMaker は推論イメージを実行します。推論イメージにバンドルされていない販売者提供のモデルアーティファクトは、ランタイムに動的に読み込まれます。
5. SageMaker は、コンテナの HTTP エンドポイントを使用して購入者の推論データをコンテナに渡し、予測結果を返します。

モデルパッケージ用の推論イメージの作成

このセクションでは、推論コードをモデルパッケージ製品の推論イメージにパッケージ化する手順を説明します。そのプロセスは、以下のステップで構成されています。

ステップ

- [ステップ 1: コンテナイメージを作成する](#)
- [ステップ 2: イメージをローカルでビルドしてテストする](#)

推論イメージは、推論ロジックを含む Docker イメージです。実行時にコンテナは HTTP エンドポイントを公開し、SageMaker がコンテナとの間でデータを渡すことを許可します。

Note

以下は、推論イメージのパッケージコードの一例です。詳細については、「[での Docker コンテナの使用 SageMaker](#)」および[AWS Marketplace SageMaker 「」の例](#)を参照してください [GitHub](#)。

以下の例ではわかりやすくするために [Flask](#) というウェブサービスを使用していますが、本番環境に対応しているとは見なされていません。

ステップ 1: コンテナイメージを作成する

推論イメージを と互換性を持たせるには SageMaker、Docker イメージが HTTP エンドポイントを公開する必要があります。コンテナの実行中に、 は推論用の SageMaker 購入者入力をコンテナの HTTP エンドポイントに渡します。推論結果は HTTP レスポンスの本文で返されます。

次のチュートリアルでは、Linux Ubuntu ディストリビューションを使用する開発環境で Docker CLI を使用します。

- [ウェブサーバースクリプトを作成する](#)
- [コンテナ実行用のスクリプトを作成する](#)
- [Dockerfile の作成](#)
- [モデルアーティファクトをパッケージ化またはアップロードする](#)

ウェブサーバースクリプトを作成する

この例では [Flask](#) という Python サーバーを使用していますが、フレームワークに適した任意のウェブサーバーを使用できます。

Note

ここではわかりやすくするために [Flask](#) を使用しています。本番環境に対応するウェブサーバーとは見なされません。

SageMaker が使用する TCP ポート 8080 で 2 つの HTTP エンドポイントを提供する Flask ウェブサーバースクリプトを作成します。想定されるエンドポイントは次の 2 つです。

- `/ping` - このエンドポイントに HTTP GET リクエスト SageMaker を行い、コンテナの準備が整っているかどうかを確認します。コンテナの準備が完了すると、コンテナはこのエンドポイントでの HTTP GET リクエストに HTTP 200 レスポンスコードで応答します。
- `/invocations` - SageMaker 推論のためにこのエンドポイントに HTTP POST リクエストを行います。推論用の入力データはリクエストの本文で送信されます。ユーザー指定のコンテンツタイプは HTTP ヘッダーで渡されます。レスポンスの本文は推論出力です。タイムアウトの詳細については、「[機械学習製品を作成するための要件とベストプラクティス](#)」を参照してください。

`./web_app_serve.py`

```
# Import modules
import json
import re
from flask import Flask
from flask import request
app = Flask(__name__)
```

```
# Create a path for health checks
@app.route("/ping")
def endpoint_ping():
    return ""

# Create a path for inference
@app.route("/invocations", methods=["POST"])
def endpoint_invocations():

    # Read the input
    input_str = request.get_data().decode("utf8")

    # Add your inference code between these comments.
    #
    #
    #
    #
    # Add your inference code above this comment.

    # Return a response with a prediction
    response = {"prediction":"a","text":input_str}
    return json.dumps(response)
```

前の例には、実際の推論ロジックはありません。実際の推論イメージについては、ウェブアプリに推論ロジックを追加し、入力を処理して実際の予測を返します。

推論イメージには、インターネットアクセスもへの呼び出しもできないため、必要な依存関係がすべて含まれている必要があります AWS のサービス。

Note

これと同じコードがリアルタイム推論とバッチ推論の両方で呼び出されます。

コンテナ実行用のスクリプトを作成する

Docker コンテナイメージ `serve SageMaker` の実行時に実行される という名前のスクリプトを作成します。次のスクリプトは HTTP ウェブサーバーを起動します。

./serve

```
#!/bin/bash

# Run flask server on port 8080 for SageMaker
flask run --host 0.0.0.0 --port 8080
```

Dockerfile の作成

ビルドコンテキストに Dockerfile を作成します。この例では Ubuntu 18.04 を使用していますが、フレームワークに適していれば、どのベースイメージからでも開始できます。

./Dockerfile

```
FROM ubuntu:18.04

# Specify encoding
ENV LC_ALL=C.UTF-8
ENV LANG=C.UTF-8

# Install python-pip
RUN apt-get update \
&& apt-get install -y python3.6 python3-pip \
&& ln -s /usr/bin/python3.6 /usr/bin/python \
&& ln -s /usr/bin/pip3 /usr/bin/pip;

# Install flask server
RUN pip install -U Flask;

# Add a web server script to the image
# Set an environment to tell flask the script to run
COPY /web_app_serve.py /web_app_serve.py
ENV FLASK_APP=/web_app_serve.py

# Add a script that Amazon SageMaker will run
# Set run permissions
# Prepend program directory to $PATH
COPY /serve /opt/program/serve
RUN chmod 755 /opt/program/serve
ENV PATH=/opt/program:${PATH}
```

Dockerfile は、以前に作成した 2 つのスクリプトをイメージに追加します。serve スクリプトのディレクトリが PATH に追加されると、コンテナの実行時にそれを実行できるようになります。

モデルアーティファクトをパッケージ化またはアップロードする

モデルのトレーニングから推論イメージまで、モデルアーティファクトを提供する方法には、以下の2つがあります。

- 推論イメージと共に静的にパッケージ化します。
- ランタイムに動的にロードします。動的に読み込まれるため、同じイメージをさまざまな機械学習モデルのパッケージ化に使用できます。

モデルのアーティファクトを推論イメージと一緒にパッケージ化する場合は、アーティファクトを Dockerfile に含めてください。

モデルアーティファクトを動的にロードする場合は、それらのアーティファクトを Amazon S3 の圧縮ファイル (.tar.gz) に個別に保存します。モデルパッケージを作成するときは、圧縮ファイルの場所を指定し、コンテナの実行/opt/ml/model/時にコンテンツを SageMaker 抽出してコンテナディレクトリにコピーします。モデルパッケージを公開すると、それらのアーティファクトは、購入者が直接アクセスできない、AWS Marketplace 所有の Amazon S3 バケツに公開および保存されます。

ステップ 2: イメージをローカルでビルドしてテストする

ビルドコンテキストには、現在、以下のファイルが存在します。

- ./Dockerfile
- ./web_app_serve.py
- ./serve
- 推論ロジックと (オプションの) 依存関係

次に、コンテナイメージをビルド、実行、テストします。

イメージを構築する

ビルドコンテキストで Docker コマンドを実行し、イメージをビルドしてタグ付けします。この例ではタグ my-inference-image を使用します。

```
sudo docker build --tag my-inference-image ./
```

この Docker コマンドを実行してイメージをビルドすると、Dockerfile の各行に基づいて Docker がイメージをビルドするときの出力が表示されます。終了すると、次のようなものが表示されます。

```
Successfully built abcdef123456
Successfully tagged my-inference-image:latest
```

をローカルで実行する

ビルドが完了したら、イメージをローカルでテストできます。

```
sudo docker run \
  --rm \
  --publish 8080:8080/tcp \
  --detach \
  --name my-inference-container \
  my-inference-image \
  serve
```

コマンドの詳細は次のとおりです。

- `--rm` - コンテナが停止したら自動的に削除します。
- `--publish 8080:8080/tcp` - ポート 8080 を公開して、HTTP リクエスト SageMaker を送信するポートをシミュレートします。
- `--detach` - コンテナをバックグラウンドで実行します。
- `--name my-inference-container` - 実行中のこのコンテナに名前を付けます。
- `my-inference-image` - ビルドされたイメージを実行します。
- `serve` - コンテナの実行時に SageMaker 実行するのと同じスクリプトを実行します。

このコマンドを実行すると、Docker は、ビルドした推論イメージからコンテナを作成してバックグラウンドで実行します。コンテナは `serve` スクリプトを実行し、テスト目的でウェブサーバーを起動します。

HTTP エンドポイントへの ping をテストします。

がコンテナ SageMaker を実行すると、定期的にエンドポイントに ping を送信します。エンドポイントがステータスコード 200 の HTTP レスポンスを返すと、コンテナ SageMaker が推論できる状態であることを知らせます。これをテストするには、次のコマンドを実行します。このコマンドはエンドポイントをテストしてレスポンスヘッダーを含めます。

```
curl --include http://127.0.0.1:8080/ping
```

出力例は次のとおりです。

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 0
Server: MyServer/0.16.0 Python/3.6.8
Date: Mon, 21 Oct 2019 06:58:54 GMT
```

推論 HTTP エンドポイントをテストします。

コンテナが ping に 200 のステータスコードを返すことで準備が整っていることを示すと、SageMaker はPOSTリクエストを介して推論データを HTTP /invocations エンドポイントに渡します。以下のコマンドを実行して、推論ポイントをテストします。

```
curl \
  --request POST \
  --data "hello world" \
  http://127.0.0.1:8080/invocations
```

出力例は次のとおりです。

```
{"prediction": "a", "text": "hello world"}
```

これら 2 つの HTTP エンドポイントが機能するので、推論イメージは と互換性があるようになりました SageMaker。

Note

モデルパッケージ製品のモデルは、リアルタイムとバッチの 2 つの方法でデプロイできます。どちらのデプロイでも、は Docker コンテナの実行中に同じ HTTP エンドポイント SageMaker を使用します。

コンテナを停止するには、次のコマンドを実行します。

```
sudo docker container stop my-inference-container
```

推論イメージの準備とテストが完了したら、次の「[イメージのアップロード](#)」に進むことができます。

アルゴリズムイメージ

Amazon SageMaker アルゴリズムでは、購入者が予測を行う前に独自のデータを持ってトレーニングする必要があります。

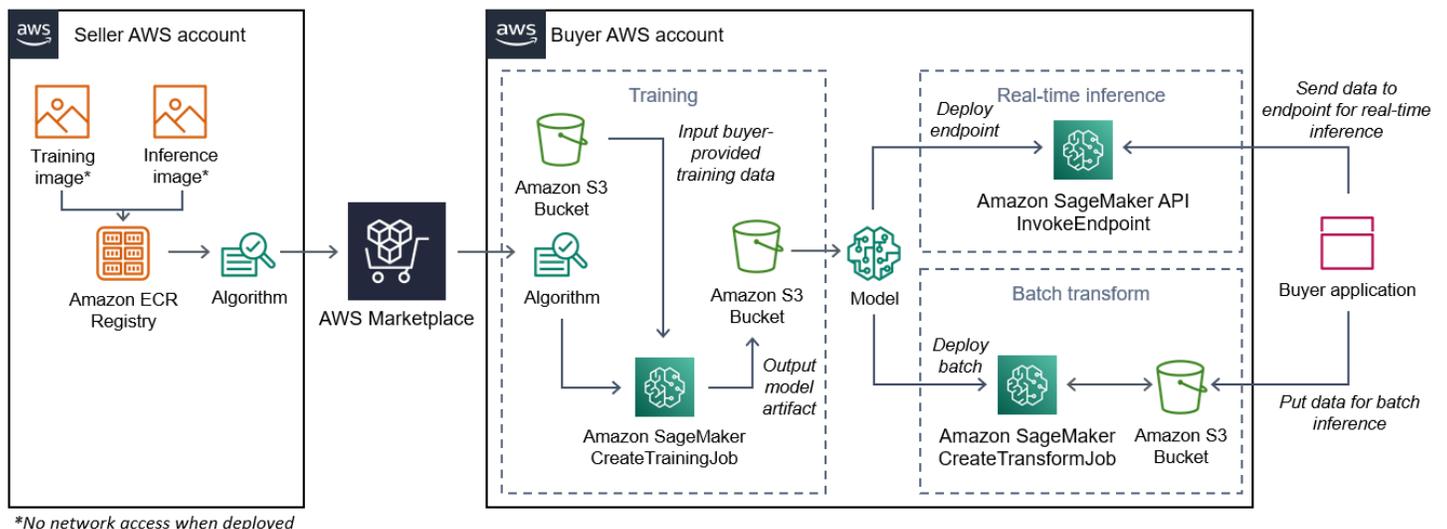
アルゴリズムには以下のコンポーネントが含まれています。

- [Amazon ECR](#) に保存されたトレーニングイメージ
- Amazon Elastic Container Registry (Amazon ECR) に保存された推論イメージ

Note

アルゴリズム製品では、トレーニングコンテナによってモデルアーティファクトが生成され、モデルのデプロイ時に推論コンテナに読み込まれます。

次の図は、アルゴリズム製品を公開して使用するワークフローを示しています。



1. 販売者はトレーニングイメージと推論イメージ (デプロイ時のネットワークアクセス不可) を作成し、Amazon ECR レジストリにアップロードします。
2. 次に、販売者は Amazon でアルゴリズムリソースを作成し SageMaker、ML 製品を に公開します AWS Marketplace。
3. 購入者は ML 製品をサブスクライブします。
4. 購入者は、互換性のあるデータセットと適切なハイパーパラメータ値を使用してトレーニングジョブを作成します。 はトレーニングイメージ SageMaker を実行し、トレーニングデータとハ

イパーパラメータをトレーニングコンテナにロードします。トレーニングジョブが完了すると、`/opt/ml/model/`にあるモデルアーティファクトが圧縮され、購入者の [Amazon S3](#) バケットにコピーされます。

- 購入者は Amazon S3 に保存されているトレーニングのモデルアーティファクトを含むモデルパッケージを作成し、モデルをデプロイします。
- SageMaker は推論イメージを実行し、圧縮されたモデルアーティファクトを抽出し、推論コンテナディレクトリパスにファイルをロードします。`/opt/ml/model/`このパスは、推論を提供するコードによって消費されます。
- モデルがエンドポイントとしてデプロイされるかバッチ変換ジョブとしてデプロイされるかにかかわらず、SageMaker は購入者に代わって推論用のデータをコンテナの HTTP エンドポイント経由でコンテナに渡し、予測結果を返します。

Note

詳細については、「[モデルのトレーニング](#)」を参照してください。

アルゴリズム用のトレーニングイメージの作成

このセクションでは、トレーニングコードをトレーニングイメージにパッケージ化する手順を説明します。アルゴリズム製品を作成するには、トレーニングイメージが必要です。

トレーニングイメージは、トレーニングアルゴリズムを含む Docker イメージです。コンテナは、コンテナ SageMaker との間でデータをコピーできるように、特定のファイル構造に従います。

アルゴリズム製品を公開する際は、トレーニングイメージと推論イメージの両方が必要です。トレーニングイメージを作成したら、推論イメージを作成する必要があります。2つのイメージは1つのイメージに結合することも、個別のイメージのままにしておくこともできます。イメージを結合するか個別のイメージのままにするかは、ユーザー次第です。通常、推論は学習よりも単純です。イメージを結合せずに個別のイメージのままにすると、推論のパフォーマンスが向上することがあります。

Note

以下は、トレーニングイメージのパッケージコードの一例です。詳細については、「[で独自のアルゴリズムとモデルを使用する](#)」および [AWS Marketplace SageMaker 「」の例](#)を参照してください [GitHub](#)。

ステップ

- [ステップ 1: コンテナイメージを作成する](#)
- [ステップ 2: イメージをローカルでビルドしてテストする](#)

ステップ 1: コンテナイメージを作成する

トレーニングイメージを Amazon と互換性を持たせるには SageMaker、トレーニングデータと設定入力をコンテナ内の特定のパス SageMaker にコピーできるように、特定のファイル構造に従う必要があります。トレーニングが完了すると、生成されたモデルアーティファクトは、が SageMaker コピー元のコンテナ内の特定のディレクトリパスに保存されます。

以下では、Linux の Ubuntu ディストリビューションの開発環境にインストールされた Docker CLI が使用されています。

- [設定入力を読み取れるようプログラムを準備します。](#)
- [データ入力を読み取れるようプログラムを準備します。](#)
- [トレーニング出力を書き込めるようプログラムを準備します。](#)
- [コンテナ実行用のスクリプトを作成する](#)
- [Dockerfile の作成](#)

設定入力を読み取れるようプログラムを準備します。

トレーニングプログラムで購入者提供の設定入力が必要な場合は、実行時に以下がコンテナ内にコピーされます。必要に応じて、プログラムはこれらの特定のファイルパスから読み取りを行う必要があります。

- /opt/ml/input/config は、プログラムの実行方法を制御する情報が保存されているディレクトリです。
- hyperparameters.json は、ハイパーパラメータ名とハイパーパラメータ値の、JSON 形式の辞書です。値は文字列のため、変換が必要になる場合があります。
- resourceConfig.json は JSON 形式のファイルで、[分散型トレーニング](#)に使用されるネットワークレイアウトを記述しています。トレーニングイメージが分散型トレーニングをサポートしていない場合、このファイルは無視してかまいません。

Note

設定入力の詳細については、[「Amazon SageMaker がトレーニング情報を提供する方法」](#)を参照してください。

データ入力を読み取れるようプログラムを準備します。

トレーニングデータは、次の 2 つのモードのいずれかでコンテナに渡すことができます。コンテナ内で実行されるトレーニングプログラムは、これら 2 つのモードのいずれかでトレーニングデータのダイジェストを作成します。

ファイルモード

- `/opt/ml/input/data/<channel_name>/` にそのチャンネルの入力データが含まれます。チャンネルは `CreateTrainingJob` オペレーションの呼び出しに基づいて作成されますが、一般的にはアルゴリズムが期待するものとチャンネルが一致することが重要です。各チャンネルのファイルは [Amazon S3](#) からこのディレクトリにコピーされ、Amazon S3 のキー構造によって示されるツリー構造が保持されます。

パイプモード

- `/opt/ml/input/data/<channel_name>_<epoch_number>` は特定のエポックのパイプです。エポックは 0 から始まり、読み込まれるたびに 1 ずつ増えていきます。実行できるエポックの数に制限はありませんが、次のエポックを読み込む前に各パイプを閉じる必要があります。

トレーニング出力を書き込めるようプログラムを準備します。

トレーニングの出力は次のコンテナディレクトリに書き込まれます。

- `/opt/ml/model/` はトレーニングアルゴリズムが生成するモデルまたはモデルアーティファクトが書き込まれるディレクトリです。モデルはどのような形式でもかまいません。1 つのファイルでも、ディレクトリ全体の `tree. SageMaker packages` でも、このディレクトリ内のすべてのファイルを圧縮ファイル (`.tar.gz`) にすることができます。このファイルは、`DescribeTrainingJob` API オペレーションによって返される、Amazon S3 の場所にあります。
- `/opt/ml/output/` は、ジョブが失敗した理由が記述された `failure` ファイルをアルゴリズムが書き込むことができるディレクトリです。このファイルの内容は `DescribeTrainingJob` の結

果の FailureReason フィールドに返されます。ジョブが成功してもこのファイルは無視されるため、このファイルを書き込む理由はありません。

コンテナ実行用のスクリプトを作成する

Docker train コンテナイメージ SageMaker の実行時に実行されるシェルスクリプトを作成します。トレーニングが完了し、モデルアーティファクトがそれぞれのディレクトリに書き込まれたら、スクリプトを終了します。

./train

```
#!/bin/bash

# Run your training program here
#
#
#
#
```

Dockerfile の作成

ビルドコンテキストに Dockerfile を作成します。この例ではベースイメージとして Ubuntu 18.04 を使用していますが、フレームワークに適していれば、どのベースイメージからでも開始できます。

./Dockerfile

```
FROM ubuntu:18.04

# Add training dependencies and programs
#
#
#
#
# Add a script that SageMaker will run
# Set run permissions
# Prepend program directory to $PATH
COPY /train /opt/program/train
RUN chmod 755 /opt/program/train
ENV PATH=/opt/program:${PATH}
```

Dockerfile は以前に作成した train スクリプトをイメージに追加します。スクリプトのディレクトリが PATH に追加され、コンテナの実行時にスクリプトを実行できるようになります。

前の例には、実際のトレーニングロジックはありません。実際のトレーニングイメージでは、トレーニングの依存関係を Dockerfile に追加し、トレーニング入力を読み取るロジックを追加してトレーニングを行い、モデルアーティファクトを生成します。

トレーニングイメージはインターネットにアクセスできないため、必要な依存関係がすべて含まれている必要があります。

詳細については、[「で独自のアルゴリズムとモデルを使用する」および AWS Marketplace AWS Marketplace SageMaker 「」の例](#)を参照してください GitHub。

ステップ 2: イメージをローカルでビルドしてテストする

ビルドコンテキストには、現在、以下のファイルが存在します。

- ./Dockerfile
- ./train
- トレーニングの依存関係とロジック

次に、このコンテナイメージをビルド、実行、テストできます。

イメージを構築する

ビルドコンテキストで Docker コマンドを実行し、イメージをビルドしてタグ付けします。この例ではタグ my-training-image を使用します。

```
sudo docker build --tag my-training-image ./
```

この Docker コマンドを実行してイメージをビルドすると、Dockerfile の各行に基づいて Docker がイメージをビルドするときの出力が表示されます。終了すると、次のようなものが表示されます。

```
Successfully built abcdef123456  
Successfully tagged my-training-image:latest
```

をローカルで実行する

完了したら、次の例に示すようにイメージをローカルでテストします。

```
sudo docker run \  
  --rm \  
  --volume '<path_to_input>:/opt/ml/input:ro' \  
  --volume '<path_to_model>:/opt/ml/model' \  
  --volume '<path_to_output>:/opt/ml/output' \  
  --name my-training-container \  
  my-training-image \  
  train
```

コマンドの詳細は次のとおりです。

- `--rm` - コンテナが停止したら自動的に削除します。
- `--volume '<path_to_input>:/opt/ml/input:ro'` - テスト入力ディレクトリをコンテナが読み取り専用で使用できるようにします。
- `--volume '<path_to_model>:/opt/ml/model'` - トレーニングテストが完了したら、モデルアーティファクトが保存されているパスをホストマシンでバインドマウントします。
- `--volume '<path_to_output>:/opt/ml/output'` - 障害理由が書き込まれる `failure` ファイル内のパスをホストマシンでバインドマウントします。
- `--name my-training-container` - 実行中のこのコンテナに名前を付けます。
- `my-training-image` - ビルドされたイメージを実行します。
- `train` - コンテナの実行時に同じスクリプト SageMaker 実行を実行します。

このコマンドを実行すると、Docker は、ビルドされたトレーニングイメージからコンテナを作成して実行します。コンテナは `train` スクリプトを実行します。これで、トレーニングプログラムが起動します。

トレーニングプログラムが終了し、コンテナが終了したら、出力モデルのアーティファクトが正しいことを確認します。さらに、ログ出力をチェックして、トレーニングジョブに関する十分な情報が提供されていることを確認し、併せて、不要なログが生成されていないことを確認します。

これで、アルゴリズム製品用のトレーニングコードのパッケージ化が完了しました。アルゴリズム製品には推論イメージも含まれるため、次のセクション、「[アルゴリズムの推論イメージの作成](#)」に進んでください。

アルゴリズムの推論イメージの作成

このセクションでは、推論コードをアルゴリズム製品の推論イメージにパッケージ化する手順を説明します。

推論イメージは、推論ロジックを含む Docker イメージです。実行時にコンテナは HTTP エンドポイントを公開し、SageMaker がコンテナとの間でデータを渡すことを許可します。

アルゴリズム製品を公開する際は、トレーニングイメージと推論イメージの両方が必要です。これをまだ確認していない場合は、「[アルゴリズム用のトレーニングイメージの作成](#)」に関する前のセクションを参照してください。2つのイメージは1つのイメージに結合することも、個別のイメージのままにしておくこともできます。イメージを結合するか個別のイメージのままにするかは、ユーザー次第です。通常、推論は学習よりも単純です。イメージを結合せずに個別のイメージのままにすると、推論のパフォーマンスが向上することがあります。

Note

以下は、推論イメージのパッケージコードの一例です。詳細については、「[で独自のアルゴリズムとモデルを使用する](#)」および [AWS Marketplace SageMaker 「」の例](#)を参照してください [GitHub](#)。

以下の例ではわかりやすくするために [Flask](#) というウェブサービスを使用していますが、本番環境に対応しているとは見なされていません。

ステップ

- [ステップ 1: 推論イメージを作成する](#)
- [ステップ 2: イメージをローカルでビルドしてテストする](#)

ステップ 1: 推論イメージを作成する

推論イメージをと互換性を持たせるには SageMaker、Docker イメージが HTTP エンドポイントを公開する必要があります。コンテナの実行中に、は購入者が提供する推論 SageMaker の入力をコンテナの HTTP エンドポイントに渡します。推論の結果は HTTP レスポンスの本文で返されます。

以下では、Linux の Ubuntu ディストリビューションの開発環境にインストールされた Docker CLI が使用されています。

- [ウェブサーバスクリプトを作成する](#)
- [コンテナ実行用のスクリプトを作成する](#)
- [Dockerfile の作成](#)
- [モデルアーティファクトを動的に読み込むためのプログラムを準備する](#)

ウェブサーバースクリプトを作成する

この例では [Flask](#) という Python サーバーを使用していますが、フレームワークに適した任意のウェブサーバーを使用できます。

Note

ここではわかりやすくするために [Flask](#) を使用しています。本番環境に対応するウェブサーバーとは見なされません。

SageMaker が使用する TCP ポート 8080 で 2 つの HTTP エンドポイントを提供する Flask ウェブサーバースクリプトを作成します。想定されるエンドポイントは次の 2 つです。

- `/ping` - このエンドポイントに HTTP GET リクエスト SageMaker を行い、コンテナの準備が整っているかどうかを確認します。コンテナの準備が完了すると、コンテナはこのエンドポイントでの HTTP GET リクエストに HTTP 200 レスポンスコードで応答します。
- `/invocations` - SageMaker 推論のためにこのエンドポイントに HTTP POST リクエストを行います。推論用の入力データはリクエストの本文で送信されます。ユーザー指定のコンテンツタイプは HTTP ヘッダーで渡されます。レスポンスの本文は推論出力です。

`./web_app_serve.py`

```
# Import modules
import json
import re
from flask import Flask
from flask import request
app = Flask(__name__)

# Create a path for health checks
@app.route("/ping")
def endpoint_ping():
    return ""

# Create a path for inference
@app.route("/invocations", methods=["POST"])
def endpoint_invocations():

    # Read the input
```

```
input_str = request.get_data().decode("utf8")

# Add your inference code here.
#
#
#
#
# Add your inference code here.

# Return a response with a prediction
response = {"prediction":"a","text":input_str}
return json.dumps(response)
```

前の例には、実際の推論ロジックはありません。実際の推論イメージについては、ウェブアプリに推論ロジックを追加し、入力を処理して予測を返します。

推論イメージはインターネットにアクセスできないため、必要な依存関係がすべて含まれている必要があります。

コンテナ実行用のスクリプトを作成する

Docker コンテナイメージ `serve SageMaker` の実行時に実行される という名前のスクリプトを作成します。このスクリプトでは、HTTP ウェブサーバーを起動します。

`./serve`

```
#!/bin/bash

# Run flask server on port 8080 for SageMaker
flask run --host 0.0.0.0 --port 8080
```

Dockerfile の作成

ビルドコンテキストに Dockerfile を作成します。この例では Ubuntu 18.04 を使用していますが、フレームワークに適していれば、どのベースイメージからでも開始できます。

`./Dockerfile`

```
FROM ubuntu:18.04
```

```
# Specify encoding
ENV LC_ALL=C.UTF-8
ENV LANG=C.UTF-8

# Install python-pip
RUN apt-get update \
&& apt-get install -y python3.6 python3-pip \
&& ln -s /usr/bin/python3.6 /usr/bin/python \
&& ln -s /usr/bin/pip3 /usr/bin/pip;

# Install flask server
RUN pip install -U Flask;

# Add a web server script to the image
# Set an environment to tell flask the script to run
COPY /web_app_serve.py /web_app_serve.py
ENV FLASK_APP=/web_app_serve.py

# Add a script that Amazon SageMaker will run
# Set run permissions
# Prepend program directory to $PATH
COPY /serve /opt/program/serve
RUN chmod 755 /opt/program/serve
ENV PATH=/opt/program:${PATH}
```

Dockerfile は以前に作成した 2 つのスクリプトをイメージに追加します。serve スクリプトのディレクトリが PATH に追加されると、コンテナの実行時にそれを実行できるようになります。

モデルアーティファクトを動的に読み込むためのプログラムを準備する

アルゴリズム製品の場合、購入者は独自のデータセットとトレーニングイメージを使用して、独自のモデルアーティファクトを生成します。トレーニングプロセスが完了すると、トレーニングコンテナはモデルアーティファクトをコンテナディレクトリに出力します /opt/ml/model/。はそのディレクトリ内のコンテンツを .tar.gz ファイルに SageMaker 圧縮し、Amazon S3 の購入者の AWS アカウントに保存します。

モデルがデプロイされ、推論イメージ SageMaker が実行され、Amazon S3 の購入者のアカウントに保存されている .tar.gz ファイルからモデルアーティファクトが抽出され、/opt/ml/model/ ディレクトリの推論コンテナにロードされます。実行時、推論コンテナコードはモデルデータを使用します。

Note

モデルアーティファクトファイルに含まれる可能性のある知的財産を保護するために、ファイル出力前の暗号化を選択できます。詳細については、「[セキュリティと知的財産](#)」を参照してください。

ステップ 2: イメージをローカルでビルドしてテストする

ビルドコンテキストには、現在、以下のファイルが存在します。

- ./Dockerfile
- ./web_app_serve.py
- ./serve

次に、このコンテナイメージをビルド、実行、テストできます。

イメージを構築する

Docker コマンドを実行し、イメージをビルドしてタグ付けします。この例ではタグ `my-inference-image` を使用します。

```
sudo docker build --tag my-inference-image ./
```

この Docker コマンドを実行してイメージをビルドすると、Dockerfile の各行に基づいて Docker がイメージをビルドするときの出力が表示されます。終了すると、次のようなものが表示されます。

```
Successfully built abcdef123456  
Successfully tagged my-inference-image:latest
```

をローカルで実行する

ビルドが完了したら、イメージをローカルでテストできます。

```
sudo docker run \  
  --rm \  
  --publish 8080:8080/tcp \  
  --volume ' <path_to_model>:/opt/ml/model:ro' \  
  --detach \  
<image_name>
```

```
--name my-inference-container \  
my-inference-image \  
serve
```

コマンドの詳細は次のとおりです。

- `--rm` - コンテナが停止したら自動的に削除します。
- `--publish 8080:8080/tcp` - ポート 8080 を公開して、ポートが HTTP リクエスト SageMaker を送信することをシミュレートします。
- `--volume '<path_to_model>:/opt/ml/model:ro'` - テストモデルアーティファクトが保存されているホストマシン上のパスを、コンテナ内の推論コードで使用できるように読み取り専用としてバインドマウントします。
- `--detach` - コンテナをバックグラウンドで実行します。
- `--name my-inference-container` - 実行中のこのコンテナに名前を付けます。
- `my-inference-image` - ビルドされたイメージを実行します。
- `serve` - コンテナの実行時に同じスクリプト SageMaker 実行を実行します。

このコマンドを実行すると、Docker は、推論イメージからコンテナを作成してバックグラウンドで実行します。コンテナは `serve` スクリプトを実行し、テスト目的でウェブサーバーを起動します。

HTTP エンドポイントへの ping をテストします。

がコンテナ SageMaker を実行すると、定期的にエンドポイントに ping を送信します。エンドポイントがステータスコード 200 の HTTP レスポンスを返すと、コンテナ SageMaker が推論できる状態であることを知らせます。

次のコマンドを実行してエンドポイントをテストし、レスポンスヘッダーを含めます。

```
curl --include http://127.0.0.1:8080/ping
```

以下の例に、出力例を示します。

```
HTTP/1.0 200 OK  
Content-Type: text/html; charset=utf-8  
Content-Length: 0  
Server: MyServer/0.16.0 Python/3.6.8  
Date: Mon, 21 Oct 2019 06:58:54 GMT
```

推論 HTTP エンドポイントをテストします。

コンテナが 200 ステータスコードを返すことで準備が完了したことを示すと、SageMaker は POST リクエストを介して推論データを HTTP /invocations エンドポイントに渡します。

以下のコマンドを実行して、推論エンドポイントをテストします。

```
curl \
  --request POST \
  --data "hello world" \
  http://127.0.0.1:8080/invocations
```

以下の例に、出力例を示します。

```
{"prediction": "a", "text": "hello world"}
```

これら 2 つの HTTP エンドポイントが機能するので、推論イメージは と互換性があるようになりました SageMaker。

Note

アルゴリズム製品のモデルは、リアルタイムとバッチの 2 つの方法でデプロイできます。どちらのデプロイでも、は Docker コンテナの実行中に同じ HTTP エンドポイント SageMaker を使用します。

コンテナを停止するには、次のコマンドを実行します。

```
sudo docker container stop my-inference-container
```

アルゴリズム製品のトレーニングイメージと推論イメージの両方の準備とテストが完了したら、「[イメージのアップロード](#)」に進みます。

イメージのアップロード

このセクションでは、推論イメージとトレーニングイメージを Amazon Elastic Container Registry にアップロードする手順を説明します。[Amazon ECR](#) は完全マネージド型の Docker レジストリ サービスです。これは、Amazon が からイメージを SageMaker プルして、推論用のモデルパッケージまたはトレーニングジョブ用のアルゴリズムを作成する場所です。これは、 がイメージ AWS Marketplace を取得してモデルパッケージとアルゴリズム製品を公開する場所でもあります。

アップロードする必要があるイメージ

モデルパッケージを公開する場合は、推論イメージのみをアップロードします。アルゴリズムを公開する場合は、推論イメージとトレーニングイメージの両方をアップロードします。推論イメージとトレーニングイメージを結合する場合は、結合されたイメージを 1 回だけアップロードしてください。

必要となる IAM アクセス許可

以下の手順では、ローカルマシンが販売者の AWS Identity and Access Management (IAM) ロールまたはユーザーの正しい AWS 認証情報を持っていることを前提としています AWS アカウント。ロールまたはユーザーは、AWS Marketplace と Amazon ECR の両方に正しいポリシーを設定する必要があります。例えば、次の AWS マネージドポリシーを使用できます。

- AWSMarketplaceSellerProductsFullAccess – へのアクセス用 AWS Marketplace
- AmazonEC2ContainerRegistryFullAccess - Amazon ECR へのアクセス用

Docker クライアントを にログインする AWS

発行元の の変数を設定します (AWS リージョン「」を参照[公開 AWS リージョン でサポート](#))。この例では、米国東部 (オハイオ) リージョンを使用します。

```
region=us-east-2
```

次のコマンドを実行して、AWS アカウント ID で変数を設定します。この例では、現在の AWS Command Line Interface (AWS CLI) 認証情報が販売者の に属していることを前提としています AWS アカウント。

```
account=$(aws sts get-caller-identity --query Account --output text)
```

リージョンの AWS アカウント Amazon ECR Docker レジストリを使用して Docker CLI クライアントを認証するには、次のコマンドを実行します。

```
aws ecr get-login-password \  
--region ${region} \  
| sudo docker login \  
--username AWS \  
--password-stdin \  

```

```
`${account}.dkr.ecr.${region}.amazonaws.com
```

リポジトリを作成してイメージをアップロードします。

アップロードしたイメージのタグに変数を設定し、アップロードしたイメージリポジトリの名前に別の変数を設定します。

```
image=my-inference-image  
repo=my-inference-image
```

Note

推論イメージとトレーニングイメージが構築されたこのガイドの前のセクションでは、`my-training-image`それぞれ `my-inference-image`と とタグ付けされました。この例では、推論イメージを作成して同じ名前のリポジトリにアップロードします。

次のコマンドを実行して、Amazon ECR にイメージリポジトリを作成します。

```
aws ecr --region ${region} create-repository --repository-name "${repo}"
```

Amazon ECR リポジトリの場所のフルネームは、次のパーツで構成されています。 `<account-id>.dkr.ecr.<region>.amazonaws.com/<image-repository-name>`

イメージをリポジトリにプッシュするには、リポジトリの場所のフルネームをイメージにタグ付ける必要があります。

イメージリポジトリの場所のフルネームとして変数を `latest` タグと共に設定します。

```
fullname="`${account}.dkr.ecr.${region}.amazonaws.com/${repo}:latest"
```

イメージにフルネームをタグ付けします。

```
sudo docker tag ${image} ${fullname}
```

最後に、Amazon ECR のリポジトリに推論イメージをプッシュします。

```
sudo docker push ${fullname}
```

アップロードが完了すると、公開元のリージョンの [Amazon ECR コンソールのリポジトリリスト](#) にイメージが表示されます。前の例では、イメージは米国東部 (オハイオ) リージョンにプッシュされました。

アップロードしたイメージをスキャンします。

[Amazon ECR コンソール](#) で、公開元の AWS リージョン を選択し、イメージがアップロードされたリポジトリを開きます。アップロードしたイメージを選択し、スキャンを開始して既知の脆弱性を確認します。は、Amazon SageMaker リソースで使用されるコンテナイメージの Amazon ECR スキャン結果を公開する前に AWS Marketplace チェックします。製品を作成するには、重要度が「重大」または「高」の脆弱性があるコンテナイメージを修正する必要があります。

イメージが正常にスキャンされたら、そのイメージを使用してモデルパッケージまたはアルゴリズムリソースを作成できます。

商品のスキャン中に誤検出のエラーがあったと思われる場合は、[AWS Marketplace Seller Operations](#) チームに連絡し、エラーに関する情報を伝えてください。

次のステップ

- [機械学習製品を作成するための要件とベストプラクティス](#) におけるファイルサイズ制限を確認する
- 「[Amazon SageMaker リソースの作成](#)」に進む

Amazon SageMaker リソースの作成

モデルパッケージまたはアルゴリズム製品を公開するには、Amazon でそれぞれの [モデルパッケージリソース](#) または [アルゴリズムリソース](#) を作成する必要があります SageMaker。

AWS Marketplace 製品のリソースを作成するときは、検証ステップを通じて認定される必要があります。検証ステップでは、公開前にモデルパッケージまたはアルゴリズムリソースをテストするためのデータを提供する必要があります。

Note

製品のイメージをまだ作成しておらず、Amazon Elastic Container Registry (Amazon ECR) にアップロードしていない場合は、その方法の情報について、「[コードをイメージにパッケージ化する](#)」および「[イメージのアップロード](#)」を参照してください。

モデルパッケージの作成

AWS Marketplaceのモデルパッケージを作成するための要件は次のとおりです。

- [Amazon ECR](#) に保存されている推論イメージ
- (オプション) [Amazon S3](#) に個別に保存されているモデルアーティファクト
- Amazon Simple Storage Service (Amazon S3) に保存されている、推論に使用されるテストデータ

Note

以下は、モデルパッケージ製品の作成に関するものです。のモデルパッケージの詳細については SageMaker、[「モデルパッケージリソースの作成」](#) を参照してください。

モデルパッケージリソースの作成

以下の手順では、モデルパッケージリソースを作成する方法を順を追って説明します。

ステップ 1: モデルパッケージリソースを作成する

1. [Amazon SageMaker コンソール](#) を開きます。
2. ページの右上を見て、発行元の AWS リージョンにいることを確認します。公開については、[「公開 AWS リージョン でサポート」](#) セクションを参照してください。前のステップで Amazon ECR にアップロードした推論イメージが同じリージョンに存在する必要があります。
3. 左側のナビゲーションメニューから [モデルパッケージ] を選択します。
4. [Create model package (モデルパッケージの作成)] を選択します。

パッケージを作成したら、推論パッケージの仕様を設定する必要があります。

ステップ 2: 推論の仕様を設定する

1. モデルパッケージの名前を指定します (例: *my-model-package*) 。
2. [推論イメージの場所] には、Amazon ECR にアップロードされた推論イメージの URI を入力します。URI は [Amazon ECR コンソール](#) でイメージを検索すると取得できます。
3. トレーニングで得たモデルアーティファクトが推論イメージのロジックにバンドルされている場合、モデルデータアーティファクトの場所は空欄のままにしておきます。それ以外の場合は、モデルアーティファクトの圧縮ファイル (.tar.gz) の、Amazon S3 の完全な場所を指定します。

4. ドロップダウンボックスを使用して、リアルタイム推論 (エンドポイントとも呼ばれる) ジョブとバッチ変換ジョブの両方でサポートされる推論イメージのインスタンスタイプを選択します。
5. [次へ] をクリックします。

モデルパッケージを作成して公開するには、期待どおりに機能するかを確認するための検証が必要です。そのためには、提供した推論用テストデータを使用してバッチ変換ジョブを実行する必要があります。検証仕様は、検証の実行 SageMaker 方法を指定します。

ステップ 3: 検証仕様を設定する

1. [AWS Marketplaceでこのモデルパッケージを公開する] を [はい] に設定します。これを [いいえ] に設定すると、このモデルパッケージを後で公開することはできません。Yes を選択すると、[このモデルパッケージが認証](#) AWS Marketplace され、検証ステップが必要です。
2. このプロセスを初めて完了する場合は、IAM ロールの [新規ロールの作成] を選択します。Amazon は、モデルパッケージをデプロイするときこのロール SageMaker を使用します。これには、Amazon ECR からのイメージの取得や Amazon S3 からのアーティファクトの取得などのアクションが含まれます。設定を確認し、ロールの作成 を選択します。ここでロールを作成すると、作成したロールに IAM [AmazonSageMakerFullAccess](#) ポリシーで説明されているアクセス許可が付与されます。
3. 検証プロファイルで JSON を編集します。許可される値の詳細については、「」を参照してください[TransformJobDefinition](#)。
 1. TransformInput.DataSource.S3Uri: 推論用のテストデータを保存する場所を設定します。
 2. TransformInput.ContentType: テストデータコンテンツタイプ (、 、 image/png 、 またはその他の値など) application/json text/plainを指定します。SageMaker は実際に入力データを検証しません。この値は、Content-type ヘッダー値としてコンテナの HTTP エンドポイントに渡されます。
 3. TransformInput.CompressionType: Amazon S3 の推論用テストデータが圧縮されていない場合は None に設定します。
 4. TransformInput.SplitType: None に設定して、Amazon S3 内の各オブジェクトを推論用にまとめて渡します。
 5. TransformOutput.S3OutputPath: 推論出力が保存される場所に設定します。
 6. TransformOutput.AssembleWith: None に設定して、各推論を Amazon S3 の個別のオブジェクトとして出力します。

4. [Create model package (モデルパッケージの作成)] を選択します。

SageMaker は Amazon ECR から推論イメージをプルし、アーティファクトを推論コンテナにコピーし、推論用のテストデータを使用してバッチ変換ジョブを実行します。検証が成功すると、ステータスは [完了済み] に変わります。

Note

検証ステップでは、テストデータによるモデルの精度は評価されません。検証ステップでは、コンテナが実行されて期待どおりに応答するかどうかを確認されます。

モデル製品リソースの作成が完了しました。「[AWS Marketplaceでの製品の公開](#)」に進みます。

アルゴリズムの作成

AWS Marketplaceのアルゴリズムを作成するための要件は次のとおりです。

- Amazon ECR に保存された推論イメージ
- Amazon ECR に保存されたトレーニングイメージ
- Amazon S3 に保存されたトレーニング用テストデータ
- Amazon S3 に保存された推論用テストデータ

Note

次のウォークスルーでは、アルゴリズム製品を作成します。詳細については、「[アルゴリズムリソースを作成する](#)」を参照してください。

アルゴリズムリソースの作成

以下の手順では、アルゴリズムパッケージのリソースを作成する方法を順を追って説明します。

ステップ 1: アルゴリズムリソースを作成する

1. [Amazon SageMaker コンソール](#) を開きます。

2. ページの右上を見て、公開元の AWS リージョンにいることを確認します (「」を参照[公開 AWS リージョンでサポート](#))。前のステップで Amazon ECR にアップロードしたトレーニングイメージと推論イメージが同じリージョンに存在する必要があります。
3. 左のナビゲーションペインで [アルゴリズム] をクリックします。
4. [Create algorithm (アルゴリズムの作成)] を選択します。

アルゴリズムパッケージを作成したら、モデルのトレーニングとチューニングの仕様を設定する必要があります。

ステップ 2: トレーニングとチューニングの仕様を設定する

1. アルゴリズムの[名前] (例えば *my-algorithm*) を入力します。
2. [トレーニングイメージ] には、Amazon ECR にアップロードされたトレーニングイメージの、完全な URI の場所を貼り付けます。URI は [Amazon ECR コンソール](#) でイメージを検索すると取得できます。
3. ドロップダウンボックスを使用して、トレーニングイメージがサポートするトレーニング用のインスタンスタイプを選択します。
4. [チャンネルの仕様]セクションで、アルゴリズムがサポートする各入力データセットに 1 つのチャンネルを追加します (入力ソースの最大数は 20 チャンネル)。詳細については、「[入力データ設定](#)」を参照してください。
5. [次へ] をクリックします。
6. アルゴリズムがハイパーパラメータとハイパーパラメータ調整をサポートしている場合は、調整パラメータを指定する必要があります。
7. [次へ] をクリックします。

Note

アルゴリズムがハイパーパラメータ調整をサポートして適切なパラメータを調整できるようになっていることが、強く推奨されます。これにより、データサイエンティストはモデルを調整して最良の結果を得ることができます。

調整パラメータがある場合、そのパラメータを設定したら、推論イメージの仕様を設定する必要があります。

ステップ 3: 推論イメージ仕様を設定する

1. [推論イメージの場所] には、Amazon ECR にアップロードされた推論イメージの URI を貼り付けます。URI は [Amazon ECR コンソール](#) でイメージを検索すると取得できます。
2. ドロップダウンボックスを使用して、リアルタイム推論 (エンドポイントとも呼ばれる) ジョブとバッチ変換ジョブの両方でサポートされる推論イメージのインスタンスタイプを選択します。
3. [次へ] をクリックします。

アルゴリズムを作成して公開するには、期待どおりに機能するかを確認するための検証が必要です。そのためには、トレーニング用のテストデータを使用するトレーニングジョブと、提供した推論用テストデータを使用してバッチ変換ジョブの両方を実行する必要があります。検証仕様は、検証の実行 SageMaker 方法を指定します。

ステップ 4: 検証仕様を設定する

1. [AWS Marketplaceでこのアルゴリズムを公開する] を [はい] に設定します。これを [いいえ] に設定すると、このアルゴリズムを後で公開することができません。Yes を選択すると、[のアルゴリズムが認証](#) AWS Marketplace され、検証仕様が必要です。
2. の機械学習パッケージを初めて作成する場合は AWS Marketplace、IAM ロール の新しいロールを作成する を選択します。Amazon は、アルゴリズムをトレーニングし、後続のモデルパッケージをデプロイするときにこのロール SageMaker を使用します。これには、Amazon ECR からのイメージの取得、Amazon S3 へのアーティファクトの保存、Amazon S3 からのトレーニングデータのコピーなどのアクションが含まれます。設定を確認し、ロールの作成 を選択します。ここでロールを作成すると、作成したロールに IAM [AmazonSageMakerFullAccess](#) ポリシーで説明されているアクセス許可が付与されます。
3. [トレーニングジョブ定義] の検証プロファイルの [JSON] ファイルを編集します。使用できる値の詳細については、[TrainingJobDefinition](#) 「」を参照してください。
 1. InputDataConfig: この JSON 配列に、トレーニング仕様のステップで指定した各チャンネルの [チャンネルオブジェクト](#) を追加します。チャンネルごとに、トレーニング用のテストデータを保存する場所を指定します。
 2. OutputDataConfig: トレーニングが完了すると、トレーニングコンテナディレクトリパス /opt/ml/model/ 内のモデルアーティファクトが圧縮され、Amazon S3 にコピーされます。圧縮ファイル (.tar.gz) が保存されている Amazon S3 の場所を指定します。
4. [変換ジョブの定義] の検証プロファイルの JSON ファイルを編集します。使用できる値の詳細については、[TransformJobDefinition](#) 「」を参照してください。

1. `TransformInput.DataSource.S3Uri`: 推論用のテストデータを保存する場所を設定します。
 2. `TransformInput.ContentType`: テストデータのコンテンツタイプを指定します。例えば、`application/json`、`text/plain`、`image/png`、またはその他の値です。Amazon SageMaker は実際の入力データを検証しません。この値は、`Content-type` ヘッダー値としてコンテナの HTTP エンドポイントに渡されます。
 3. `TransformInput.CompressionType`: Amazon S3 の推論用テストデータが圧縮されていない場合は `None` に設定します。
 4. `TransformInput.SplitType`: S3 内のオブジェクトをどのように分割するかを選択します。例えば、`None` では、Amazon S3 内の各オブジェクトが推論用にまとめて渡されます。詳細については、「Amazon SageMaker API リファレンス [SplitType](#)」の「」を参照してください。
 5. `TransformOutput.S3OutputPath`: 推論出力が保存される場所に設定します。
 6. `TransformOutput.AssembleWith`: `None` に設定して、各推論を Amazon S3 の個別のオブジェクトとして出力します。
5. [アルゴリズムパッケージの作成] を選択します。

SageMaker は Amazon ECR からトレーニングイメージをプルし、データを使用してテストトレーニングジョブを実行し、モデルアーティファクトを Amazon S3 に保存します。次に、Amazon ECR から推論イメージを取得して、アーティファクトを Amazon S3 から推論コンテナにコピーし、推論用のテストデータを使用してバッチ変換ジョブを実行します。検証が成功すると、ステータスは [完了済み] に変わります。

Note

検証ステップでは、テストデータによるトレーニングまたはモデルの精度は評価されません。検証ステップでは、コンテナが実行されて期待どおりに応答するかどうかを確認されません。

検証ステップではバッチ処理のみが検証されます。リアルタイム処理が製品で機能することを検証するのはユーザーだけです。

アルゴリズム製品リソースの作成が完了しました。「[AWS Marketplaceでの製品の公開](#)」に進みます。

AWS Marketplaceでの製品の公開

モデルパッケージまたはアルゴリズムを公開する前に、以下が必要です。

- AWS Marketplace 販売 AWS アカウント 者として登録されている。これは、[AWS Marketplace 管理ポータル](#)で行えます。
- AWS Marketplace 管理ポータルの [\[設定\]](#) ページの完了済みセラープロファイル。
- 有料製品を公開するには、Tax Interview と銀行フォームに必要事項を記入する必要があります。これは無料製品の公開には必要ありません。詳細については、「[Seller registration process](#)」を参照してください。
- AWS Marketplace 管理ポータル および Amazon にアクセスするには、アクセス許可が必要です SageMaker。詳細については、「[必要なアクセス許可](#)」を参照してください。

公開プロセスの概要

公開プロセスには次の 4 つのステップがあります。

1. 製品の送信 - モデルパッケージまたはアルゴリズム製品の説明、使用情報、その他の詳細を記載したリストを作成します。製品を公開用に提出してから、ステータスが次のステップに変わるまでに約 1 時間かかります。
2. テスト製品 - AWS Marketplace 販売者として登録 AWS アカウント されている を使用して、のリストをプレビューし AWS Marketplace、サブスクライブして、製品をテストします。さらに、許可されている他の AWS アカウント は、製品をプレビューおよびテストできます。変更が必要な場合は、戻ってリストの詳細を編集できます。
3. 公開のサインオフ - 製品を起動する準備ができたなら、[こちら](#)に戻り AWS Marketplace 管理ポータル、サインオフして [こちら](#)を公開を選択します。
4. 製品の公開開始 - 製品が AWS Marketplaceで公開されました。更新や製品修正を含む新しいバージョンを公開して、製品を維持できます。

必要なアクセス許可

Amazon SageMaker 製品を公開するには、サインインしている AWS Identity and Access Management ユーザーまたはロールに、次の IAM アクションの 1 つまたは両方が必要です。

- sagemaker:DescribeModelPackage - モデルパッケージを一覧表示する場合
- sagemaker:DescribeAlgorithm - アルゴリズムを一覧表示する場合

必要な AWS Marketplace アクセス許可、または販売者アカウントの管理については、[AWS Marketplace 「販売者のポリシーとアクセス許可」](#)を参照してください。

製品リストの作成

以下は、モデルパッケージとアルゴリズム製品の両方 AWS Marketplace について、で製品リストを作成するためのチュートリアルです。

Note

リストを作成する前に、「[機械学習製品を作成するための要件とベストプラクティス](#)」で指定されている必要なリソースがあることを確認してください。

このプロセスには以下のステップがあります。

ステップ

- [ステップ 1: 新しいリストを作成する](#)
- [ステップ 2: 一般的な製品情報を提供する](#)
- [ステップ 3: 起動オプションを追加する](#)
- [ステップ 4: 価格と条件を設定する](#)
- [ステップ 5: 製品を公開用に送信する](#)

ステップ 1: 新しいリストを作成する

新しい機械学習製品リストを作成するには

1. 販売者にサインイン AWS アカウントし、に移動します[AWS Marketplace 管理ポータル](#)。
2. トップメニューで [製品]、[機械学習] の順に移動します。
3. [新しいリストを作成] を選択します。

Note

[新しい製品] ページの [製品概要] セクションでは、現在のステータス、プライバシー設定、製品タイプ、作成者、製品 ID を確認できます。

ステップ 2: 一般的な製品情報を提供する

一般的な製品情報を提供するには

1. [一般的な製品情報] セクションの [製品の説明] で、[追加] を選択します。
 - a. [製品の可視性] セクションでは、以下のいずれかのオプションを選択します。
 - パブリック – 製品は、最初はテスト AWS アカウント 用に限定された のセットで利用できます。サインオフして公開すると、製品は一般に公開され、すべてのお客様がサブスクライブできるようになります。
 - プライベート – 製品は、AWS アカウント 指定した にのみ表示されます。今後、この製品を公開することはできません。
 - b. [製品のタイトル]、[簡単な製品の説明]、[製品の概要]、[製品カテゴリ 1]、およびその他の詳細を入力します。これらの値は、後で変更できます。製品の説明については、「[機械学習製品を作成するための要件とベストプラクティス](#)」を参照してください。
 - c. 完了したら [続行] を選択します。
2. [プロモーションリソース] には、製品ロゴ、検索キーワード、関連するリソースリンクを入力します。これらの値は、後で変更できます。
 - 完了したら [続行] を選択します。
3. [サポート情報] で、製品のサポートを提供するかどうかを選択します。
 - a. [はい] を選択した場合は、サポートと連絡先の詳細を入力してください。これらの値は後で変更できます。
 - b. 完了したら [続行] を選択します。
4. リージョンの可用性 AWS リージョン で、製品を一覧表示する特定の を選択します。

デフォルト値は、[現在および将来サポートされるすべてのリージョンで利用可能にする] です。

 - 完了したら [続行] を選択します。

Note

公開用に下書きを送信した後は、この選択を変更することはできません。

製品を公開する次のステップでは、販売するモデルパッケージまたはアルゴリズムである起動オプションを提供します。

ステップ 3: 起動オプションを追加する

起動オプションを追加するには

1. [起動オプション] セクションの [ARN を入力] に、モデルパッケージまたはアルゴリズムの Amazon リソースネーム (ARN) を入力します。

ARN は、Amazon SageMaker コンソールの [モデルパッケージ](#) または [アルゴリズム](#) ページにあります。

Example モデルパッケージの ARN

```
arn:aws:sagemaker:<region>:<account-id>:model-package/<model-package-name>
```

Example アルゴリズムの ARN

```
arn:aws:sagemaker:<region>:<account-id>:algorithm/<algorithm-name>
```

2. [追加] を選択します。
3. 以下の手順は、モデルパッケージを公開するか、アルゴリズム製品を公開するかによって異なります。購入者向けのバージョン番号を除き、バージョンの詳細は後で変更できます。
 1. ステップ 1: バージョンの詳細と Git リポジトリリンク を入力し、サンプル Jupyter Notebook と GitHub リポジトリのバージョン番号、リリースノート、URLs を指定します。
 2. アルゴリズム製品の場合のみ、[ステップ 2: トレーニングデータ入力の詳細の入力] で、トレーニングデータの説明、トレーニングアルゴリズムの概要の他、トレーニングデータリソースの例も記載してください。

アルゴリズムメトリクス、チャンネル仕様、ハイパーパラメータは、でアルゴリズムリソースを作成したときに指定した値に基づいて、製品詳細ページに自動的に表示されます SageMaker。

以下の例は、セラーにはトレーニングデータ入力の詳細がどのように表示されるか、購入者にはトレーニングデータ入力の詳細がどのように表示されるかを示しています。

Example トレーニングデータ入力の例 - セラービュー

Enter details describing the training data inputs

Information to train a model

Describing the training data including an example training data resource along with an overview of training algorithm. Algorithm Metrics, Channel specification and Hyperparameters will be automatically displayed on the product detail page based on details in your algorithm container image See examples [here](#). Note, if you have any details related to trained model input and outputs, please remove from here and enter in Steps 3 and 4.

Example input(s) for training job:
[Bank Marketing Dataset from UCI](<https://archive.ics.uci.edu/ml/datasets/bank+marketing>)
 AutoGluon-Tabular requires no manual data preprocessing as long as your data is a valid CSV table
 Rows in your table represent different examples (data points), columns represent different variables (features).
 The first line of your CSV file should contain names for each column.
 Columns in your CSV file can be strings/text-fields/Numeric.
 Your data must contain the column that you identify as 'label' in your hyperparameter configuration.

572 of 600 characters used

Example トレーニングデータ入力の例 - 購入者ビュー

Usage Information

Training

Example input(s) for training job:
[Bank Marketing Dataset from UCI](#) [↗](#)
 AutoGluon-Tabular requires no manual data preprocessing as long as your data is a valid CSV table
 Rows in your table represent different examples (data points), columns represent different variables (features).
 The first line of your CSV file should contain names for each column.
 Columns in your CSV file can be strings/text-fields/Numeric.
 Your data must contain the column that you identify as 'label' in your hyperparameter configuration.

[↕ Show less](#)

次の例は、カスタム属性 (呼び出しパラメータ) がセラーにどのように表示されるか、カスタム属性 (呼び出しパラメータ) が購入者にどのように表示されるかを示しています。

Example カスタム属性 (呼び出しパラメータ) の例 - セラービュー

Custom attributes (invocation parameters) - optional Remove

Names
Short label for parameter data, can be comma delimited list

Maximum 100 characters

Description of parameter data
Brief summary of the parameter field

55 of 500 characters used

Parameter data type
Select one

Minimum Value - optional Maximum 9 characters **Maximum Value - optional** Maximum 9 characters

Is this parameter variable always required?
 Yes
 No, it's optional (must enter a default value)

Default Value

Maximum 200 characters

Example カスタム属性 (呼び出しパラメータ) の例 - 購入者ビュー

▼ Custom attributes (invocation parameters)

Field name
threshold

Description
Threshold of the confidence score of detected objects

Data type Continuous (Float)	Range Min: 0.0 Max: 1.0	Required No
---------------------------------	-------------------------------	----------------

Default value
0.3

3. ステップ 3: 入力の詳細の入力では、モデルまたはアルゴリズムの入力詳細とサンプル入力ファイルの URL を指定します。

以下の例は、セラーにはモデルデータ入力の詳細がどのように表示されるか、購入者にはモデルデータ入力の詳細がどのように表示されるかを示しています。

Example モデルデータ入力の例 - セラービュー

Model input details
Help customers understand your model capabilities by providing details related to model input (summary, limitations, mime types, and sample data for realtime and batch invocation) and input data descriptions (required for text/csv and application/json mime types). See examples [here](#).

Model input summary
Describe the model input format specification in text.

B I H | </> | ≡ | ≡ | % | 🔍 | 🗑️

This model can analyze images that are supplied as image bytes or stored in an Amazon S3 bucket.

96 of 500 characters used

Limitation for input type - optional
Define any limitations on the input data, such as file size and/or image aspect ratio

The minimum size is 80 pixels for both height and width. The image can be no larger than 1024X1024 pixels, otherwise the detection performance may degrade dramatically. Note, that images with aspect ratio close to 1.0 are best.

227 of 300 characters used

Input mime types
Select all the input data formats permitted

- Select multiple -

image/bmp X image/png X image/jpeg X application/x-image X

Choose to provide raw text or URL for realtime sample input data

URL
 Text

Realtime: Sample input data

Maximum 150 characters

Batch job: Sample input data
Provide URL to a folder that shows an example of a batch input that supports multiple records

Maximum 150 characters

Example モデルデータ入力の例 - 購入者ビュー

Input

Summary

This model can analyze images that are supplied as image bytes or stored in an Amazon S3 bucket.

Limitations for input type

The minimum size is 80 pixels for both height and width. The image can be no larger than 1024X1024 pixels, otherwise the detection performance may degrade dramatically. Note, that images with aspect ratio close to 1.0 are best.

Input MIME type

image/bmp, image/png, image/jpeg

Sample input data

[view data](#)

- [ステップ 4: 出力の詳細の入力] では、モデルまたはアルゴリズムの出力詳細とサンプル出力をテキストまたは URL として指定します。

使用に関する情報については、「[機械学習製品を作成するための要件とベストプラクティス](#)」を参照してください。

次の例は、モデルデータ出力の詳細がセラーにどのように表示されるか、モデルデータ出力の詳細が購入者にどのように表示されるかを示しています。

Example モデルデータ出力の例 - セラービュー

Model output details

Help customers understand your model capabilities by providing details related to model output (summary, limitations, mime types, and sample output data for realtime and batch invocation) and output data descriptions (required for text/csv and application/json mime types). See examples [here](#).

Model output summary

Describe the model output format specification in text.

B I H | </> | ≡ | ≡ | % | 🗑️ | ⓘ

The model detects instances of common objects such as computer, in an image. The response includes an array of detected object labels (id field) with bounding box pixel coordinates, and an associated level of confidence.

220 of 500 characters used

Limitation for output type - optional

Define any limitations on the output data

Maximum file size is...

0 of 300 characters used

Output mime types

Select one or more output formats

- Select multiple - ▾

application/json ×

Choose to provide raw text or URL for realtime sample output data

URL

Text

Realtime: Sample output data

Ensure the sample output data corresponds to the input sample data

```
[{"right":603,"bottom":528,"top":177,"score":0.9921523332595825,"id":"person","left":439},
{"right":687,"bottom":539,"top":184,"score":0.9885265231132507,"id":"person","left":577},
{"right":611,"bottom":246,"top":228,"score":0.21156705915927887,"id":"cell phone","left":598}]
```

273 of 1000 characters used

Batch job: Sample output data

Provide a link to a file or folder and ensure example data corresponds to the input sample data

<https://github.com/zhrshold/gluoncv-sagemaker-examples/blob/master/example/input/playground.jpg>

Maximum 150 characters

Example モデルデータ出力の例 - 購入者ビュー

Output

Summary

The model detects instances of common objects such as computer, in an image. The response includes an array of detected object labels (id field) with bounding box pixel coordinates, and an associated level of confidence.

Output MIME type

application/json

Sample output data

```
[
  {
    "right": 603,
    "bottom": 528,
    "top": 177,
    "score": 0.9921523332595825,
    "id": "person",
    "left": 439
  },
  {
    "right": 687,
    "bottom": 539,
    "top": 184,
    "score": 0.9885265231132507,
    "id": "person",
    "left": 577
  },
  {
    "right": 611,
    "bottom": 246,
    "top": 228,
    "score": 0.21156705915927887,
    "id": "cell phone",
    "left": 598
  }
]
```

5. [ステップ 5: サポートされているインスタンスを確認して作成する] では、推奨インスタンスを設定します。

- これがモデルパッケージ製品の場合は、バッチ変換とリアルタイムデプロイの両方で、サポートされているインスタンスから推奨インスタンスタイプを選択してください。
- アルゴリズム製品の場合は、推奨インスタンスタイプトレーニングジョブも選択してください。

モデルパッケージやアルゴリズムリソースがサポートしていないインスタンスタイプは選択できません。サポートされているインスタンスタイプは、Amazon でこれらのリソースを作成したときに選択されました SageMaker。

4. 完了したら [続行] を選択します。

Note

購入者の満足度を高めるには、製品の期待される入力と出力を説明した明確な使用情報 (例を含む) が不可欠です。詳細については、「[機械学習製品を作成するための要件とベストプラクティス](#)」を参照してください。

製品を公開する次のステップでは、価格と条件を設定します。

ステップ 4: 価格と条件を設定する

価格と条件を設定するには

1. [価格と条件] セクションで、[オファーを追加] を選択します。
2. [料金表] を設定します。

ソフトウェアを無料で提供したり、有料価格を設定したり、無料試用期間を有効にしたりできます。詳細については、「[機械学習製品の料金](#)」を参照してください。

3. エンドユーザーライセンス契約 (EULA) として使用するプレーンテキストファイルをアップロードします。
4. [保存して閉じる] を選択します。

製品に関するすべての情報を入力しました。次のステップでは、製品をテストできるように、限定販売で公開します。

ステップ 5: 製品を公開用に送信する

製品を公開用に送信するには

1. 「新製品」ページの「公開用に送信」セクションの「追加のテストアカウント – オプション」で AWS アカウント IDs を入力します。
2. [公開用に送信] を選択します。

これにより、ユーザー (およびオプションのテスター) AWS Marketplace がサブスクライブしてテストに使用できるプレビューリストを に作成することで、公開プロセスが開始されます。

これで、製品をテストする準備が整いました。機械学習製品のテストの詳細については、「[製品のテスト](#)」を参照してください。

製品をテストした後、変更が必要な場合は上記の手順をやり直しできます。購入者に製品を提供する準備が整ったら、[公開を承認](#)できます。

製品のテスト

製品を最初に送信してから、プレビューリストの準備が整うまでに約 1 時間かかります。ステータスがテスト製品 に変わった後、販売者アカウントおよびその他の許可リストに登録されている AWS アカウント は、 のリストをプレビューし AWS Marketplace、製品をサブスクライブしてテストできます。

出品情報のプレビューを見るには

1. で AWS Marketplace 管理ポータル、製品概要ページに移動します。
2. [ステージングされた製品に移動] を選択します。
3. 変更を加える場合は、[製品の編集] を選択し、「[製品リストの作成](#)」と同じ手順に従ってください。
4. すべての購入者が閲覧できるように製品を公開する準備が整ったら、「[公開のサインオフ](#)」の手順に従ってください。

公開前に製品をテスト AWS アカウント する他の を追加するには、[AWS Marketplace Seller Operations チーム](#)に連絡して AWS アカウント IDsを指定します。許可リストに登録されているアカウントには、製品詳細ページの製品バージョンの横に Limited バッジが表示されます。

公開のサインオフ

このステップは、説明、価格、使用情報を記入し、製品をテストした後に行います。

公開をサインオフするには

1. 販売者にサインイン AWS アカウント し、 に移動します [AWS Marketplace 管理ポータル](#)。
2. トップメニューで [製品]、[機械学習] の順に移動します。
3. 製品の [製品の概要] に移動します。
4. [サインオフして公開] を選択します。

製品の更新

の [Machine Learning Listings](#) ページを使用して AWS Marketplace 管理ポータル、次の方法でモデルパッケージまたはアルゴリズム製品を更新できます。

- [新しいバージョンの追加](#) - 新しいモデルパッケージまたはアルゴリズムリソースを既存の製品の新しいバージョンとして追加できます。
- [バージョンの制限](#) - 既存の製品の以前のバージョンを制限できます。
- [製品の削除](#) - 製品全体を削除できます。

新しいバージョンを追加する

モデルパッケージまたはアルゴリズムリソースの新しいバージョンを追加するには

1. AWS Marketplace 管理ポータルの「[Machine Learning Listings](#)」ページに移動します。
2. 既存の製品の [製品の概要] に移動します。
3. [製品の編集] を選択します。
4. [起動オプション] で [編集] を選択します。
5. リソースの ARN を追加するには、[バージョン] ページに移動し、[新しいバージョンを追加] を選択します。

起動オプションの追加方法の詳細については、「[製品リストの作成](#)」を参照してください。

Note

使用情報は製品バージョンごとに異なります。新しいバージョンに使用情報を追加する場合は、引き続き「[機械学習製品を作成するための要件とベストプラクティス](#)」に従ってください。

購入者が AWS Marketplace リストから製品を起動すると、異なるバージョンを選択できます。購入者が Amazon SageMaker コンソールから製品を起動すると、最新バージョンのみが表示されます。

バージョンを制限する

モデルパッケージまたはアルゴリズムリソースのバージョンを制限するには

1. 既存の製品の [製品の概要] に移動します。
2. [製品の編集] を選択します。
3. [起動オプション] で [編集] を選択します。
4. [バージョン] ページで [バージョンを制限] を選択します。
5. [製品の概要] に戻り、[公開用に送信] を選択します。

Note

既に製品をサブスクライブしている購入者は、制限付きバージョンのモデルパッケージまたはアルゴリズムを引き続き使用できます。ただし、新規購入者にはそれらの制限付きバージョンをオプションとして表示することはできません。

製品を削除する

製品を削除するには

1. AWS Marketplace 管理ポータル「[Machine Learning Listings](#)」ページにある公開製品のリストに移動します。
2. 削除する製品を選択し、[アクション] ドロップダウンリストで [リストを非公開にする] を選択します。
3. リクエストに関して AWS Marketplace 担当者から連絡があった場合に備えて、E メールアドレスと出品を削除する理由を記入してください。

Note

から製品を削除すると AWS Marketplace、新しい購入者は製品をサブスクライブできなくなります。ただし、既存の購入者は製品を使用し続けられ、これは最低 90 日サポートされる必要があります。非公開の出品情報を別の製品に置き換える予定がある場合は、削除リクエストの詳細に新しい出品情報を明記してください。

機械学習製品を作成するための要件とベストプラクティス

モデルパッケージとアルゴリズム製品を購入者が簡単にテストできるかを確認することが重要です。以下のセクションでは、機械学習 (ML) 製品リストを作成するための要件と ML 製品のベストプラクティスについて説明します。要件と推奨事項の完全な要約については、「[ML 製品リストの要件と推奨事項の要約](#)」を参照してください。

Note

公開された製品が要件を満たしていない場合、これらの要件を満たすために の AWS Marketplace 担当者から連絡されることがあります。

トピック

- [必須アセット](#)
- [ML 製品の一般的なベストプラクティス](#)
- [使用状況情報の要件](#)
- [入力と出力の要件](#)
- [Jupyter Notebook の要件](#)
- [ML 製品リストの要件と推奨事項の要約](#)

必須アセット

機械学習製品リストを作成する前に、次の必須アセットがあることを確認してください。

- Amazon リソースネーム (ARN) — 公開元の でモデルパッケージまたはアルゴリズムリソースの ARN AWS リージョン を指定します (「」を参照 [公開 AWS リージョン でサポート](#)) 。

- モデルパッケージの ARN の形式は `arn:aws:sagemaker:<region>:<account-id>:model-package/<model-package-name>` です。
- アルゴリズムの ARN の形式は `arn:aws:sagemaker:<region>:<account-id>:algorithm/<algorithm-name>` です。
- [the section called “使用状況情報の要件”](#) - 入力、出力、およびコード例に関する詳細を提供します。
- [the section called “入力と出力の要件”](#) - ファイルまたはテキストのいずれかを提供します。
- [the section called “Jupyter Notebook の要件”](#) - 製品の使用方法をすべてデモンストレーションします。

ML 製品の一般的なベストプラクティス

使用している機械学習製品について、以下の情報を提供します。

- 製品の説明には、以下を含めます。
 - モデルの機能
 - ターゲットとなる顧客
 - 最も重要なユースケース
 - モデルのトレーニング方法、または使用されたデータの量
 - パフォーマンスメトリクスと、使用された検証データ
 - 医療機器の場合、モデルが診断用かどうか
- デフォルトでは、機械学習製品は一般に公開されるよう設定されています。ただし、非公開で製品を作成することもできます。詳細については、「[製品リストの作成](#)」を参照してください。
- (オプション) 有料製品の場合は、顧客に対して 14～30 日間の無料試用期間を提供し、製品を試してもらいます。詳細については、「[機械学習製品の料金](#)」を参照してください。
- (オプション) モデルパッケージ製品について、商品一覧ページでリアルタイムの製品デモを有効にする場合は、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。製品デモでは、見込み客が自分でモデルをサブスクライブしたりデプロイしたりしなくても、出品ページでモデルを直接試すことができます。

使用状況情報の要件

購入者の満足度を高めるには、製品の期待される入力と出力を説明した明確な使用情報 (例を含む) が不可欠です。

新しいバージョンのリソースを製品リストに追加するたびに、使用状況情報を提供する必要があります。

初めて公開する新製品の使用情報を追加するには、AWS Marketplace 管理ポータル コンソールにサインインします。[製品] ドロップダウンから [機械学習] を選択します。製品を選択します。[製品概要] の [起動オプション] で、モデルパッケージまたはアルゴリズムリソースの ARN を入力し、[追加] を選択します。

特定のバージョンの既存の使用状況情報を編集するには、[起動オプション] の [編集] を選択して、[バージョンを編集] を選択します。

入力と出力の要件

バイヤーが製品を理解して使用できるように、入力と出力の例を含め、形式を明確に説明することが重要です。製品を理解すると、購入者は、入力データに対して必要な変換を行って最適な推論結果を得られるようになります。

Amazon SageMaker リソースを製品リストに追加すると、次のプロンプトが表示されます。

推論の入力と出力

推論入力では、リアルタイムエンドポイントとバッチ変換ジョブの両方の入力形式を指定します。データの前処理に必要なコードスニペットを含めてください。サポートされている MIME コンテンツタイプ (例えば、image/jpeg、image/png、image/bmp)、値の説明 (該当する場合)、および制限事項を含めてください。でホストされている入力サンプルを含めます [GitHub](#)。

推論出力では、リアルタイムエンドポイントとバッチ変換ジョブの両方の出力形式を指定します。出力 MIME コンテンツタイプ (例えば、application/json、image/jpeg) と値の説明 (該当する場合) を含めてください。でホストされている出力サンプルを含めます [GitHub](#)。

サンプルには、お使いの製品に対応する入力ファイルを提供してください。モデルが多クラス分類を実行する場合は、クラスごとに少なくとも1つのサンプル入力ファイルを提供してください。

トレーニング入力

「モデルをトレーニングするための情報」セクションに、データの前処理に必要な入力データ形式とコードスニペットを指定します。サポートされている MIME コンテンツタイプ (例えば、image/jpeg、image/png、image/bmp)、値の説明 (該当する場合)、および制限事項を含めてください。でホストされている入力サンプルを必ず含めてください [GitHub](#)。

購入者が提供できるオプション機能と必須機能の両方を説明し、PIPE 入力モードがサポートされているかどうかを明記します。[分散型トレーニング](#) (複数の CPU/GPU インスタンスによるトレーニング) がサポートされている場合は、その旨を明記してください。チューニングについては、推奨されるハイパーパラメータを一覧で表示してください。

Jupyter Notebook の要件

SageMaker リソースを製品リストに追加するときは、 でホストされているサンプル Jupyter Notebook へのリンクを提供し[GitHub](#)、購入者にデータのアップロードや検索を依頼することなく、ワークフロー全体を示します。

AWS SDK for Python (Boto)を使用します。優れたサンプルノートブックを開発して提供できれば、購入者はリストを簡単に試して使用することができます。

モデルパッケージ製品の場合、サンプルノートブックには、入力データの準備、リアルタイム推論用エンドポイントの作成、バッチ変換ジョブの実行などが示されています。詳細については、「[のモデルパッケージのリストとサンプルノートブック](#)」を参照してください GitHub。サンプルノートブックについては、「[generic_sample_notebook](#)」と「[auto_insurance](#)」を参照してください。後者のサンプルノートブックは、パラメータを入力したり AWS リージョン、購入者がサンプルデータを検索したりすることなく、すべてので動作します。

Note

未開発のサンプル Jupyter Notebook ノートブックで、考えられる入力やデータの前処理手順が複数表示されていないと、購入者が製品の価値提案を完全に理解するのが難しい場合があります。

アルゴリズム製品の場合、サンプルノートブックは、完全なトレーニング、チューニング、モデル作成、リアルタイム推論用のエンドポイントの作成、バッチ変換ジョブのパフォーマンスを示しています (「」の「[アルゴリズムのリストとサンプルノートブック](#)」を参照 GitHub)。サンプルノートブックについては、「」の「[amazon_demo_product and automl](#)」を参照してください GitHub。これらのサンプルノートブックはすべてのリージョンで機能し、パラメータを入力したり、購入者がサンプルデータを探したりする必要はありません。

Note

サンプルのトレーニングデータが不足していると、購入者が Jupyter Notebook を正常に実行できなくなります。サンプルノートブックが開発されていないと、購入者が製品を使用できなくなり、採用が見合わされるおそれがあります。

ML 製品リストの要件と推奨事項の要約

次の表に、機械学習製品の出品ページの要件と推奨事項の概要を示します。

詳細	モデルパッケージリスト用	アルゴリズムリスト用
Product descriptions		
サポートされているコンテンツタイプ (例: 「イメージ内で X を検出」) に対して製品が何をするかを詳しく説明します。	必須	必須
製品に関する説得力のある差別化情報を提供します (「最良」や根拠のないクレームなどの形容詞は避けてください)。	推奨	推奨
この製品の最も重要なユースケース (複数可) を一覧表示します。	必須	必須
トレーニングされたデータ (ソースとサイズ) を記述し、既知の制限があればリストします。	必須	該当しない

詳細	モデルパッケージリスト用	アルゴリズムリスト用
モデルが構築されたコアフレームワークについて説明します。	推奨	推奨
検証データに関するモデルパフォーマンスメトリクスを要約します (例: 「Z データセットを使用してベンチマークされた」XX.YY パーセントの精度)。	必須	該当しない
推奨インスタンスタイプのモデルレイテンシーやスループットメトリクスを要約します。	必須	該当しない
アルゴリズムカテゴリを記述します。例えば、この決定フォレスト回帰アルゴリズムは、ブートストラップ集約の一般的な手法とランダムな特徴の選択を使用して構築された木構造分類子のアンサンブルに基づいています。	該当しない	必須
Usage information		

詳細	モデルパッケージリスト用	アルゴリズムリスト用
<p>推論の場合は、リアルタイムエンドポイントジョブとバッチ変換ジョブの両方の入力形式を指定します。サポートされている MIME コンテンツタイプ (例えば、image/jpeg、image/png、image/bmp)、値の説明 (該当する場合)、および制限事項を含めてください。入力と出力の要件 を参照してください。</p>	必須	必須
<p>推論の場合は、リアルタイムエンドポイントジョブとバッチ変換ジョブの両方の入力サンプルを指定します。サンプルは デモ でホストする必要があります。GitHub。入力と出力の要件 を参照してください。</p>	必須	必須
<p>推論の場合は、リアルタイムエンドポイントジョブとバッチ変換ジョブの両方の出力形式を指定します。出力 MIME コンテンツタイプ (例えば、application/json、image/jpeg) と値の説明 (該当する場合) を含めてください。入力と出力の要件 を参照してください。</p>	必須	必須

詳細	モデルパッケージリスト用	アルゴリズムリスト用
推論には、リアルタイムエンドポイントジョブとバッチ変換ジョブの両方の出力サンプルを指定します。サンプルはでホストする必要があります GitHub。 入力と出力の要件 を参照してください。	必須	必須
推論には、エンドポイントまたはバッチ変換ジョブを使用する例を指定します。 AWS Command Line Interface (AWS CLI) コマンドまたは AWS SDK を使用してコード例を含めます。	必須	必須
トレーニングには、入力形式を指定します。サポートされている MIME コンテンツタイプ (例: image/jpeg 、 image/png 、 image/bmp)、該当する場合は値の説明、制限 (例: 必要なデータの最小行) を含めます。 入力と出力の要件 を参照してください。	該当しない	必須
トレーニングでは、 でホストされている入力サンプルを提供します GitHub。 入力と出力の要件 を参照してください。	該当しない	必須

詳細	モデルパッケージリスト用	アルゴリズムリスト用
<p>トレーニングには、トレーニングジョブの実行例を指定します。サポートされているハイパーパラメータ、その範囲、および全体的な影響について説明します。アルゴリズムがハイパーパラメータチューニング、分散トレーニング、または GPU インスタンスをサポートしているかどうかを指定します。AWS CLI コマンドや AWS SDK の使用などのコード例を含めます。</p>	該当しない	必須
<p>製品の完全な使用 GitHub を示す でホストされている Jupyter Notebook を提供します。 Jupyter Notebook の要件 を参照してください。</p>	必須	必須
<p>ユーザーマニュアルやサンプルデータなど、製品の使用に関連する技術情報を提供します。</p>	推奨	推奨

サービスの制限とクォータ

このセクションでは、AWS Marketplaceの機械学習 (ML) 製品の制限とクォータについて説明します。

ネットワークの隔離

セキュリティ上の理由から、コンテナ化された製品を購入者がサブスクライブすると、Docker コンテナはネットワークアクセスのない隔離された環境で実行されます。コンテナの作成時には、イン

ターネット経由での呼び出しに依存しないでください。その呼び出しは失敗します。への呼び出しも失敗 AWS のサービスします。

イメージのサイズ

Docker イメージのサイズは、Amazon Elastic Container Registry (Amazon ECR) の [Service Quotas](#) によって管理されます。Docker イメージのサイズは、トレーニングジョブ、バッチ変換ジョブ、エンドポイント作成時の起動時間に影響します。パフォーマンスを向上させるため、最適な Docker イメージのサイズを維持してください。

ストレージサイズ

エンドポイントを作成すると、Amazon はエンドポイントをホストする各 ML コンピューティングインスタンスに Amazon Elastic Block Store (Amazon EBS) ストレージボリュームをア SageMaker タッチします。(エンドポイントは、リアルタイム推論または Amazon SageMaker ホスティングサービスとも呼ばれます)。ストレージボリュームのサイズは、インスタンスタイプによって異なります。詳細については、「Amazon SageMaker デベロッパーガイド」の [「ホストインスタンスストレージボリューム」](#) を参照してください。

バッチ変換については、「Amazon SageMaker デベロッパーガイド」の [「バッチ変換でのストレージ」](#) を参照してください。

インスタンスサイズ

SageMaker は、さまざまな ML ユースケースに合わせて最適化されたインスタンスタイプの選択を提供します。インスタンスタイプは、CPU、GPU、メモリ、およびネットワーク容量のさまざまな組み合わせで構成されます。インスタンスタイプでは、ML モデルの構築、トレーニング、デプロイに適したリソースの組み合わせを柔軟に選択できます。詳細については、[「Amazon SageMaker ML インスタンスタイプ」](#) を参照してください。

推論のペイロードサイズ

エンドポイントの場合、1 回の呼び出しあたりの入力データの最大サイズが 6 MB に制限されます。この値を調整することはできません。

バッチ変換の場合、1 回の呼び出しあたりの入力データの最大サイズは 100 MB です。この値を調整することはできません。

推論の処理時間

エンドポイントの場合、1回の呼び出しあたりの最大処理時間は 60 秒です。この値を調整することはできません。

バッチ変換の場合、1回の呼び出しあたりの最大処理時間は 60 分です。この値を調整することはできません。

Service Quotas

トレーニングと推論に関連するクォータの詳細については、[「Amazon SageMaker Service Quotas」](#) を参照してください。

非同期推論

で発行されたモデルパッケージとアルゴリズムは、[Amazon SageMaker 非同期推論](#) 用に設定されたエンドポイントにデプロイ AWS Marketplace できません。非同期推論用に設定されたエンドポイントの場合、モデルにネットワーク接続が必要です。すべての AWS Marketplace モデルはネットワーク分離で動作します。詳細については、[「No network access」](#) を参照してください。

サーバーレス推論

で発行されたモデルパッケージとアルゴリズムは、[Amazon SageMaker Serverless Inference](#) 用に設定されたエンドポイントにデプロイ AWS Marketplace できません。サーバーレス推論用に設定されたエンドポイントの場合、モデルにネットワーク接続が必要です。すべての AWS Marketplace モデルはネットワーク分離で動作します。詳細については、[「No network access」](#) を参照してください。

マネージドスポットトレーニング

のすべてのアルゴリズムについて AWS Marketplace、[マネージドスポットトレーニング](#) のチェックポイントが実装されている場合でも、 の値は 3,600 秒 (60 分) にMaxWaitTimeInSeconds設定されます。この値を調整することはできません。

Docker イメージと AWS アカウント

公開するには、イメージを AWS アカウント 販売者の が所有する Amazon ECR リポジトリに保存する必要があります。別の が所有するリポジトリに保存されているイメージを公開することはできません AWS アカウント。

組み込みアルゴリズムまたはからのモデルパッケージの公開 AWS Marketplace

[Amazon SageMaker 組み込みアルゴリズム](#)または AWS Marketplace サブスクリプションのアルゴリズムを使用してトレーニングジョブから作成されたモデルパッケージは公開できません。

トレーニングジョブのモデルアーティファクトは引き続き使用できますが、モデルパッケージを公開するには独自の推論イメージが必要です。

公開 AWS リージョン でサポート

AWS Marketplace は、以下の両方 AWS リージョン が当てはまるからのモデルパッケージとアルゴリズムリソースの公開をサポートします。

- [Amazon SageMaker がサポート](#)するリージョン
- デフォルトでオプトインされている[利用可能なリージョン](#) (例えば、[describe-regions](#) が返す "OptInStatus": "opt-in-not-required" など)

モデルパッケージまたはアルゴリズム製品の公開に必要なすべてのアセットは、公開元と同じリージョンに保存する必要があります。これには以下が含まれます。

- Amazon で作成されるモデルパッケージとアルゴリズムのリソース SageMaker
- Amazon ECR リポジトリにアップロードされる推論イメージとトレーニングイメージ
- Amazon Simple Storage Service (Amazon S3) に保存され、モデルパッケージリソースのモデルデプロイ中に動的にロードされるモデルアーティファクト (存在する場合)
- Amazon S3 に保存されている推論およびトレーニング検証用のテストデータ

でサポートされている任意のリージョンで製品を開発およびトレーニングできます SageMaker。ただし、公開する前に、AWS Marketplace が発行元をサポートしているリージョンにすべてのアセットをコピーして、そのリージョンでリソースを再作成する必要があります。

公開 AWS リージョン 元のに関係なく、リストプロセス中に公開先のリージョンを選択し、製品を使用可能にすることができます。

トラブルシューティング

このセクションでは、機械学習製品の公開プロセス中に発生する可能性のある一般的なエラーについて説明します。該当する問題がリストにない場合は、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。

一般: モデルパッケージまたはアルゴリズムの Amazon リソースネーム (ARN) を に追加すると、400 エラーが発生します。AWS Marketplace 管理ポータル

Amazon SageMaker コンソールを使用してリソースを作成した場合は、 でこのモデルパッケージを公開するプロセスの最終ページで「はい」、または でこのアルゴリズムを公開 AWS Marketplace するには「はい」を選択する必要があります。AWS Marketplace[いいえ]を選択すると、後で公開することはできません。[はい]を選択しても、モデルパッケージやアルゴリズムは公開されません。ただし、モデルパッケージまたはアルゴリズムリソースは作成時に検証されるため、AWS Marketplaceでの使用に必要です。

AWS SDK を使用して [モデルパッケージを作成したり、アルゴリズムを作成したりする場合は](#)、パラメータ `CertifyForMarketplace` が に設定されていることを確認します `true`。

認定および検証済みのモデルパッケージまたはアルゴリズムリソースを再作成したら、新しい ARN を AWS Marketplace 管理ポータルに追加します。

一般: モデルパッケージまたはアルゴリズムの ARN を に追加すると 404 エラーが発生する AWS Marketplace 管理ポータル

このエラーは、さまざまな理由で発生します。

- ARN が無効の可能性がありま。正しい ARN を使用していることを確認します。
 - モデルパッケージの場合、ARN は `arn:aws:sagemaker:us-east-2:000123456789:model-package/my-model-package-name` に似ています。
 - アルゴリズムの場合、ARN は `arn:aws:sagemaker:us-east-2:000123456789:algorithm/my-algorithm` に似ています。
- モデルパッケージまたはアルゴリズムリソースが販売者アカウント AWS アカウント と同じに作成されていません。公開するリソースとアセットがすべて、公開元の販売者アカウントにあることを確認してください。
- 公開に使用するユーザーまたはロールに、モデルパッケージまたはアルゴリズムリソースにアクセスするための適切な IAM 権限がありません。ユーザーまたはロールに、次のアクセス許可があることを確認してください。

- モデルパッケージの場合、モデルパッケージリソースに対するアクション `sagemaker:DescribeModelPackage` が許可されている必要があります。
- アルゴリズムの場合は、アルゴリズムリソースに対するアクション `sagemaker:DescribeAlgorithm` が許可されている必要があります。

一般: でアルゴリズム製品の料金を指定すると 500 エラーが発生する AWS Marketplace 管理ポータル

このエラーは、トレーニングイメージのみで、それに付随する推論イメージがないアルゴリズムリソースを公開しようとした場合に発生することがあります。で公開されるアルゴリズムリソースには、両方のコンポーネント AWS Marketplace が必要です。詳細については、「[で製品を準備する SageMaker](#)」を参照してください。

Amazon SageMaker: モデルパッケージまたはアルゴリズムリソースを作成すると、「クライアントエラー: レジストリへのアクセスが拒否されました」という失敗メッセージが表示されます

このエラーは、モデルパッケージまたはアルゴリズムの作成に使用されているイメージが、別の AWS アカウントに属する [Amazon ECR](#) リポジトリに保存されている場合に発生することがあります。モデルパッケージまたはアルゴリズムの検証は、クロスアカウントイメージをサポートしていません。公開に AWS アカウント 使用している が所有する Amazon ECR リポジトリにイメージをコピーします。次に、新しいイメージロケーションを使用してリソースを作成します。

Amazon SageMaker: 「開始されていません」と「クライアントエラー: スキャンがスケジュールされていません」と表示されます。モデルパッケージまたはアルゴリズムリソースを作成すると、失敗するメッセージが表示される

このエラーは、 が Amazon ECR に保存されている Docker コンテナイメージのスキャンを開始 SageMaker できない場合に発生する可能性があります。この場合は、[Amazon ECR コンソール](#)を開き、イメージがアップロードされた先のリポジトリを見つけてイメージを選択し、[スキャン]を選択してください。

レポート作成

AWS Marketplace は、購入者、財務、使用状況、税金に関するデータを含む Amazon SageMaker 製品のレポートを生成します。すべてのレポートは、[レポートページ](#) AWS Marketplace 管理ポータルので利用できます。詳細については、「[販売者レポート](#)」を参照してください。

以下のサブセクションでは、 を使用する機械学習製品の財務 SageMaker 情報がどのように報告されるかをまとめています。

日別ビジネスレポート

日次ビジネスレポートには、各購入者と製品のインスタンスタイプ、使用時間、ソフトウェア料金による収益、その他の詳細が記載されています。購入者は匿名の一意のカスタマーリファレンス ID によって識別されます。詳細については、「[日次ビジネスレポート](#)」を参照してください。

月別収益レポート

月次収益レポートには、ソフトウェアの使用に対して購入者に請求された月次収益が表示されます。詳細については、「[月別請求済み収益レポート](#)」を参照してください。

支払いレポート

月次支払いレポートには、ソフトウェア料金の決済期間中にお客様に代わって収集された総額の内訳が表示されます。レポートに反映される決済総額は、お客様の銀行口座に入金される金額と一致する必要があります。詳細については、「[支払いレポート](#)」を参照してください。

その他のレポートと分析

利用可能なその他のレポートについては、「[販売者レポート](#)」を参照してください。

AWS Marketplaceから利用可能な [販売者配送データフィードサービス](#) を使用してカスタムレポートを作成することもできます。

SaaS ベースの製品

Software as a Service (SaaS) 製品では、インフラストラクチャで AWS ホストされているソフトウェアをデプロイし、購入者に AWS 環境内のソフトウェアへのアクセスを許可します。販売者は、ソフトウェア内での顧客のアクセスやアカウント作成、リソースのプロビジョニング、アカウント管理まで行います。

API を Amazon API Gateway と統合する方法については、「Amazon API Gateway デベロッパーガイド」の「[AWS Marketplace で API Gateway API を販売する](#)」を参照してください。

SaaS 製品についてサポートが必要な場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

トピック

- [SaaS 製品を開始する](#)
- [SaaS 製品の計画](#)
- [SaaS 製品ガイドライン](#)
- [SaaS 製品の料金](#)
- [SaaS 無料トライアル](#)
- [SaaS の顧客のオンボーディング](#)
- [SaaS 製品に関する Amazon SNS 通知](#)
- [AWS Marketplace Metering Service API および使用権限管理サービス API へのアクセス](#)
- [レポート作成](#)
- [SaaS 製品統合のコード例](#)
- [PrivateLink での AWS の使用 AWS Marketplace](#)

SaaS 製品を開始する

この章では、販売者が Software as a Service (SaaS) 製品を作成および管理する方法について説明します。このセクションでは、SaaS 製品を AWS Marketplace に導入する方法について説明します。また、SaaS 製品の請求モデルに基づいて、SaaS 製品を適切な AWS Marketplace API オペレーションと統合する方法についても説明します。

前提条件

開始するには、以下の前提条件を満たす必要があります。

1. [AWS Marketplace 管理ポータル](#) にアクセスして使用する。これは、販売者として登録し、で販売する製品を管理するために使用するツールです AWS Marketplace。詳細については、「[AWS Marketplace 管理ポータル](#)」を参照してください。
2. 販売者として登録し、税金および銀行の情報を提出します。詳細については、「[販売者登録プロセス](#)」を参照してください。
3. AWS Marketplaceで SaaS 製品を作成および統合する方法を計画する。詳細については、「[SaaS 製品の計画](#)」を参照してください。

トピック

- [SaaS 製品のライフサイクル](#)
- [SaaS 製品の作成](#)
- [初期 SaaS 製品ページを作成する](#)
- [SaaS 製品設定](#)
- [SaaS サブスクリプション製品の統合](#)
- [SaaS 契約製品を統合する](#)
- [SaaS 契約を pay-as-you-go 製品と統合する](#)
- [サーバーレス SaaS 統合ソリューションのデプロイ](#)

SaaS 製品のライフサイクル

で SaaS 製品を作成すると AWS Marketplace、最初は可視性が制限されて公開され、アカウントのみがアクセスできるようになります。準備ができたら、それを AWS Marketplace カタログに公開して、購入者が製品をサブスクライブして購入できるようにすることができます。

SaaS 製品ページでは、製品のリストを表示できます。製品ライフサイクルの段階に応じて、製品のステータスは以下のいずれかになります。

- ステージング - 情報を追加中の、不完全な製品。セルフサービスエクスペリエンスの最初の保存して終了で、変更リクエストが成功すると、送信した完了したステップの情報を含む未公開の製品が作成されます。このステータスから、引き続き商品に情報を追加したり、変更リクエストを通じて送信済みの情報を変更したりできます。

- 制限あり - 製品がシステムに送信され、システム内のすべての検証に合格すると完成します。その後、製品は制限あり状態にリリースされます。この時点で、製品にはアカウントと許可リストに登録したユーザーだけがアクセスできる詳細ページがあります。商品は詳細ページからテストできません。詳細またはヘルプについては、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。
- パブリック - 購入者が商品を閲覧してサブスクライブできるように商品を公開する準備ができたなら、可視性を更新する変更リクエストを使用します。このリクエストは、AWS Marketplace Seller Operations チームが AWS ポリシーに照らして製品を確認および監査するためのワークフローを開始します。製品が承認され、変更リクエストが処理されると、製品のステータスが制限ありからパブリックに移行します。AWS ガイドラインの詳細については、「[SaaS 製品ガイドライン](#)」を参照してください。
- 制限付き - 新規ユーザーによる製品のサブスクライブを停止する場合は、可視性の更新変更リクエストを使用して製品を制限できます。制限付き ステータスは、既存のユーザーが引き続き製品を使用できることを意味します。ただし、製品は一般には公開されなくなり、新規ユーザーも利用できなくなります。

ステージング、制限あり、パブリックの各ステータスで製品を更新できます。詳細については、「[製品情報の更新](#)」を参照してください。

SaaS 製品の作成

で Software as a Service (SaaS) 製品を販売するには AWS Marketplace、以下のタスクを完了する必要があります。

1. で SaaS 製品を作成します AWS Marketplace。
2. SaaS [サブスクリプション](#)、[契約](#)、または [契約](#) を [pay-as-you-go](#) 製品と統合します AWS Marketplace。
3. [サブスクリプション](#)、[契約](#)、または 製品と の統合との [契約](#) [pay-as-you-go](#) をテストします AWS Marketplace。
4. 製品を発売用に送信します。

トピック

- [セルフサービスを使用して SaaS 製品を作成する](#)
- [SaaS 製品を作成する \(レガシー\)](#)

セルフサービスを使用して SaaS 製品を作成する

セルフサービス AWS Marketplace を使用して で SaaS 製品を作成するには

1. SaaS 製品の出品を決定する

で販売する SaaS 製品がある AWS Marketplace。 [SaaS 製品の計画](#) する方法を確認し、理解します。

2. 価格とオファータイプを決定する

SaaS 製品には、サブスクリプション、契約、との契約の 3 つのオファータイプがあります pay-as-you-go。オファータイプの選択は、SaaS 製品を と統合する方法に影響します AWS Marketplace。詳細については、「[料金の計画](#)」を参照してください。

3. アセットを収集する

製品の提出に必要なアセットを収集します。製品のアセットには以下が含まれます。

- 製品ロゴ URL - 提供する製品の鮮明なロゴイメージが含まれている公開されている Amazon S3 URL です。
- エンドユーザー使用許諾契約 (EULA) URL - 製品には PDF ファイル形式で入手できる EULA が必要です。顧客が製品の AWS Marketplace ページで EULA を確認できる Amazon S3 バケットへのリンクを提供する必要があります。
- 製品登録 URL - これは、購入者が製品を AWS Marketplace で正常にサブスクライブした後にリダイレクトされる URL です。
- 製品に関するメタデータ - AWS Marketplace 管理ポータルでの製品作成ウィザードでメタデータを入力します。
- 製品のサポート情報 - この情報には、製品のサポートチャネルの E メールアドレスと URL が含まれます。

4. 製品を統合に送信する

[初期 SaaS 製品ページを作成する](#) を使用して販売者アカウントから。AWS Marketplace は製品を限定製品として公開します AWS Marketplace 管理ポータル。つまり、統合とテストにのみ使用できます。製品コードと、Amazon Simple Notification Service (SNS) トピックは、製品概要ページに表示されます。

Note

お客様と AWS Marketplace Seller Operations チームは、高額なコストをかけずに製品をテストできるように、製品を割引価格で維持する必要があります。製品の公開をリクエストする際に、製品の実際の料金を確認します。

5. との統合 AWS Marketplace

製品は、アクセスを許可する前のサブスクリプションの検証や、場合によっては使用量の測定を含め、顧客が製品をオンボードして使用できるようにする必要があります。との統合方法は、製品に使用しているオファertype AWS Marketplace によって異なります。オファertype に基づく統合の詳細については、以下のトピックを参照してください。

- [サブスクリプション統合](#)
- [契約統合](#)
- [pay-as-you-go 統合との契約](#)

製品を と統合する最後のステップ AWS Marketplace は、統合が正しく機能することを確認するためにテストすることです。

6. 製品を発売用に送信する

統合を確認し、製品を公開する準備ができたなら、[可視性を更新] を選択します。AWS Marketplace Seller Operations チームが製品を審査して価格を更新してから、表示対象を [公開] に更新します。

Note

製品の検証と公開は手動プロセスで行われ、AWS Marketplace Seller Operations チームが行います。エラーがない場合は、最初の製品リストのバージョンを公開するまでに 7 ~ 10 営業日かかることがあります。タイミングについては、「[タイミングと心構え](#)」を参照してください。

SaaS 製品を作成する (レガシー)

(AWS Marketplace レガシー) で SaaS 製品を作成するには

1. SaaS 製品の出品を決定する

で販売する SaaS 製品がある AWS Marketplace。 [SaaS 製品の計画](#) する方法を確認し、理解します。

2. 料金モデルを決定する

SaaS 製品には、サブスクリプション、契約、との契約の 3 つの料金モデルがあります pay-as-you-go。 オファertypeの選択は、SaaS 製品を と統合する方法に影響します AWS Marketplace。 詳細については、「 [料金の計画](#) 」を参照してください。

3. アセットを収集する

製品の提出に必要なアセットを収集します。 製品のアセットには以下が含まれます。

- 製品ロゴ URL - 提供する製品の鮮明なロゴイメージが含まれている公開されている URL です。
- エンドユーザーライセンス契約 (EULA) URL - 製品には EULA が必要です。 また、お客様が製品の AWS Marketplace ページで読み、確認できるように、EULA へのリンクを提供する必要があります。
- 製品登録 URL - この URL は、顧客が AWS Marketplace で製品を正常にサブスクライブした後にリダイレクトされる場所です。
- 製品に関するメタデータ - AWS Marketplace 管理ポータルで製品作成ウィザードでメタデータを入力します。
- 製品のサポート情報 - これには、製品のサポートチャネルの E メールアドレスと URL が含まれます。

4. 製品を統合に送信する

[初期 SaaS 製品ページを作成する](#) を使用して販売者アカウントから。 AWS Marketplace は製品を限定製品として公開します AWS Marketplace 管理ポータル。 つまり、統合とテストに使用できるのはアカウントのみです。 Seller AWS Marketplace Operations チームは、製品コード、Amazon Simple Notification Service (Amazon SNS) トピック、製品ページの URL が記載された E メールメッセージを送信します。 この情報により、製品 AWS Marketplace でとの統合の作成とテストに使用する環境が完成します。 Seller AWS Marketplace Operations チームから受け取った E メールメッセージを使用して、製品に関する通信を行います。

Note

お客様と AWS Marketplace Seller Operations チームは、高額なコストをかけずに製品をテストできるように、製品を割引価格で維持する必要があります。製品の公開をリクエストする際に、製品の実際の料金を確認します。

5. との統合 AWS Marketplace

製品は、アクセスを許可する前のサブスクリプションの検証や、場合によっては使用量の測定を含め、顧客が製品をオンボードして使用できるようにする必要があります。との統合方法は、製品に使用しているオファertype AWS Marketplace によって異なります。オファertype に基づく統合の詳細については、以下のトピックを参照してください。

- [サブスクリプション統合](#)
- [契約統合](#)
- [pay-as-you-go 統合との契約](#)

製品を と統合する最後のステップ AWS Marketplace は、統合が正しく機能することを確認するためにテストすることです。

6. 製品を発売用に送信する

統合を検証し、製品を稼働させる準備ができたなら、end-to-end テストと起動のために AWS Marketplace Seller Operations チームに送信します (前に作成した E メールケースを使用)。

7. 作成する

end-to-end テストが完了したら、元の料金で製品ページを確認する必要があります。製品を作成したときに受け取った E メールケースに返信して、ページを承認します (「[SaaS 製品の作成](#)」を参照)。承認後、AWS Marketplace Seller Operations チームは製品ページを で公開します AWS Marketplace。この時点で、顧客は製品の検索とサブスクライブを行えるようになります。

初期 SaaS 製品ページを作成する

Software as a Service (SaaS) アプリケーションメタデータを使用し、 を使用して AWS Marketplace カタログに初期 SaaS 製品を作成します AWS Marketplace 管理ポータル。

初期 SaaS 製品ページを作成するには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. [Products (製品)] で [SaaS] を選択します。

Note

のいずれかで SaaS リストを作成する場合は AWS GovCloud (US) Regions、[???](#)プロセスを使用します。

3. [SaaS 製品の作成] を選択し、[SaaS 製品] を選択します。
4. SaaS 製品 ID とコードを生成します。タグベースの承認をサポートするオプションのタグを追加することもできます。

Note

タグベースの認証の詳細については、「ユーザーガイド」の「[タグを使用した AWS リソースへのアクセスの制御](#)」を参照してください。

5. セルフサービスエクスペリエンスを使用して AWS Marketplace リストを作成します。製品情報、製品デプロイの詳細、および公開オファーの詳細を追加します。オプションで、許可リストにアカウントを追加して製品をテストすることもできます。

Note

ステップを完了する前にセッションを終了する必要がある場合は、[保存して終了] オプションを選択して、現在の選択内容をステージング領域に保存します。このオプションにより、入力した情報の検証を求めるリクエストが作成されます。リクエストが検証されている間は、製品を編集することはできません。リクエストが成功すると、[製品の作成を再開する] を選択して製品の作成を続行できます。

リクエストが成功しなかった場合は、製品リクエストログに表示される検証エラーが原因です。リクエストを選択してエラーを表示し、[アクション] の [新規にコピー] を選択してエラーを修正し、リクエストを再送信できます。前の手順を更新するには、製品詳細ページを開いて変更リクエストを送信します。

Note

テスト中は、デフォルトで1ディメンションあたり 0.01 USD に設定されます。この料金設定により、高額な請求を発生させることなく、制限あり状態で製品をテストできます。実際の料金は、製品を公開する際に提示します。

6. [送信] を選択します。次に、 は情報 AWS Marketplace を検証します。検証が成功すると、 は製品を Limited ステータスで AWS Marketplace リリースします。検証が成功すると、製品をプレビュー、統合、テストできます。

Note

検証が処理されている間は、製品を編集することはできません。製品が最初に公開されると、製品の作成 AWS アカウント に使用した と AWS Marketplace Seller Operations チームのテストアカウントのみがアクセスできます。SaaS 製品ページから製品を表示する場合は、 で表示 AWS Marketplace を選択して、AWS Marketplace 購入者に表示される製品の詳細を表示できます。この詳細リストは他の AWS Marketplace ユーザーには表示されません。

SaaS 製品設定

で [Software as a Service \(SaaS\) 製品を作成](#)したら AWS Marketplace、必要に応じて製品の設定の多くを変更できます。詳細については、以下のトピックを参照してください。

トピック

- [変更リクエストを管理する](#)
- [製品情報を更新する](#)
- [AWS アカウント IDsの許可リストを更新する](#)
- [製品の可視性を更新する](#)
- [料金条件を更新する](#)
- [料金設定ディメンションを追加する](#)
- [料金設定ディメンションを更新する](#)
- [料金ディメンションを制限する](#)

- [購入者が製品にアクセスする方法を決定する](#)
- [国別の可用性を更新する](#)
- [製品の返金ポリシーを更新する](#)
- [エンドユーザー使用許諾契約書 \(EULA\) を更新する](#)

変更リクエストを管理する

[セルフサービスリスティング](#)では、変更リクエストを使用して製品に変更を加えます。現在のリクエストは、リクエストタブ [AWS Marketplace 管理ポータル](#) の にあります。ナビゲーションバーの下にある [変更をリクエスト] ドロップダウンリストから新しいリクエストを行うことができます。

SaaS 製品の変更リクエストを作成するには

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [製品] タブのドロップダウンリストから [SaaS] を選択します。
3. リクエストが送信されると、処理が開始されます。変更リクエストは、[レビュー中]、[変更を準備中]、[変更を適用中] のステータスになります。
4. リクエストの処理が完了すると、そのステータスは以下のいずれかの値に変わります。
 - 成功 - このステータスは、リクエストされた変更が処理され、変更がシステムに反映されたことを示します。
 - 失敗 - このステータスは、リクエストに何か問題が生じたため、変更が処理されなかったことを示します。ステータスが、[失敗] の場合は、リクエストを選択して、問題の修正方法に関する推奨事項を示すエラーコードを検索できます。エラーのトラブルシューティングを行い、変更の新しいリクエストを作成できます。処理を速めるには、[新しいリクエストにコピー] 機能を使用して失敗したリクエストの詳細をコピーできます。必要な変更を加えてリクエストを再送信できます。

更新で始まる変更リクエストには、プロジェクトの現在の詳細が読み込まれます。その後、更新を行うと、既存の詳細が上書きされます。リクエストの追加と制限のペアは、リクエストが成功するたびに (セルフサービスエクスペリエンスで [保存して終了] と [送信] アクションを選択した後) プロビジョニングされる更新専用です。つまり、既存のサブスクライバーは、サブスクリプションまたは契約が終了するまで製品を使い続けることができます。ただし、[制限付き] ステータスの製品には新しいサブスクライバーを追加することはできません。

製品情報を更新する

製品を作成した後に、AWS Marketplaceでその製品に関連する情報を変更する必要がある場合があります。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) ページの [\[SaaS 製品\]](#) タブで、変更する製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[製品情報を更新\]](#) を選択します。
4. 変更する次のフィールドのいずれかを更新します。

- [\[製品のタイトル\]](#)
- [\[SKU\]](#)
- [短い説明](#)
- [\[詳しい説明\]](#)
- [製品ロゴの URL](#)
- [\[ハイライト\]](#)
- [\[製品カテゴリ\]](#)
- [キーワード](#)
- [\[製品動画 URL\]](#)
- [リソース](#)
- [\[サポート情報\]](#)

Note

ロゴの形式について詳しくは、「[会社と製品のロゴ要件](#)」を参照してください。

5. 製品情報を更新するには、[\[送信\]](#) を選択します。
6. リクエストが [\[リクエスト\]](#) タブに [\[レビュー中\]](#) ステータスで表示されていることを確認します。新しいリクエストが表示されるようにするには、ページの更新が必要になる場合があります。

AWS アカウント IDsの許可リストを更新する

製品を限定された状態で表示できる AWS アカウント IDs のリストを変更できます。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) ページの [\[SaaS 製品\]](#) タブで、変更する製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンリストから [\[許可リストを更新\]](#) を選択します。リストには、現在許可リストに登録されている AWS アカウント IDs が表示されます。
4. [\[許可リストに登録された AWS アカウント\]](#) フィールドに AWS アカウント ID を入力し、カンマで区切ります。
5. AWS アカウント IDs の許可リストを更新するには、[送信](#) を選択します。

製品の可視性を更新する

でクイック起動エクスペリエンスを表示できる購入者を変更するには AWS Marketplace、可視性の更新 を使用できます。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) ページから、変更する製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンから [\[可視性を更新\]](#) を選択します。

Note

この変更リクエストを使用して、製品を [\[制限あり\]](#) ステータスから [\[パブリック\]](#) ステータスに移行するようリクエストできます。ただし、変更リクエストをパブリックに移行するには、AWS Marketplace Seller Operations チーム承認プロセスを実行する必要があります。

4. 公開するときは、商品の実際の価格を入力します。この価格は、リストが一般に公開されることが承認された後に適用されます。
5. リクエストをレビューに送信するには、[\[送信\]](#) を選択します。
6. [\[リクエスト\]](#) タブの [\[リクエストステータス\]](#) が [\[レビュー中\]](#) と表示されていることを確認します。リクエストが完了すると、ステータスが [\[成功\]](#) になります。

料金条件を更新する

SaaS 製品の料金をディメンションごとに変更する場合は、[\[料金条件を更新\]](#) を使用できます。

Note

いずれかのディメンションの料金を引き上げると、少なくとも今後 90 日間は料金更新オプションを利用できなくなります。値下げと値上げの両方を更新する場合は、まず値下げを更新してください。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. **[SaaS 製品]** ページの **[SaaS 製品]** タブで、変更する製品を選択します。
3. **[変更をリクエスト]** ドロップダウンリストから **[パブリックオファーを更新]** を選択し、**[料金条件を更新]** を選択します。
4. フィールドには現在の価格があらかじめ入力されています。現在の料金を削除して、新しい料金を追加できます。
5. リクエストをレビューに送信するには、**[送信]** を選択します。
6. **[リクエスト]** タブの **[リクエストステータス]** が **[レビュー中]** と表示されていることを確認します。リクエストが完了すると、ステータスが **[成功]** または **[失敗]** に更新されます。

料金設定ディメンションを追加する

製品の請求に使用するディメンションを追加できます。ディメンションは、製品を使用する際に購入者に請求される基本的な測定単位です。

Note

既存の料金設定ディメンションの名前または説明を更新するには、「[the section called “料金設定ディメンションを更新する”](#)」を参照してください。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. **[SaaS 製品]** タブから、変更する製品を選択します。
3. **[変更をリクエスト]** ドロップダウンから **[料金設定ディメンションを更新]** を選択し、**[料金設定ディメンションを追加]** を選択します。
4. ディメンション API ID、表示名、説明を入力して製品に新しいディメンションを追加し、**[次へ]** を選択します。

Note

API ID と名前は、すべてのディメンションにわたって一意である必要があります。ディメンションの作成後に API ID と単位を変更することはできません。

5. 追加した各ディメンションの料金を定義し、[次へ] を選択して変更を確認します。

Note

ディメンションを追加できるのは、製品用に選択した料金モデル (例えば、契約、使用量、SaaS 契約 (追加使用量あり) など) のみです。限定製品の場合、新しく追加されたディメンションの価格は 0.01 USD に設定されます。商品が一般に公開されるようになったら、料金を更新できます。

6. [送信] を選択して、リクエストを送信します。
7. [リクエスト] タブで、リクエストステータスが [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] に変わります。

料金設定ディメンションを更新する

製品の請求に使用するディメンションを更新できます。ディメンションは、製品を使用する際に購入者に請求される基本的な測定単位です。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) タブから、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [料金設定ディメンションを更新] を選択し、[ディメンション情報を更新] を選択します。
4. 更新するディメンションを検索し、名前または説明を選択します。
5. 新しい名前または説明を入力し、チェックマークを選択して更新を確定します。

Note

ディメンション名は一意である必要があります。

6. [送信] を選択して、リクエストを送信します。

7. [リクエスト] タブで、リクエストステータスが [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] に変わります。

料金ディメンションを制限する

製品に現在掲載されているディメンションを制限できます。このリクエストにより、選択したディメンションが製品から削除されます。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) タブから、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [料金設定ディメンションを更新] を選択し、[料金設定ディメンションを制限] を選択します。
4. 限定商品や公開商品については、[\[お問い合わせ\]](#) ボタンを使用して AWS Marketplace Seller Operations チームに連絡するよう求められます。フォームを使用して、製品リストから削除するディメンションの詳細を入力します。

購入者が製品にアクセスする方法を決定する

顧客が製品にアクセスする方法として、以下のいずれかのオプションを選択できます。

- [the section called “SaaS URL フルフィルメントオプションを更新する”](#) — 顧客は、AWS Marketplaceで製品をサブスクライブした後にリダイレクトされるサイトの URL を使用します。
- [the section called “クイック起動を設定する”](#) — 顧客は、簡略化されたプロセスを利用して製品を設定して起動できます。この設定は、限定的な可視性または公開された可視性の、既存の製品に対して行うことができます。

SaaS URL フルフィルメントオプションを更新する

SaaS 製品のフルフィルメントに使用される URL を更新するには、[フルフィルメントオプションを更新] を使用します。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) ページの [SaaS 製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンリストから [パブリックオファーを更新] を選択し、[デフォルトのフルフィルメント URL を更新] を選択します。

4. [フルフィルメント URL] フィールドに、SaaS 製品のフルフィルメントオプションの新しい URL を入力します。
5. リクエストをレビューに送信するには、[送信] を選択します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] または [失敗] に更新されます。

クイック起動を設定する

にリストされている SaaS 製品 AWS Marketplace では、多くの場合、サブスクライブしている購入者のアカウント (IAM ロールなど) に AWS リソースをデプロイする必要があります。クイック起動を使用すると、AWS CloudFormation テンプレートを使用して、購入者にガイド付き step-by-step、指示、リソースデプロイを提供できます。購入者は CloudFormation テンプレートを使用して製品を設定および起動します。

Tip

クイック起動設定プロセスの詳細については、[SaaS クイック起動を有効にする](#) ラボを参照してください。

顧客が SaaS 製品を起動するために使用できるクイック起動エクスペリエンスを設定するには、[フルフィルメントオプション] タブを使用します。

1. [\[SaaS 製品\]](#) ページの [SaaS 製品] タブで、変更する製品を選択します。

Note

クイック起動エクスペリエンスを設定するには、製品が、限定的な可視性または公開された可視性である必要があります。

2. 商品詳細ページで、[フルフィルメントオプション] タブを選択します。
3. [クイック起動] で、[有効化と設定] ボタンを選択します。
4. [アカウントログイン情報] で、購入者がログインしたりアカウントを作成したりできるサイトの URL を入力します。この URL から、購入者エクスペリエンスの新しいタブが開きます。その後、購入者はログインまたはアカウントを作成し、AWS Marketplace に戻ってテンプレートを起動します。

5. [AWS CloudFormation テンプレート] で、[AWS CloudFormation テンプレートを追加] ボタンを選択し、次の情報を入力します。
 1. タイトル — CloudFormation デプロイの名前を指定します。
 2. [説明] - テンプレートの説明を入力します。
 3. [スタック名] -スタックの名前を指定します。この名前は、 の購入者のスタック名です CloudFormation。
 4. CloudFormation テンプレート URL – テンプレートの Amazon Simple Storage Service (Amazon S3) URL を指定します。はこのテンプレート AWS を確認し、最終的なテンプレート URL AWS を提供します。

 Note

顧客の起動プロセスを簡略化するために、設定プロセスに関連するテンプレートの数を最小限に抑えることをお勧めします。理想的には、製品の使用に必要なリソースをデプロイするテンプレートを 1 つ用意するとよいでしょう。CloudFormation テンプレートに関する質問については、AWS Marketplace 事業開発パートナーまたは [AWS Marketplace Seller Operations](#) チームにお問い合わせください。

5. 必要な IAM アクセス許可 – CloudFormation テンプレートのデプロイに必要なアクセス許可を指定します。購入者との間で、[AWS Secrets Manager](#) にシークレットとして保存されているデプロイパラメータを共有する場合は、ポリシーに以下のアクションを含める必要があります。
 - `secretsManager:ListSecrets`
 - `secretsManager:DescribeSecret`
 - `secretsManager:ReplicateSecretToRegions`
 - `secretsManager:GetSecretValue`

 Note

製品に販売者提供の CloudFormation デプロイパラメータ (API キーや[外部 IDs など](#)) が必要な場合は、`PutDeploymentParameter` オペレーションを使用してパラメータを顧客と共有します。詳細については、「AWS Marketplace デプロイサービス API リファレンス」の「[PutDeploymentParameter](#)」を参照してください。

6. (オプション) [手動設定手順] では、製品を手動で設定する購入者向けの説明を入力します。製品のオンボーディングガイドとドキュメントへのリンクを含めることを検討してください。
7. 起動の詳細には、CloudFormation スタックのデプロイ後に購入者が製品にアクセスする URL を指定します。
8. (オプション) クイック起動の許可リストに登録されたアカウントには、可視性が制限され AWS アカウント たクイック起動エクスペリエンスを表示できる のカンマ区切りリストを指定します。
9. [送信] ボタンを選択します。クイック起動エクスペリエンスは、限定的な可視性になります。つまり、自らのアカウントと、許可されているアカウントにのみが表示されます。限定的な可視性になっている場合は、製品をサブスクライブして [アカウントの設定] ボタンを選択した後に、[設定と起動] ページを使用して設定をテストできます。
10. 準備ができたら、クイック起動エクスペリエンスを AWS Marketplace カタログに公開できます。製品詳細ページの [フルフィルメントオプション] タブにある [クイック起動の可視性を更新] ボタンを使用します。

可視性を公開に変更すると、AWS Marketplace Seller Operations チームが設定を確認し、購入者テストを実施し、エクスペリエンスを公開します。

Note

クイック起動エクスペリエンスを有効にする際にサポートが必要な場合は、[AWS Marketplace 販売者オペレーション](#) チームにお問い合わせください。

国別の可用性を更新する

製品を提供できる国を定義することができます。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. [\[SaaS 製品\]](#) タブから、変更する製品を選択します。
3. [\[変更をリクエスト\]](#) ドロップダウンから [\[パブリックオファーを更新\]](#) を選択し、[\[国別の可用性を更新\]](#) を選択します。
4. 以下のオプションのいずれかを選択します。
 - すべての国 - サポートされているすべての国で利用できます。
 - 除外対象のすべての国 - 選択した国を除くすべてのサポート対象国で利用できます。

- カスタムリスト - 製品が販売されている特定の国のリスト。
5. [送信] を選択して、リクエストを送信します。
 6. [リクエスト] タブで、リクエストステータスが [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] に変わります。

製品の返金ポリシーを更新する

[返金ポリシーを更新] を使用して、製品の返金ポリシーを更新できます。

1. <https://aws.amazon.com/marketplace/management/tour/> AWS Marketplace 管理ポータル で を開き、販売者アカウントにサインインします。
2. [SaaS 製品] ページの [SaaS 製品] タブで、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンリストから [パブリックオファーを更新] を選択し、[返金ポリシーを更新] を選択します。
4. 現在の返金ポリシーの詳細がテキストボックスに表示されます。必要に応じて、これらの詳細を確認して変更します。リクエストを送信すると、現在の返金ポリシーが上書きされます。
5. リクエストをレビューに送信するには、[送信] を選択します。
6. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] または [失敗] に更新されます。

エンドユーザー使用許諾契約書 (EULA) を更新する

製品をサブスクライブする新規ユーザー向けの EULA を更新できます。

1. [AWS Marketplace 管理ポータル](#) を開き、販売者アカウントにサインインします。
2. [SaaS 製品] タブから、変更する製品を選択します。
3. [変更をリクエスト] ドロップダウンから [パブリックオファーを更新] を選択し、[EULA を更新] を選択します。
4. [AWS Marketplace の標準契約 \(SCMP\)](#) を選択するか、カスタム EULA を送信してください。カスタム EULA の場合は、契約の Amazon Simple Storage Service (Amazon S3) URL を指定する必要があります。

Note

Amazon S3 バケットへのパブリックアクセスが許可されている必要があります。

5. [送信] を選択して、リクエストを送信します。
6. [リクエスト] タブで、リクエストステータスが [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] に変わります。

SaaS サブスクリプション製品の統合

製品をと統合 AWS Marketplace することは、の 1 つのステップです [SaaS 製品の作成](#)。Software as a Service (SaaS) サブスクリプション製品をと統合するには AWS Marketplace、コードを記述し、いくつかの顧客シナリオに正常に対応できることを証明する必要があります。以下のセクションでは、これらのシナリオとその対応方法について説明し、統合テストの概要を説明します。

Note

開始する前に、[AWS Marketplace](#) で software-as-a-service (SaaS) 製品に適した料金モデルを選択していることを確認してください AWS Marketplace。詳細については、「[SaaS 製品の計画](#)」を参照してください。

トピック

- [シナリオ: サービスが新規顧客を検証する](#)
- [シナリオ: 使用量を計測する](#)
- [シナリオ: ユーザーサブスクリプションの変更を監視する](#)
- [シナリオ: 顧客のサブスクリプションを確認する](#)
- [SaaS サブスクリプション製品統合のテスト](#)

シナリオ: サービスが新規顧客を検証する

顧客が製品をサブスクライブすると、その顧客は登録 URL (一時的な x-amzn-marketplace-token トークンを含む HTTP POST リクエスト) にリダイレクトされます。このリクエストには、次の方法で応答します。

1. AWS Marketplace Metering Service で [ResolveCustomer](#) API オペレーションを呼び出して、CustomerIdentifier、CustomerAWSAccountId、ProductCode のトークンを交換します。

2. CustomerIdentifier、CustomerAWSAccountID、ProductCode は、今後の呼び出しに備えてシステム内に残しておきます。顧客が有効なサブスクリプションを持っているかどうかと、顧客について必要な情報をすべて保存する必要があります。
3. リクエストへの対応として、ユーザーが初めて使用する時の操作を (サービスに応じて) 表示する必要があります。

シナリオ: 使用量を計測する

顧客がサービスを利用し始めたら、計測レコードを 1 時間ごとに送信する必要があります。計測の詳細については、[使用量の計測](#) を参照してください。

AWS CloudTrail を使用してアクティビティをモニタリングし、請求情報が に送信されていることを確認することをお勧めします AWS。計測レコードを送信する場合は、次の点に注意してください。

- 計測リクエストは、毎時 0 分に重複排除されます。
- 送信したレコードは 1 時間ごとに累積されます。
- 過去 1 時間以内にレコードがない場合でも、ベストプラクティスとして、使用量 0 の計測レコードを 1 時間ごとに送信することを強くお勧めします。

シナリオ: ユーザーサブスクリプションの変更を監視する

Amazon Simple Queue Service (Amazon SQS) キューを設定し、製品の Amazon SNS トピックをサブスクライブします。SNS トピック情報は、製品の作成時に AWS Marketplace Seller Operations チームから受け取った E メールメッセージに含まれていました。詳細については、「[SaaS 製品の作成](#)」を参照してください。SNS トピックをサブスクライブすると、特定の顧客へのアクセス権の付与や取り消しなど、顧客のサブスクリプションの変更に関する通知が届きます。

Note

Amazon SNS トピックの Amazon リソースネーム (ARN) は `arn:aws:sns:us-east-1:<account id>:aws-mp-subscription-notification-<product code>` のようになります。

応答する必要がある通知は以下のとおりです。

- `subscribe-success` - 顧客がサブスクライブされ、顧客 ID に照合して正常に計測できる。

- `unsubscribe-pending` - 顧客がサブスクリプション解除中である。最新の計測レコードはすべて送信する必要があります。
- `unsubscribe-success` - 顧客が購読を解除した。顧客の計測レコードは受け付けられなくなります。顧客リソースを閉鎖する手順に従い、保存ポリシーを遵守してください。
- `subscribe-fail` - 顧客のサブスクリプションが失敗した。顧客 ID に照合して計測したり、顧客に代わってリソースを作成したりしないでください。

シナリオ: 顧客のサブスクリプションを確認する

顧客に代わってリソースを作成する前に、顧客が製品にアクセスできることを確認してください。Amazon SQS 経由で受け取った通知から顧客の最新のステータスを保存し、顧客にアクセス許可があるかどうかを確認できます。

SaaS サブスクリプション製品統合のテスト

SaaS サブスクリプション製品を と統合したら AWS Marketplace、詳細なテストを実施して、統合が成功していることを確認する必要があります。次の手順では、製品統合を確認するステップを示します。

Note

自分のアカウントを使って製品をサブスクライブし、正常に統合できたことを確認します。料金を一時的に下げて、それらのアカウントで高額な料金を発生させずに購入フローをテストすることができます。価格を一時的に引き下げたり、追加のテストアカウントに製品へのアクセスを許可したりする方法については、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。

製品が発売された後も、サービスは新規顧客向けに引き続きこれらのシナリオに対応する必要があります。

1. 許可したアカウントを使用して、製品をサブスクライブしてカスタマーエクスペリエンスをテストします。
2. 許可したアカウントでサブスクライブした後、アカウントが登録 URL にリダイレクトされることと、そのリダイレクトが一時トークンを含む POST リクエストであることを確認します。将来の呼び出しに備えて、アプリケーションに顧客 ID が保持されていることを確認します。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。

- 上記の手順でテストアカウントを検証したら、そのアカウントをアプリケーションにオンボードします。例えば、テスト用の顧客でフォームに入力して、新しいユーザーを作成することができます。または、SaaS アプリケーションにアクセスするための次の手順を別途用意します。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。
- テスト顧客がオンボーディングされたら、 の BatchMeterUsage API オペレーションを使用して、課金 AWS 目的で計測レコードを に送信するリクエストを行います AWS Marketplace Metering Service。これは [シナリオ: 使用量を計測する](#) をテストします。
- サブスクリプションの変更をテストします。考えられるシナリオには、サブスクリプションの解除、サブスクリプションの成功、サブスクリプションの失敗などがあります。これは [シナリオ: ユーザーサブスクリプションの変更を監視する](#) をテストします。
- サブスクリプションの成功を確認します。テストアカウントでサブスクリプションの成功のメッセージの Amazon SNS 通知を受け取ると、計測を開始できます。Amazon SNS 通知を受け取る AWS Marketplace Metering Service 前に に送信されたレコードは計測されません。Amazon SNS これは [シナリオ: 顧客のサブスクリプションを確認する](#) をテストします。

Note

請求に関する問題を防ぐため、プログラムでこの通知を待ってから顧客のためのリソースを起動することを強くお勧めします。

- すべての統合要件を満たし、ソリューションをテストしたら、AWS Marketplace Seller Operations チームに通知します。BatchMeterUsage API オペレーションで計測レコードが正常に送信されたことを確認して、ソリューションに関する一連の最終テストを実行します。

統合とテストが完了したら、最終レビューを実行し、製品をパブリックに一覧表示できます AWS Marketplace。詳細については、「[SaaS 製品の作成](#)」を参照してください。

SaaS 契約製品を統合する

製品を と統合 AWS Marketplace することは、 の 1 つのステップです [SaaS 製品の作成](#)。Software as a Service (SaaS) 契約製品を と統合するには AWS Marketplace、コードを記述し、いくつかの顧客シナリオに正常に対応できることを証明する必要があります。以下のセクションでは、これらのシナリオとその対応方法について説明し、統合テストの概要を説明します。

Note

開始する前に、AWS Marketplaceで対象の Software as a Service (SaaS) 製品に適した料金モデルを選択していることを確認してください。詳細については、「[SaaS 製品の計画](#)」を参照してください。

トピック

- [シナリオ: サービスが新規顧客を検証する](#)
- [シナリオ: サービスが顧客のリクエストを処理する](#)
- [シナリオ: ユーザーサブスクリプションの変更を監視する](#)
- [SaaS 契約製品統合のテスト](#)

シナリオ: サービスが新規顧客を検証する

顧客が製品をサブスクライブすると、登録 URL にリダイレクトされます。これは一時的な x-amzn-marketplace-token トークンを含む HTTP POST リクエストです。このリクエストには、次の方法で応答します。

1. AWS Marketplace Metering Service で [ResolveCustomer](#) API オペレーションを呼び出して、CustomerIdentifier、CustomerAWSAccountId、ProductCode のトークンを交換します。
2. AWS Marketplace Entitlement Service で [GetEntitlements](#) API オペレーションを呼び出して、顧客がアクセスできるサブスクリプションと数量 (該当する場合) を確認します。
3. CustomerIdentifier、CustomerAWSAccountId、ProductCode は、今後の呼び出しに備えてシステム内に残しておきます。顧客が有効なサブスクリプションを持っているかどうかと、顧客について必要な情報をすべて保存しておきます。
4. リクエストへの対応として、ユーザーが初めて使用するときの操作を (サービスに応じて) 表示する必要があります。

シナリオ: サービスが顧客のリクエストを処理する

顧客がサービスにリクエストをしたら、次のシナリオに適切なアクションまたはメッセージで対応する必要があります。

- 顧客のシステムには顧客 ID がありません。つまり、まだサブスクライブしていないということです。サブスクライブ方法をユーザーに伝える必要があります。
- 顧客 ID があり、GetEntitlements API オペレーションによって適切なエンタイトルメントが返されます。このシナリオでは、リクエストに応答する必要があります。
- 顧客 ID はあるが、GetEntitlements API オペレーションから返されるエンタイトルメントがないか、リクエストを満たすのに十分な数量がありません。このシナリオでは、アクセスの処理方法とユーザーエクスペリエンスの管理方法を決定する必要があります。

シナリオ: ユーザーサブスクリプションの変更を監視する

Amazon Simple Queue Service (Amazon SQS) キューを設定し、製品の Amazon SNS トピックをサブスクライブします。SNS トピック情報は、製品を作成したときに AWS Marketplace オペレーションチームから受信した E メールに記載されていたものです。詳細については、「[SaaS 製品の作成](#)」を参照してください。SNS トピックをサブスクライブすると、特定の顧客へのアクセス権の付与や取り消しなど、顧客エンタイトルメントの変更に関する通知が届きます。

Note

SNS トピックの Amazon リソースネーム (ARN) は `arn:aws:sns:us-east-1:<account id>:aws-mp-entitlement-notification-<product code>` のようになります。

返答が必要な通知は、以下のものだけです。

- entitlement-updated - 顧客エンタイトルメントが変更されたため、GetEntitlements API オペレーションを呼び出して新しいステータスを確認する必要があります。カスタマーストアを更新し、該当する場合は (例えば、顧客の契約が失効している場合)、カスタマーリソース閉鎖のプラクティスに従い、リテンションポリシーを遵守します。

Note

詳細については、「[使用権限の確認](#)」を参照してください。

SaaS 契約製品統合のテスト

SaaS 契約製品を と統合したら AWS Marketplace、詳細なテストを実施して、統合が成功していることを確認する必要があります。次の手順では、製品統合を確認するステップを示します。

Note

自分のアカウントを使って製品をサブスクライブし、正常に統合できたことを確認します。料金を一時的に下げて、それらのアカウントで高額な料金を発生させずに購入フローをテストすることができます。価格を一時的に引き下げたり、追加のテストアカウントに製品へのアクセスを許可したりする方法については、[お問い合わせください](#)。
製品が発売された後も、サービスは新規顧客向けに引き続きこれらのシナリオに対応する必要があります。

1. 許可したアカウントを使用して、製品を契約してカスタマーエクスペリエンスをテストします。
2. アカウントで契約した後、アカウントが登録 URL にリダイレクトされることと、そのリダイレクトが一時トークンを含む POST リクエストであることを確認します。アプリケーションが今後の通話に備えて顧客 ID を保持し、顧客が持つエンタイトルメントを正しく処理するようにしてください。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。
3. 上記の手順でテストアカウントを検証したら、そのアカウントをアプリケーションにオンボードします。例えば、テスト用の顧客でフォームに入力して、新しいユーザーを作成することができます。または、SaaS アプリケーションにアクセスするための次の手順を別途用意します。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。
4. GetEntitlements API オペレーションからエンタイトルメントが返されない場合は、オンボーディング中または検証の進行中に、エンタイトルメントのないユーザーのアクセスとエクスペリエンスを管理する方法を決定します。これは [シナリオ: サービスが顧客のリクエストを処理する](#) をテストします。
5. サブスクリプションの変更をテストします。サブスクリプション解除、サブスクリプション成功、サブスクリプション失敗のシナリオをアプリケーションが正しく処理することを確認します。これは [シナリオ: ユーザーサブスクリプションの変更を監視する](#) をテストします。
6. すべての統合要件を満たし、ソリューションをテストしたら、AWS Marketplace 運用チームに通知します。次に、GetEntitlements API オペレーションが正常に呼び出され、新規顧客が十分にオンボーディングされたことを確認して、ソリューションをテストします。

統合とテストが完了したら、最終レビューを実行し、製品をパブリックに一覧表示できます AWS Marketplace。詳細については、「[SaaS 製品の作成](#)」を参照してください。返金リクエストフォームに記入して、テストサブスクリプションをキャンセルすることもできます。サブスクリプションのキャンセルの詳細については、「[the section called “AWS Marketplace 製品の返金プロセス”](#)」を参照してください。

SaaS 契約を pay-as-you-go 製品と統合する

製品をと統合 AWS Marketplace することは、の 1 つのステップです [SaaS 製品の作成](#)。Software as a Service (SaaS) 契約製品をと統合するには AWS Marketplace、コードを記述し、いくつかの顧客シナリオに正常に対応できることを証明する必要があります。以下のセクションでは、これらのシナリオとその対応方法について説明し、統合テストの概要を説明します。

Note

開始する前に、AWS Marketplace で対象の Software as a Service (SaaS) 製品に適した料金モデルを選択していることを確認してください。詳細については、「[SaaS 製品の計画](#)」を参照してください。

トピック

- [シナリオ: サービスが新規顧客を検証する](#)
- [シナリオ: サービスが顧客のリクエストを処理する](#)
- [シナリオ: 使用量を計測する](#)
- [シナリオ: ユーザーエンタイトルメントの変更を監視する](#)
- [SaaS 契約製品統合のテスト](#)

シナリオ: サービスが新規顧客を検証する

顧客が製品をサブスクライブすると、登録 URL にリダイレクトされます。これは一時的な x-amzn-marketplace-token トークンを含む HTTP POST リクエストです。このリクエストには、次の方法で応答します。

1. AWS Marketplace Metering Service で [ResolveCustomer](#) API オペレーションを呼び出して、CustomerIdentifier、CustomerAWSAccountId、ProductCode のトークンを交換します。

2. AWS Marketplace Entitlement Service で [GetEntitlements](#) アクションを呼び出して、顧客がアクセスできるサブスクリプションと数量 (該当する場合) を確認します。
3. CustomerIdentifier、CustomerAWSAccountId、ProductCode は、今後の呼び出しに備えてシステム内に残しておきます。顧客が有効なサブスクリプションを持っているかどうかと、顧客について必要な情報をすべて保存しておきます。
4. リクエストへの対応として、ユーザーが初めて使用するときの操作を (サービスに応じて) 表示する必要があります。

シナリオ: サービスが顧客のリクエストを処理する

顧客がサービスにリクエストをしたら、次のシナリオに適切なアクションまたはメッセージで対応する必要があります。

- 顧客のシステムには顧客 ID がありません。つまり、まだサブスクライブしていないということです。サブスクライブ方法を説明するメッセージを送る必要があります。
- 顧客 ID があり、GetEntitlements API オペレーションによって適切なエンタイトルメントが返されます。このシナリオでは、リクエストに回答する必要があります。
- 顧客 ID はあるが、GetEntitlements API オペレーションから返されるエンタイトルメントがないか、リクエストを満たすのに十分な数量がありません。このシナリオでは、アクセスの処理方法とユーザーエクスペリエンスの管理方法を決定する必要があります。

シナリオ: 使用量を計測する

顧客がサービスを利用し始めたら、計測レコードを 1 時間ごとに送信する必要があります。計測の詳細については、[使用量の計測](#) を参照してください。

AWS CloudTrail を使用してアクティビティをモニタリングし、請求情報が に送信されていることを確認することをお勧めします AWS。計測レコードを送信する場合は、次の点に注意してください。

- 計測リクエストは、毎時 0 分に重複排除されます。
- 送信したレコードは 1 時間ごとに累積されます。
- 過去 1 時間以内にレコードがない場合でも、ベストプラクティスとして、使用量 0 の計測レコードを 1 時間ごとに送信することを強くお勧めします。

シナリオ:ユーザーエンタitlementメントの変更を監視する

Amazon Simple Queue Service (Amazon SQS) キューを設定し、製品の Amazon SNS トピックをサブスクライブします。SNS トピックは 2 つあり、1 つはエンタitlementメントの変更に、もう 1 つはサブスクリプションの変更に使われます。トピック情報は、製品の作成時に AWS Marketplace Seller Operations チームから受け取った E メールメッセージに含まれていました。詳細については、「[SaaS 製品の作成](#)」を参照してください。SNS トピックをサブスクライブすると、特定の顧客へのアクセス権の付与や取り消しなど、顧客サブスクリプションの変更に関する通知が届きます。

Note

サブスクリプション変更の SNS トピックの Amazon リソースネーム (ARN) は、arn:aws:sns:us-east-1:<account id>:aws-mp-subscription-notification-<product code> のようになります。エンタitlementメント変更の SNS トピックの ARN は arn:aws:sns:us-east-1:<account id>:aws-mp-entitlement-notification-<product code> のようになります。

返答する必要がある通知は次のとおりです。

- entitlement-updated (エンタitlementメント SNS トピック内) - 顧客エンタitlementメントが変更されたため、GetEntitlements API オペレーションを呼び出して新しいステータスを確認する必要があります。カスタマーストアを更新し、該当する場合は (例えば、顧客の契約が失効している場合)、カスタマーリソース閉鎖のプラクティスに従い、リテンションポリシーを遵守します。
- subscribe-success (サブスクリプション SNS トピック内) - 顧客はサブスクライブしており、顧客 ID と正常に照合できます。
- unsubscribe-pending (サブスクリプション SNS トピック内) - 顧客はサブスクリプション解除中です。最新の計測レコードはすべて送信する必要があります。
- unsubscribe-success (サブスクリプション SNS トピック内) - 顧客がサブスクリプションを解除しました。顧客の計測レコードは受け付けられなくなります。顧客リソースを閉鎖する手順に従い、保存ポリシーを遵守してください。
- subscribe-fail (サブスクリプション SNS トピック内) - 顧客のサブスクリプションが失敗しました。顧客の顧客 ID と照合して計測したり、顧客に代わってリソースを有効にしたりしないでください。

Note

詳細については、「[使用権限の確認](#)」を参照してください。

SaaS 契約製品統合のテスト

契約を pay-as-you-go 製品と統合したら AWS Marketplace、詳細なテストを実施して、統合が成功していることを確認する必要があります。次の手順では、製品統合を確認するステップを示します。

Note

自分のアカウントを使って製品をサブスクライブし、正常に統合できたことを確認します。料金を一時的に下げて、それらのアカウントで高額な料金を発生させずに購入フローをテストすることができます。価格を一時的に引き下げたり、追加のテストアカウントに製品へのアクセスを許可したりする方法については、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。製品が発売された後も、サービスは新規顧客向けに引き続きこれらのシナリオに対応する必要があります。

1. 許可したアカウントを使用して、製品を契約してカスタマーエクスペリエンスをテストします。
2. アカウントで契約した後、アカウントが登録 URL にリダイレクトされることと、そのリダイレクトが一時トークンを含む POST リクエストであることを確認します。アプリケーションが今後の通話に備えて顧客 ID を保持し、顧客が持つエンタイトルメントを正しく処理するようにしてください。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。
3. 上記の手順でテストアカウントを検証したら、そのアカウントをアプリケーションにオンボードします。例えば、テスト用の顧客でフォームに入力して、新しいユーザーを作成することができます。または、SaaS アプリケーションにアクセスするための次の手順を別途用意します。これは [シナリオ: サービスが新規顧客を検証する](#) の一部をテストします。
4. GetEntitlements API オペレーションからエンタイトルメントが返されない場合は、オンボーディング中または検証の進行中に、エンタイトルメントのないユーザーのアクセスとエクスペリエンスを管理する方法を決定します。これは [シナリオ: サービスが顧客のリクエストを処理する](#) をテストします。
5. テスト顧客がオンボーディングされたら、の BatchMeterUsage API オペレーションを使用して、課金 AWS 目的で計測レコードを に送信するリクエストを行います AWS Marketplace Metering Service。これは [シナリオ: 使用量を計測する](#) をテストします。

6. サブスクリプションの変更をテストします。サブスクリプション解除、サブスクリプション成功、サブスクリプション失敗のシナリオをアプリケーションが正しく処理することを確認します。これは [シナリオ:ユーザーエンタイトルメントの変更を監視する](#) をテストします。
7. すべての統合要件を満たし、ソリューションをテストしたら、AWS Marketplace Seller Operations チームに通知します。次に、GetEntitlements API オペレーションが正常に呼び出され、新規顧客が十分にオンボーディングされたことを確認して、ソリューションをテストします。また、BatchMeterUsage API オペレーションを使用して計測レコードが正常に送信されていることも確認します。

統合とテストが完了したら、最終レビューを実行し、製品をパブリックに一覧表示できます AWS Marketplace。詳細については、「[SaaS 製品の作成](#)」を参照してください。

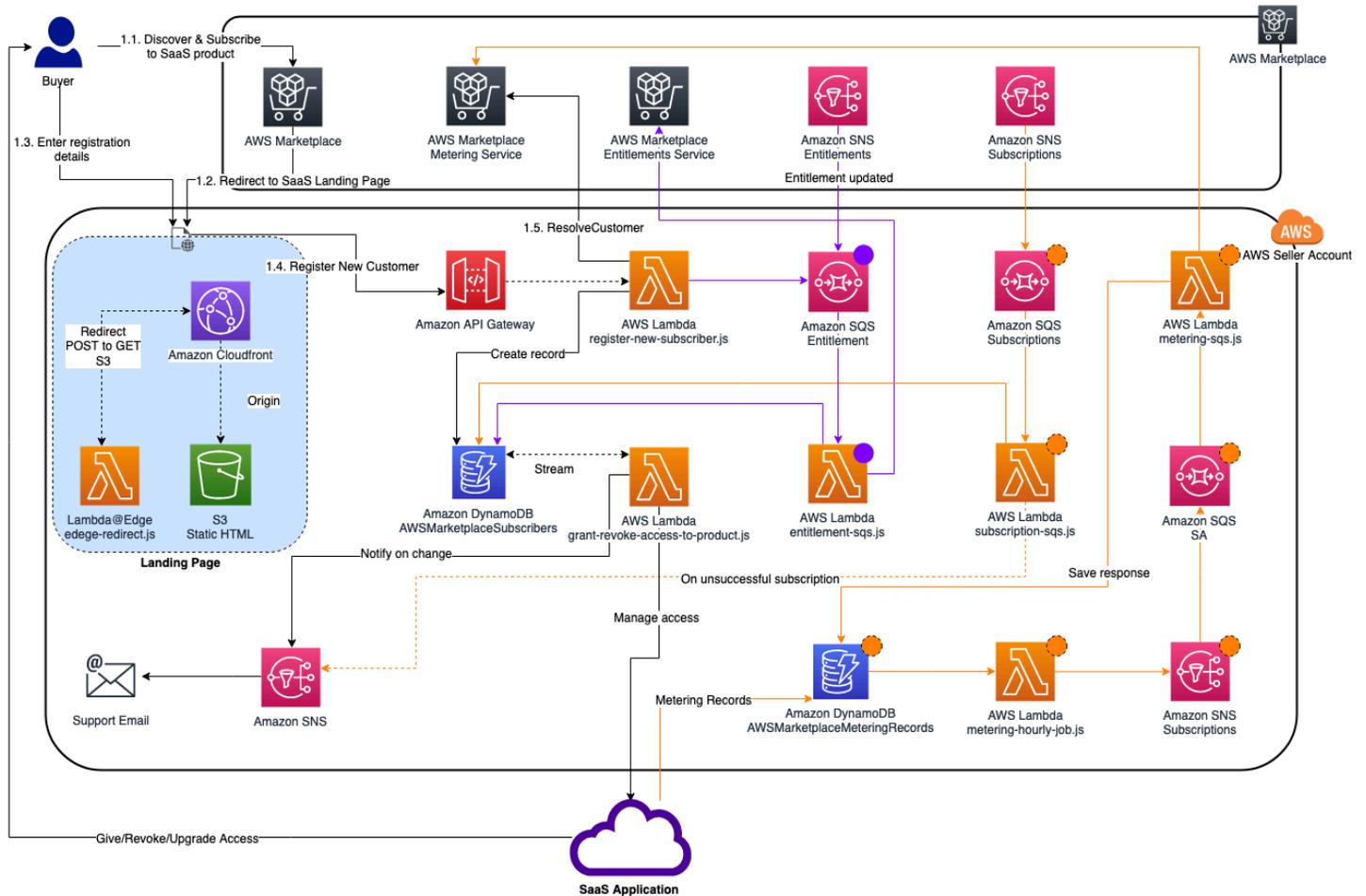
サーバーレス SaaS 統合ソリューションのデプロイ

AWS Marketplace サーバーレス SaaS 統合デプロイは、ベンダーの SaaS ソリューションを の対応するリストに正常に統合するために必要なコア機能を満たします AWS Marketplace。これらの機能には、新規顧客登録の受け入れ、顧客のアクセスの許可と取り消し、顧客の資格の更新、計測された使用量の報告などが含まれます。

ここで紹介する動画では、AWS Marketplace サーバーレス SaaS 統合の AWS クイックスタートをデプロイする方法について説明します。この自動リファレンスデプロイを使用すると、新しい Software as a Service (SaaS) ソリューションを AWS Marketplace 販売者アカウントで簡単に統合し、出品プロセスを高速化し、go-to-market 時間を大幅に短縮できます。

[AWS Marketplace サーバーレス SaaS 統合のための AWS クイックスタート](#)

次の図は、AWS 環境での AWS Marketplace サーバーレス SaaS 統合が、次のイベントワークフローをどのように設定するかを示しています。



に serverless SaaS 統合をデプロイする方法の詳細については AWS クラウド、[AWS Marketplace Serverless SaaS 統合クイックスタートリファレンスデプロイガイド](#)を参照してください。このクイックスタートリファレンスガイドは、新しい SaaS 出品で必要な統合を完了するための軽量のサーバーレスソリューションを求めている、登録済み AWS Marketplace 販売者を対象としています。

SaaS 製品の計画

Software as a Service (SaaS) 製品を に追加する前に AWS Marketplace、まずいくつかの計画を行う必要があります。このステップは、製品が成功を収めるために不可欠です。計画が不十分だと、請求に問題が生じたり、AWS Marketplaceで製品の再作成が必要になったりする場合があります。

⚠ Important

製品の設定のほとんどは、設定後に変更することはできません。で製品を作成した後に変更する必要がある場合は AWS Marketplace、おそらく正しい設定で新しい製品を作成する必要があります。

料金の計画

の SaaS 製品には 3 つの料金モデルがあります AWS Marketplace。製品に適した料金モデルを選択することは、製品を計画するにあたり最も重要な決定です。間違った価格モデルを選択すると、数週間の遅れが生じる可能性があります。料金モデルによって、顧客の支払いオプションと、作成、テスト、デプロイする必要がある請求統合コードが決まります。さまざまなタイプの価格モデルについて詳しくは、「[SaaS 製品の価格設定](#)」を参照してください。

Note

すべての SaaS 価格モデルは無料トライアルをサポートしています。詳細については、「[SaaS 無料トライアル](#)」を参照してください。

請求の統合の計画

SaaS 製品を に配置する利点の 1 つは、請求の統合 AWS Marketplace です。この利点を活用するには、選択した料金モデルに応じて AWS Marketplace Metering Service または AWS Marketplace Entitlement Service と統合する必要があります。これらの 2 つのサービスにより、請求と使用量のレポートが正確に行われるようになります。

統合を計画したら、有効にする前に製品との統合をテストする必要があります。統合とテストの詳細については、「[AWS Marketplace Metering Service API および使用権限管理サービス API へのアクセス](#)」を参照してください。

Amazon SNS インテグレーションを計画する

SaaS 製品についてサブスクライブできる Amazon Simple Notification Service (Amazon SNS) トピックは 2 つあります。詳細については、「[SaaS 通知](#)」を参照してください。これらのメッセージは、お客様が開始 AWS したサブスクリプションや契約の変更をプログラムで処理するのに役立ちます。Amazon SNS 通知は、顧客が製品登録ウェブサイトで新しいアカウントを登録できるようにするプログラムによるトリガーとして使用できます。また、サブスクリプションの有効期限が切れた顧客による製品へのアクセスを拒否することもできます。通知の処理をどのようにプログラムしているかに応じて、顧客が通知を受け取る方法を選択できます。

顧客が製品にアクセスする方法の計画

このセクションでは、購入者が製品にアクセスできるようにする方法について説明します。

SaaS 製品の登録ウェブサイトの計画

SaaS 製品を購入する顧客は、その製品にアクセスする必要があります。顧客が製品にアクセスする方法を計画および実装することが必要です。SaaS 製品は、以下のアクセスオプションをサポートしています。

- クイック起動
- AWS PrivateLink
- 販売者の製品ウェブサイト

登録ウェブサイトを使用して AWS Marketplace の顧客を認証するには、「[SaaS 顧客オンボーディング](#)」を参照してください。

顧客が製品にアクセスするためのクイック起動の使用

クイック起動デプロイオプションを使用すると、購入者が製品の設定、デプロイ、発売に必要な時間とリソースを削減できます。クイック起動を使用すると、購入者がプロセス中にアクセスしなければならないサイトの数が減ります。詳細については、「[クイック起動の設定](#)」を参照してください。

顧客 AWS PrivateLink 向けの を使用した SaaS 製品へのアクセス

[PrivateLink での AWS の使用 AWS Marketplace](#) を使用して Amazon Virtual Private Cloud (Amazon VPC) エンドポイントサービスとして、サービスを設定できます。顧客は、VPC エンドポイントを作成して、AWS クラウド クラウド仮想ネットワークを通じて販売者のソフトウェアにアクセスできます。または、販売者が所有および管理するウェブサイトを通じてソフトウェア製品へのアクセスを提供することもできます。その場合、顧客がインターネット間の接続を作成します。

独自の Web サイトの使用

販売者の SaaS 製品は、販売者の環境でホストされており、販売者が管理および保守を行うウェブサイトなどのパブリックエンドポイントを通じてインターネット経由でアクセスする必要があります。一般的に、販売者には、顧客が製品の登録、製品の使用のためのサインイン、製品のサポートへのアクセスのために使用するウェブサイトがあります。

SaaS 製品ガイドライン

AWS Marketplace は、お客様の安全、安全、信頼できるプラットフォームを推進 AWS Marketplace するために、のすべての SaaS 製品およびサービスについてこれらのガイドラインを維持しています。

すべての製品および関連するメタデータは、送信時にレビューされ、現在の AWS Marketplace ガイドラインを満たしているか超えているかが確認されます。これらのガイドラインは、進化するセキュリティ要件を満たすために、見直され、調整されます。さらに、は製品 AWS Marketplace を継続的に見直して、これらのガイドラインの変更を満たしていることを確認します。製品がコンプライアンスに準拠していない場合、製品の更新が必要となる場合があります。場合によっては、問題が解決するまで新規サブスクライバーが製品を一時的に利用できなくなることがあります。

製品設定ガイドライン

すべての SaaS 製品は、次の製品設定ガイドラインに従う必要があります。

- 少なくとも 1 つの価格ディメンションの料金が \$0.00 を超える必要があります。
- すべての価格ディメンションは、実際のソフトウェアに関連するものでなければならず、ソフトウェアに関係のない他の製品またはサービスを含むことはできません。
- AWS GovCloud (US) リージョンでのみ提供される SaaS 製品には、製品タイトルの GovCloud どこかに を含める必要があります。

顧客情報の要件

すべての SaaS 製品は、次の顧客情報の要件に従う必要があります。

- SaaS 製品は、AWS Marketplace に記載されているディメンションに基づいて完全に課金される必要があります。
- クレジットカードや銀行口座情報など、SaaS 製品に対する顧客支払い情報はいかなる時点でも収集できません。

製品使用ガイドライン

すべての SaaS 製品は、次の製品使用ガイドラインに従う必要があります。

- で製品をサブスクライブした後 AWS Marketplace、お客様は SaaS アプリケーション内にアカウントを作成し、ウェブコンソールにアクセスできるはずですが、顧客がすぐにアプリケーションにアクセスできない場合は、アクセスするタイミングに関する具体的な指示を記載したメッセージを提供する必要があります。アカウントが作成された後に、アカウントが作成されたことを確認する通知を明確な次のステップとともに顧客に送信する必要があります。
- 顧客がすでに SaaS アプリケーションにアカウントを持っている場合、顧客はフルフィルメントランディングページからログインできる必要があります。

- 顧客は、関連する契約またはサブスクリプション使用の情報など、SaaS アプリケーション内でサブスクリプションのステータスを確認できる必要があります。
- 顧客は、アプリケーションの使用、トラブルシューティング、返金のリクエスト (該当する場合) などの問題について、簡単にサポートを受けることができる必要があります。サポートの問い合わせオプションは、フルフィルメントランディングページで指定する必要があります。
- 製品ソフトウェアとメタデータには、AWS Marketplaceでは利用できない他のクラウドプラットフォーム、追加の製品、アップセルサービス、または無料トライアルオファーにユーザーをリダイレクトする言葉を含めてはいけません。

SaaS 製品の無料トライアルの詳細については、「[SaaS 無料トライアル](#)」を参照してください。

- 製品が別の製品または別の ISV 製品のアドオンである場合、製品の説明には、それが他の製品の機能を拡張するものであり、これがないと製品の有用性が非常に限られることを明記する必要があります。例えば、この製品は <製品名> の機能を拡張するものであり、それがなければ、この製品の有用性は非常に限られています。<製品名> は、このリストのすべての機能を利用するには、独自のライセンスが必要な場合がありますのでご注意ください。

アーキテクチャガイドライン

すべての SaaS 製品は、次のアーキテクチャガイドラインに従う必要があります。

- アプリケーションの一部は、所有 AWS アカウント する でホストする必要があります。
- すべてのアプリケーションコンポーネントは、管理するインフラストラクチャでホストされている必要があります。顧客のインフラストラクチャに追加のリソースを必要とするアプリケーションは、次のガイドラインに従う必要があります。
 - (AWS STS) や AWS Security Token Service (AWS Identity and Access Management IAM) など、安全な方法でリソースをプロビジョニングします。
 - プロビジョニングされたすべての AWS のサービスの説明、IAM ポリシーステートメント、IAM ロールまたはユーザーが顧客のアカウントでどのようにデプロイおよび使用されるか含む追加ドキュメントを提供します。
 - 製品の説明に、顧客がトランザクションとは AWS Marketplace 別に追加の AWS インフラストラクチャ料金が発生した場合、追加のインフラストラクチャ料金を支払う責任があることを説明する通知を含めます。
 - 製品によってエージェントがデプロイされる場合は、そのエージェントを AWS アカウントにデプロイする方法についての説明について顧客に提供する必要があります。

- お客様のインフラストラクチャで実行されているリソースを必要とするアプリケーションには AWS Marketplace、による追加のレビューが行われます。これには 2~4 週間かかる場合があります。
- プロバイダーとして登録され、SaaS 公開リクエストを送信 AWS アカウントした から AWS Marketplace APIs を正常に呼び出します。SaaS 価格モデルによって、呼び出される API が決まります。
 - SaaS 契約 – [GetEntitlements](#) の AWS Marketplace Entitlement Service。
 - 消費を伴う SaaS 契約 – [GetEntitlements](#) AWS Marketplace Entitlement Service および AWS Marketplace Metering Service [BatchMeterUsage](#) で。
 - SaaS サブスクリプション – AWS Marketplace Metering Service [BatchMeterUsage](#) 内。
- AWS GovCloud (US) リージョンでのみ提供される SaaS 製品は、他の AWS リージョン と AWS GovCloud (US) リージョン間のアーキテクチャの境界、製品のユースケース、製品に推奨されないワークロードについて概説する必要があります。

SaaS 製品の料金

購入者が Software as a Service (SaaS) 製品を購入すると AWS Marketplace、AWS Marketplace は請求識別子を提供します。販売者はその請求識別子を使用して、AWS Marketplace Entitlement Service および AWS Marketplace Metering Service を呼び出します。その後、お客様は環境内の製品 AWS にアクセスするか、作成した Virtual Private Cloud (VPC) エンドポイント接続を介して製品にアクセスします。

Note

すべての SaaS 価格モデルは無料トライアルをサポートしています。詳細については、「[SaaS 無料トライアル](#)」を参照してください。

SaaS 料金モデル

料金モデル	説明
SaaS サブスクリプション	購入者に SaaS 製品の時間単位の使用料を請求する pay-as-you-go モデル。詳細については、「 SaaS サブスクリプションの料金 」を参照してください。

料金モデル	説明
SaaS 契約	購入者にソフトウェアの使用について前払いで請求するか、販売者が柔軟な支払いスケジュールを提示することができます。顧客は、契約を超えた追加使用量に対して支払うこともできます。詳細については、「 SaaS 契約の料金 」を参照してください。
との SaaS 契約 pay-as-you-go	購入者にソフトウェアの使用について前払いで請求するか、販売者が柔軟な支払いスケジュールを提示することができます。購入者には、契約価格に加えて、従量制の追加料金も請求されます。詳細については、「 SaaS 契約の料金 」を参照してください。

SaaS 製品を で利用できるようにするには AWS Marketplace、[SaaS サブスクリプション料金モデル](#) と [SaaS 契約料金モデル](#) のどちらを提供するかを決定します。

SaaS サブスクリプションの料金

Software as a Service (SaaS) サブスクリプションの場合、 は、お客様が当社に送信する計測レコードに基づいて顧客に AWS Marketplace 請求します。料金は必ず、顧客のアカウントにデプロイされたソフトウェアから 1 時間ごとに計測し、報告する必要があります。その後、すべての使用量は毎月計算され、AMI ベースの AWS Marketplace サービスと同じメカニズムを使用して毎月請求されます。当社は、お客様からの計測レコードの受領に基づき、製品の使用料金を請求することができます。製品の計測レコードの送受信を正常に行う必要があります。

サブスクリプション価格で SaaS 製品を公開するには、まず次の手順を実行します。

1. で新しい SaaS 製品を作成し AWS Marketplace 管理ポータル、新しい SaaS サブスクリプションを選択します。
2. [全般] タブのフィールドに必要な情報を入力します。製品コードを書き留めます。
3. [料金] タブの [料金設定] で、製品の料金を最も正確に表すカテゴリを選択します。料金カテゴリは、AWS Marketplace ウェブサイトの顧客に表示されます。[帯域幅] (GB/s、MB/s)、[データ] (GB、MB、TB)、[ホスト] (時間)、[リクエスト]、[階層] (時間)、[ユーザー] (時間) のいずれかを選

択できます。事前定義カテゴリのどれもニーズに適合しない場合は、汎用的な [単位] カテゴリを選択できます。

次に、料金ディメンションを定義します。各料金ディメンションは、単位あたりの価格を設定できる機能またはサービスを表します。ディメンションの例として、ユーザー数、スキャンされたホスト数、取り込まれたログの GB 数があります。最大で 24 のディメンションを定義できます。定義するディメンションごとに、以下の情報を追加する必要があります。

- ディメンション API 名 - 計測レコードを [AWS Marketplace Metering Service](#) に送信する際に使用する API 名。この名前は顧客がどのディメンションを使用したかを表します。この名前は請求レポートに記載されます。そのレポートにアクセスできるのは販売者だけであるため、わかりやすい名前にする必要はありません。この名前を設定した後に変更することはできません。
- ディメンションの説明 - 製品のディメンションを記述した顧客向けステートメント。説明は 70 文字以内でなければならず、ユーザーフレンドリである必要があります。説明の例としては、「1 時間あたりの管理者数」や「プロビジョニングされた Mbps あたりの帯域幅」などがあります。製品が発行された後に、この説明を変更することはできません。
- ディメンションレート - この製品の FCP 単位あたりのソフトウェア料金 (USD)。このフィールドでは小数点以下 3 桁がサポートされています。

SaaS サブスクリプションの終了時

顧客は、AWS Management Consoleを通じて SaaS サブスクリプション製品のサブスクリプションを解除できます。SaaS サブスクリプション終了プロセスの主なポイントは次のとおりです。

1. その顧客の Amazon SNS トピックを通じて、SaaS 製品に unsubscribe-pending 通知が送信されます。
2. その顧客の残り使用量を計測するまでに 1 時間の猶予があります。
3. その時間が経過すると、販売者は unsubscribe-success 通知を受信します。この時点で、この顧客の計測レコードを送信することはできません。

サブスクリプションを解除した顧客の SaaS アプリケーションで機能をどのように無効化するかは、販売者に任されています。例えば、製品でその顧客の既存の仕事完了するが、その顧客が新しい仕事を作成できないようにすることが考えられます。使用が無効になったことを示すメッセージを顧客に表示するとよいでしょう。お客様は を通じて製品に再サブスクライブできます AWS Marketplace。

SaaS サブスクリプションがキャンセルされたとき

SaaS サブスクリプションキャンセルプロセスの主なポイントは次のとおりです。

1. お客様は、AWS Marketplace ウェブサイトの Your Marketplace ソフトウェアページから SaaS サブスクリプション製品のサブスクリプションをキャンセルできます。

その顧客の Amazon SNS トピックを通じて、SaaS 製品に通知が送信されます。
2. その顧客の残り使用量を計測するまでに 1 時間の猶予があります。
3. 販売者は、キャンセルが進行中の製品から顧客に通知します。顧客が製品を介してキャンセルを指示している場合は、顧客を AWS Marketplace にリダイレクトします。今後料金が発生しないようにするには、でキャンセルを確認する必要があります AWS Marketplace。

SaaS 契約の料金

Software as a Service (SaaS) 契約の場合、お客様は、対象ソフトウェアの購入をもって、契約が成立したものとみなします。契約に基づき、お客様は SaaS 製品の指定された使用量の資格があります。はこれらの資格を SaaS アプリケーションに AWS Marketplace 伝達します。これは、AWS Marketplace Entitlement Service を介して行います。SaaS 契約の料金モデルを使用するとき、アプリケーションは計測レコードを送信しません。代わりに、を呼び出して使用権限を検証します AWS Marketplace Entitlement Service。使用カテゴリ、ディメンション、契約期間を定義します。

AWS Marketplace は、お客様とお客様の間の契約に基づいて、前払いで、またはお客様が定義した支払いスケジュールでお客様に請求します。その時点で、それらのリソースの使用権限が付与されます。契約を超える追加の使用については、ソフトウェアが使用状況を報告し、を通じて当社が受け取った計測レコードに基づいて顧客に AWS Marketplace 請求する必要があります AWS Marketplace Metering Service。

契約価格で SaaS 製品を公開するには、まず次の手順を実行します。

1. で新しい SaaS 製品を作成し AWS Marketplace 管理ポータル、新しい SaaS 契約 を選択します。
2. [全般] タブのフィールドに必要な情報を入力します。製品コードを書き留めます。
3. [価格設定] タブ:
 - a. [価格設定] では、顧客に提供する [契約期間] を選択します。契約期間ごとに異なる価格を入力できます。毎月、1 年、2 年、3 年のオプションから 1 つ以上を選択できます。プライベート オファーを作成する場合は、カスタム期間を月単位 (最大 60 か月) で選択できます。

- b. [提供する契約タイプを選択] では、次のオプションから顧客が製品をどのように購入できるかを選択します。
- 購入者は提供されているオプションを 1 つ以上選択可能 - 顧客は販売者が提供している各価格ディメンションの数量を選択できます。
 - 購入者は提供されている階層を 1 つ選択可能 - 顧客は、機能、サービス、および使用量のさまざまな組み合わせが含まれている選択肢から 1 つの階層を選択します。
- c. 製品の料金を最もよく表す使用単位カテゴリを選択します。料金カテゴリは、AWS Marketplace ウェブサイトの顧客に表示されます。[帯域幅] (GB/s、MB/s)、[データ] (GB、MB、TB)、[ホスト] (時間)、[リクエスト]、[階層] (時間)、[ユーザー] (時間) のいずれかを選択できます。事前定義カテゴリのどれもニーズに適合しない場合は、汎用的な [単位] カテゴリを選択できます。
4. カテゴリを選択したら、料金ディメンションを定義します。各料金ディメンションは、単位あたりの料金を設定できる機能またはサービスを表します。ディメンションの例として、ユーザー数、スキャンされたホスト数、取り込まれたログの GB 数があります。定義したディメンションごとに、名前、説明、価格、および API 名を追加します。名前、価格、および説明は顧客に表示されます。API 名は、AWS Marketplace 次のようにでの追跡とレポートに使用します。
- [AWS Marketplace Entitlement Service](#) を呼び出して、顧客が購入したディメンションを取得する。
 - [AWS Marketplace Metering Service](#) を呼び出して、顧客が使用したディメンションを示す。

契約の料金ディメンションごとに、そのディメンションで契約を超えた追加使用量に応じて顧客が支払うことを選択できます。使用した量に対してのみ顧客が支払う、契約料金なしのディメンションを追加することもできます。

ウィザードを使用して SaaS 製品の契約を作成する場合、料金ディメンションに次のフィールドを定義する必要があります。

- ディメンション API 名 - 使用権限 API を呼び出すときに使用される名前です。この名前は請求レポートに表示され、レポートは外部向けではありません。API 名は最大 15 文字です。この名前を設定した後に変更することはできません。
- ディメンション表示名: - 顧客に表示されるディメンションの名前。この名前は、顧客が製品のディメンションを理解できる名前にします。名前は、最大長が 24 文字で、わかりやすいものである必要があります。この値は変更できます。
- ディメンション説明: - 製品のディメンションに関する追加の情報を提供する、顧客向けのディメンションの説明。説明の最大長は 70 文字です。

- デイメンション - 月別料金 - このデイメンションの 1 か月間オプションの、単位あたりのソフトウェア料金。このフィールドでは小数点以下 3 桁がサポートされています。
- デイメンション - 1 年料金 - このデイメンションの 12 か月間オプションの、単位あたりのソフトウェア料金。このフィールドでは小数点以下 3 桁がサポートされています。月額料金ではありません。この価格は、12 か月間の 1 回限りの料金を表している必要があります。
- デイメンション - 2 年料金 - このデイメンションの 24 か月間オプションの、単位あたりのソフトウェア料金。このフィールドでは小数点以下 3 桁がサポートされています。
- デイメンション - 3 年料金 - このデイメンションの 36 か月間オプションの、単位あたりのソフトウェア料金。このフィールドでは小数点以下 3 桁がサポートされています。

例: データストレージアプリケーション

	月額料金	12 か月料金	24 か月料金	追加使用の Pay-as-you-go 料金
暗号化されていないデータ (GB)	1.50 USD/GB	16.00 USD/GB	30.00 USD/GB	0.1 USD/GB
暗号化されているデータ (GB)	1.55 USD/GB	16.60 USD/GB	31.20 USD/GB	0.11 USD/GB

例: ログモニタリング製品

	月額料金	12 か月料金	追加使用の Pay-as-you-go 料金
Basic (10 ホストのモニタリング、5 コンテナのモニタリング)	100 USD	1,000 USD	
Standard (20 ホストのモニタリング、10 コンテナのモニタリング)	200 USD	2,000 USD	

	月額料金	12 か月料金	追加使用の Pay-as-you-go 料金
Pro (40 ホストのモニタリング、20 コンテナのモニタリング)	400 USD	4000 USD	
1 時間あたりのモニタリング対象追加ホスト			0.1 USD
1 時間あたりのモニタリング対象追加コンテナ			0.2 USD

Note

価格は、1 か月、12 か月、24 か月、36 か月の期間に対して定義できます。製品にこれらのオプションを 1 つ以上提供することができます。期間は各ディメンション間で同じである必要があります。例えば、ReadOnlyUsers と AdminUsers のディメンションがあると仮定します。ReadOnlyUsers の年間料金を提示する場合、AdminUsers の年間料金も提示する必要があります。

SaaS 契約アップグレード

顧客は、期間が長いこと以外でより高い価値の契約にアップグレードできます。たとえば、数量が多い契約や上位の使用権限にアップグレードできます。顧客には既存の契約に案分計算されたクレジットが付与されます。顧客は既存の契約のサイズを縮小することはできません。サイズを縮小できるのは更新時のみであり、そうしない場合は更新をキャンセルします。

使用権限は、SaaS 製品で AWS Marketplace Entitlement Service を呼び出すことによって確認されます。

自動更新

お客様が SaaS 契約 AWS Marketplace を使用して製品を購入すると、契約条件の自動更新に同意できます。顧客は 1 年、2 年、または 3 年の期間中に、使用権限に対して毎月支払い続けます。顧客

は常に更新設定を変更することができます。顧客は更新をキャンセルすることも、異なる数量や期間で契約を更新することもできます。

SaaS 契約が終了したとき

SaaS 契約製品には、契約の有効期限があります。契約が終了すると、次のイベントが発生します。

1. SaaS 製品は、購入者の資格が変更されたことを示す entitlement-updated 通知を受け取ります。は空のレスポンス AWS Marketplace Entitlement Service を返します。
2. その顧客の残り使用量を計測するまでに 1 時間の猶予があります。この時間が経過した後は、この顧客の計測レコードを送信することはできません。

SaaS 契約がキャンセルされたとき

SaaS 契約のキャンセルプロセスの主なポイントは次のとおりです。

1. 顧客は、AWS Support で SaaS 契約製品のキャンセルと返金をリクエストできます。

お客様は、を通じて 48 時間以内に返金をリクエストする必要があります AWS Support。

通常、全額または日割り計算による払い戻しは、3~5 営業日以内に行われます。

2. その顧客の Amazon SNS トピックを通じて、SaaS 製品に通知が送信されます。
3. 追加料金が発生した場合は、1 時間以内に最終的な計測レコードを顧客に送信してください。
4. 販売者は、キャンセルが進行中の製品から顧客に通知します。顧客が製品を介してキャンセルを指示している場合は、顧客を AWS Marketplace にリダイレクトします。今後料金が発生しないようにするには、でキャンセルを確認する必要があります AWS Marketplace。

SaaS 無料トライアル

販売者は、(AMMP) で Software as a Service (SaaS) の無料トライアルオファーを作成できます。AWS Marketplace 管理ポータル 顧客は、SaaS 無料トライアルオプションを使用して、大規模な購入決定を下す前にソフトウェア製品を評価できます。顧客が製品をサブスクライブすると、その製品は有料ユーザーに行うのと同じ方法でエンタイトルメントチェックを行います。

各 AWS アカун トは、SaaS 製品の無料トライアルを 1 回だけ使用できます。無料トライアル中に付与された無料利用額は、AWS 組織内のリンクされたアカウント間で共有されません。1 つのメイン支払いアカウント内の複数の連結アカウントが、それぞれ個別の無料トライアルを作成できます。

Note

Seller Data Delivery Service (SDDS) を使用している場合、Amazon Simple Storage Service (Amazon S3) バケットに[契約詳細トライアルレポート](#)が届きます。レポートには、サブスクライバーの名前と ID、オファー ID、契約の開始日と終了日などの契約の詳細が含まれます。販売者は、新しいサブスクリプションが作成されると、[Amazon Simple Notification Service \(Amazon SNS\)](#) 通知も受け取ります。Amazon SNS 通知には、isFreeTrialTermPresent 無料トライアル契約を識別するフラグが含まれています。

SaaS 無料トライアルオファーの作成

販売者は、AWS Marketplace 管理ポータル (AMMP) で SaaS 無料トライアルオファーを作成できます。

SaaS 無料トライアルオファーを作成するには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. で AWS Marketplace 管理ポータル、次のいずれかを選択します。
 - オファーを作成または管理
 - [オファー] タブ
3. [オファー] ページで [公開無料トライアル] タブを選択し、すべての SaaS 無料トライアルを確認します。
4. [無料トライアルオファーを作成] を選択します。販売者は、公開 SaaS 製品ごとに 1 つ、SaaS 無料トライアルオファーを作成できます。
5. [オファーの基本] で、[製品] を選択し、[次へ] を選択します。
6. [無料トライアル設定] で以下を行います。
 - a. [無料トライアルの期間 (日数)] に日数を入力します。
無料トライアル期間の範囲は 7~90 日間です。
 - b. 既存の公開オファーの [製品ディメンション] を表示します。

SaaS サブスクリプションの無料トライアルの製品ディメンションを変更することはできません。

SaaS 契約の無料トライアルでは、ディメンションごとに数量制限を設定したり、ディメンションを削除または追加したりできます。

7. サービス契約をご覧ください。

EULA バージョンで、[AWS Marketplaceの標準契約] または [カスタム EULA] のいずれかを選択し、[オファーを確認] を選択します。

8. オファーに関するすべての情報を検証して確認し、[オファーを作成] を選択します。

SaaS 無料トライアルオファーのキャンセル

販売者は、AWS Marketplace 管理ポータルからいつでも無料トライアルのオファーをキャンセルできます。

SaaS 無料トライアルオファーをキャンセルするには

1. [AWS Marketplace 管理ポータル](#)にサインインします。
2. で AWS Marketplace 管理ポータル、次のいずれかを選択します。
 - オファーを作成または管理
 - [オファー] タブ
3. [オファー] ページで、オファーを選択します。
4. [オファーを表示] を選択します。
5. [オファーをキャンセル] を選択します。

オファーがキャンセルされると、そのオファーの有効な契約は有効期限が切れるまで有効です。キャンセルされたオファーの新しい契約は作成できません。

SaaS の顧客のオンボーディング

Software as a Service (SaaS) サブスクリプションと SaaS 契約では、顧客は を通じて製品をサブスクライブします AWS Marketplace が、AWS 環境内の製品にアクセスします。製品をサブスクライブした後、顧客は販売者が SaaS 製品の一部として作成および管理するウェブサイトへリダイレクトされ、自分のアカウントを登録して製品を設定します。

SaaS 製品のリストを作成するときに、登録ランディングページの URL を指定します。当社はその URL を使用して、顧客がサブスクライブすると、顧客を販売者の登録ランディングページへリダイ

レクトします。販売者のソフトウェアの登録ランディングページで、販売者は顧客のアカウントを作成するために必要な情報を収集します。使用量の通知について顧客にメールで連絡する予定がある場合は、顧客のメールアドレスを収集することをお勧めします。

登録のランディングページは、 のフォームデータ内の `x-amzn-marketplace-token` トークンを、請求用の顧客の識別子 `AWS Marketplace` とともに識別して受け入れることができる必要があります。次に、そのトークン値を に渡 `AWS Marketplace Metering Service` して、一意の顧客識別子、顧客 `AWS` アカウント ID、および対応する製品コードについて解決する必要があります。コード例については、 [ResolveCustomer コード例](#) を参照してください。

Note

登録トークンは特定のサブスクリプション顧客向けに解決され、生成された各トークンの有効期限は 4 時間です。呼び出し元が同じトークンを使用して API を呼び出している限り、トークンの有効期限が切れるまで同じ応答値を返し続けます。

新しい購入者を受け入れるための SaaS 製品の設定

新しい顧客を受け入れ、適切に計測するために SaaS ソフトウェアを正しく設定する責任を負います。次のプロセスでは、新規顧客のソフトウェアへのアクセスを特定、実装、計測するための推奨方法の 1 つについて説明します。

1. 顧客が `AWS Marketplace` ウェブサイトの製品ページにアクセスすると、製品をサブスクライブすることを選択します。
2. 顧客の `AWS` アカウント が製品のサブスクライブされています。つまり、製品から送信されたサブスクリプションレコードと計測レコードは、顧客の `AWS` 請求書の一部になります。
3. その顧客に対して、その顧客の顧客 ID と販売者の製品コードが含まれている登録トークンが生成されます。
4. 顧客は、ソフトウェアの登録ランディングページにリダイレクトされます。そのページでは、顧客 ID が含まれている登録トークンを受け入れる必要があります。
5. 顧客のブラウザは、ソフトウェアの登録ランディングページ URL に `POST` リクエストを送信します。そのリクエストには、顧客の登録トークンが入っている `POST` パラメータ、`x-amzn-marketplace-token` が 1 つ含まれています。登録ウェブサイトの観点からは、顧客がこのパラメータを持つフォームを送信しました。登録トークンは不透明な文字列です。オフタイプが無料トライアルの場合は、2 つ目のパラメータ、`free-trial` 値を含む `x-amzn-marketplace-offer-type` がリクエストに追加されます。

- この登録トークンを顧客識別子、顧客 AWS アカウント ID、製品コードと引き換えるには、ウェブサイトが AWS Marketplace Metering Service [ResolveCustomer](#) を呼び出す必要があります。顧客識別子は顧客の AWS アカウント ID ではありませんが、製品間で共通なので、顧客レコードの一部として内部ソースに保存する必要があります。製品コードは、AWS が提供する SaaS 製品の一意的文字列です。各 AWS 製品には一意の製品コードが 1 つあり、登録時に割り当てられます。

 Note

ResolveCustomer 呼び出しの例については、「[ResolveCustomer コード例](#)」を参照してください。

- 顧客は、製品でアカウントを作成するか、または既存のアカウントにサインインするように指示されます。

 Note

製品内の既存の顧客アカウントを設定またはリンクするためにチームが手動で行う必要がある場合は、お問い合わせフォームを使用して顧客の連絡先情報を収集できます。連絡先情報を収集し、AWS Marketplace 固有の顧客 ID (ステップ 6 で取得) を解決したら、顧客に通知メッセージを表示します。通知には、アカウントの設定中であることを記載し、販売者からの連絡を待つように伝えます。予想される時期と連絡先情報を顧客に伝えます。また、同じ詳細を記載した E メールメッセージを顧客に送信します。

- 顧客は、その SaaS 製品に固有の認証情報を使用して、販売者のウェブサイトにサインインしています。アカウントのデータベースでは、顧客ごとのエントリを含むことができます。アカウントデータベースには、ステップ 6 で取得した AWS 顧客識別子を入力する顧客識別子の列が必要です。販売者のシステム内の他のアカウントでこの顧客 ID が共有されていないことを確認します。を通じて複数の製品をサブスクライブするお客様の場合 AWS Marketplace、お客様識別子は変わりません。各サブスクリプションには一意の製品コードがあります。
- 販売者登録プロセス中に、顧客が製品をサブスクライブした、またはサブスクリプションを解除したときに通知する、Amazon SNS トピックをサブスクライブします。これらは、顧客のアクションを通知する JSON 形式の Amazon SNS 通知です。
 - 資格通知 - 契約を含む価格モデルの製品では、購入者が新しい契約を作成、アップグレード、更新したとき、または契約が期限切れになったときに通知されます。販売者のアカウントデータベースには、サブスクリプション状態用に列が追加されている必要があります。詳細につい

ては、「[Amazon SNS トピック: aws-mp-entitlement-notification](#)」を参照してください。

- サブスクリプション通知 - 契約やサブスクリプションなど、あらゆる価格モデルの製品について、購入者が製品をサブスクライブしたときまたはサブスクリプションを解除したときに通知が届きます。詳細については、「[Amazon SNS トピック: aws-mp-subscription-notification](#)」を参照してください。

Amazon Simple Queue Service (Amazon SQS) を使用してこれらのメッセージをキャプチャすることを推奨します。販売者が `subscribe-success` でサブスクリプション通知を受信すると、その顧客アカウントは計測する準備ができています。この通知を受信する前に販売者が送信したレコードは計測されません。その方法については、「Amazon Simple Queue Service デベロッパーガイド」の「[ステップ 2: Amazon SQS キューにメッセージを送信する許可を Amazon SNS トピックに付与する](#)」を参照してください。

Note

`subscribe-success` 通知を受信しない限り、製品サブスクリプションをアクティブ化しないでください。

10. 販売者は、販売者のデータベースに保存されている顧客 ID を使用して、AWS Marketplace Metering Service を通じて使用量を計測し、AWS Marketplace Entitlement Service を通じて使用権限をチェックします。

セキュリティと注文

販売者は、からすぐに返された顧客識別子、AWS またはシステムが署名した顧客識別子のみを信頼する責任があります。登録トークンは約 1 時間後に期限切れになる可能性があるため、すぐに解決することをお勧めします。登録トークンを解決したら、登録が完了するまで、顧客のブラウザセッションで署名済みの属性として顧客 ID を保存します。

SaaS 製品に関する Amazon SNS 通知

通知を受け取るには、製品の作成時に提供される AWS Marketplace Amazon Simple Notification Service (Amazon SNS) トピックをサブスクライブします。このトピックでは、製品の顧客のサブスクリプションおよび契約の使用権限の変更に関する通知が提供されます。これにより、特定の顧客に対するアクセス許可の付与および取り消しをいつ行うかを把握できます。

Note

製品の作成プロセス中に SNS トピックの実際の Amazon リソースネーム (ARN) に発行されます。例: `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE`

Software as a Service (SaaS) 製品では、次の Amazon SNS トピックを使用できます。

- [Amazon SNS トピック: `aws-mp-entitlement-notification`](#) - このトピックでは、購入者が新しい契約を作成したり、アップグレードしたり、更新したり、有効期限が切れたりしたときに通知されます。これは、契約 (SaaS 契約および SaaS 契約 (追加使用量あり) (超過) とも呼ばれる) を含む価格モデルの製品でのみ利用できます。
- [Amazon SNS トピック: `aws-mp-subscription-notification`](#) - このトピックでは、購入者が製品をサブスクライブまたはサブスクライブ解除したときに通知します。これには、プライベートオファーの場合は `offer-identifier` が、SaaS 無料トライアルの場合は無料トライアルフラグが含まれます。これは、契約やサブスクリプション (SaaS サブスクリプション、SaaS 契約、SaaS 契約 (追加使用量あり) (超過) とも呼ばれる) を含むすべての価格モデルで利用できます。

これらの通知に応答するシナリオの詳細は、次のトピックを参照してください。

- [SaaS サブスクリプション製品の統合](#)
- [SaaS 契約製品を統合する](#)
- [SaaS 契約を `pay-as-you-go` 製品と統合する](#)

Amazon SNS トピック: `aws-mp-entitlement-notification`

`aws-mp-entitlement-notification` トピック内の各メッセージの形式は次のとおりです。

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLEX",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXXX",
}
```

`<action-name>` は常に `entitlement-updated` になります。

Note

- 使用権限メッセージの場合、アクション (新規、アップグレード、更新、期限切れ) に関係なく、メッセージは同じです。更新の内容を確認するには、以降に `GetEntitlement` を呼び出す必要があります。
- SaaS 契約 (追加使用量あり) (超過) の場合、販売者には [aws-mp-subscription-notification SNS トピック](#) が提供されます。これは、超過料金が追加された場合に出品者が受け取る追加の通知です。販売者が新規顧客を獲得すると、販売者に `entitlement-updated` (何らかのアクションを指す場合があります) が届くだけでなく、新規顧客であることを示すサブスクライブメッセージが届きます。
- 将来の日付の契約 (FDAs の場合、このトピックは契約開始日に開始されます (契約署名日ではありません))。また、契約のキャンセル、交換、更新、失効など、使用権限に後続の変更が発生したときにも開始されます。

契約料金の製品 (との契約を含む pay-as-you-go) は、これらのメッセージに応答する必要があります。応答する方法の詳細については、[シナリオ: ユーザーサブスクリプションの変更を監視する](#) を参照してください。

Amazon SNS トピック: `aws-mp-subscription-notification`

`aws-mp-subscription-notification` トピック内の各メッセージの形式は次のとおりです。

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLEX",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXXX",
  "offer-identifier": "offer-abcexample123",
  "isFreeTrialTermPresent": "true"
}
```

`offer-identifier` は、オファーがプライベートオファーの場合にのみ通知に表示されます。

購入者のサブスクリプションが無料トライアルかどうかは、`isFreeTrialTermPresent` プロパティによって示されます。このプロパティの JSON 値は boolean データ型ではありません。代わりに、値は文字列データ型に変換されます。詳細については、「[SaaS 無料トライアル](#)」を参照してください。

<*action-name*> は、通知によって異なります。指定できるアクションは以下のとおりです。

- `subscribe-success` - この `subscribe-success` メッセージは、出品者が計測レコードの送信を開始できる時期を通知します。
- `subscribe-fail` - メッセージが生成された場合、購入者 `subscribe-fail` が既に から販売者の SaaS ランディングページに移行しているにもかかわらず AWS Marketplace、支払いが失敗した可能性があります。販売者は、商品の使用を許可する前に `subscribe-success` メッセージを待つ必要があります。
- `unsubscribe-pending` - 購入者がサブスクリプションを解除すると、最初に `unsubscribe-pending` メッセージが送信されます。これは、購入者が完全にキャンセルされる前に、販売者が最終的な計測レコードを送信するまでの時間が限られている (約 1 時間) ことを示しています。
- `unsubscribe-success` - `unsubscribe-success` このメッセージはキャンセルの完了を知らせるもので、それ以降の計測レコードは受け付けられません。

Note

- 購入者がサブスクリプションを解除し、最後の `unsubscribe-success` メッセージが送信される前にすぐに再サブスクライブに成功した場合、最後の `unsubscribe-success` メッセージは送信されず、代わりに `subscribe-success` メッセージが送信されます。
- 将来の日付の契約 (FDAs の場合、サブスクライブの成功アクションは契約開始日に開始されます (契約署名日ではありません))。

サブスクリプション料金の製品 (との契約を含む `pay-as-you-go`) は、これらのメッセージに応答する必要があります。応答の管理方法については、次のトピックを参照してください。

- [SaaS サブスクリプション製品の統合](#)
- [SaaS 契約を `pay-as-you-go` 製品と統合する](#)

SNS トピックへの SQS キューのサブスクライブ

Amazon SQS キューを指定された SNS トピックにサブスクライブすることを推奨します。SQS キューを作成し、そのキューをトピックにサブスクライブする方法の詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[Amazon SNS トピックへ Amazon SQS キューをサブスクライブする](#)」を参照してください。

Note

AWS Marketplace SNS トピックをサブスクライブできるのは、製品の販売 AWS アカウントに使用したからのみです。ただし、メッセージを別のアカウントに転送することはできません。詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[別のアカウントの Amazon SQS キューへ Amazon SNS メッセージを送信する](#)」を参照してください。

通知の SQS キューのポーリング

SQS キューを SNS トピックにサブスクライブすると、メッセージは SQS に保存されます。継続的にキューをポーリングし、メッセージを探して必要に応じて処理するサービスを定義する必要があります。

AWS Marketplace Metering Service API および使用権限管理サービス API へのアクセス

このセクションでは、AWS Marketplace Metering Service または AWS Marketplace Entitlement Service との統合プロセスについての概要を示します。これは、Software as a Service (SaaS) 製品の顧客使用状況の請求とレポートが正確であることを確認するために使用されます。限定された状態で公開済みの SaaS サブスクリプション製品または SaaS 契約製品を送信済みであることを前提としています。限定された状態では、テストアカウントを使用して、製品が適切に構成されていて機能していることを確認できますが、製品は一般公開されていません。

Note

SaaS 製品が別の方法で計測を処理する別の AWS マネージドサービスと統合されている場合 (Amazon SageMaker Ground Truth や など AWS WAF)、AWS Marketplace 計測サービスと統合する必要はありません。顧客への二重請求を避けるため、製品の計測は 1 つのシステムでのみ行う必要があります。

トピック

- [使用量の計測](#)
- [使用権限の確認](#)

• [SaaS 製品統合チェックリスト](#)

と認証情報の設定については AWS CLI、「[ユーザーガイド](#)」の「[AWS CLI の設定](#)」の「[AWS Command Line Interface](#)」を参照してください。AWS Python SDK を初めて使用する場合は、Boto 3 [クイックスタート](#) を参照してください。

使用量の計測

Software as a Service (SaaS) サブスクリプションの場合、すべての使用量を計測し、指定した計測レコード AWS に基づいて から顧客に請求されます。SaaS 契約では、販売者は、顧客の契約による使用権限を超えた使用量のみを計測します。アプリケーションが顧客の使用量を計測すると、アプリケーションは蓄積された使用量を AWS に提供します。アプリケーションは、製品を作成するときに定義した価格ディメンション (所定の 1 時間に、転送された GB 数やスキャンされたホスト数など) を計測します。例えば、アプリケーションに送信されたデータ量に基づいて課金する場合、データ量を測定し、対応する計測レコードを 1 時間に 1 回送信できます。は、計測データを使用して、製品の作成時に指定した料金とともに顧客の請求 AWS 額を計算します。

Note

オプションとして、追跡するプロパティ全体で使用量を分割できます。これらのプロパティはタグとして購入者に公開されます。これらのタグにより、購入者はコストをタグ値別の使用量に分けて表示できます。例えば、ユーザーごとに課金し、ユーザーに Department プロパティがある場合、Department のキーを持つタグを使用して使用量の割り当てを作成し、値ごとに 1 つの割り当てを行うことができます。これによってレポートする料金、サイズ、総使用量は変更されませんが、顧客は製品に適したカテゴリ別にコストを確認できます。詳細については、「[ベンダー計測タグ付け \(オプション\)](#)」を参照してください。

時間単位で計測する

すべてのお客様の AWS の使用状況を、一度に最大 25 個のバッチで 1 時間ごとに に報告することをお勧めします。これにより、お客様は使用量とコストをできるだけきめ細かく把握できます。1 時間を超える期間 (1 日など) の使用量を集計する場合は、次の考慮事項に注意してください。

- AWS は、お客様から計測レコードを受け取った場合にのみ、製品の使用に対して顧客に請求できます。製品の計測レコードが正常に送受信されるようにしておくことは販売者の責任です。を使用して AWS CloudTrail、送信するレコードが正しいことを確認できます。その情報を使用する

と、継続的な監査を実行することもできます。詳細については、「[を使用した AWS Marketplace Metering API コールのログ記録 AWS CloudTrail](#)」を参照してください。

- これが料金モデル「サブスクリプション」の SaaS (料金モデル「契約」または「従量制契約」ではない) である場合、購入者はいつでもサブスクリプションを解除できます。購入者がこのサブスクリプション解除アクションを開始すると、販売者はunsubscribe-pending通知を受け取り、最終unsubscribe-success通知の前に、未報告のすべての使用量で 1 時間送信します。2 回目の通知以降のものは受け付けられません。他の 2 つの料金モデルには、サブスクリプション時間に基づいて期間が設定されており、購入者はサブスクリプション中にサブスクリプションを解除することはできません。自動更新をオフにできるのはのみです。自動更新でない場合、その期間の終了時に同じ通知が送信されます。
- 計測レコードを 1 時間ごとに送信せず、アプリケーションまたはネットワークが停止した場合、レコードはさらに遅れます。これにより、サブスクリプションの有効期限が切れた後にアプリケーションまたはネットワークの停止が回復した場合、使用状況がレポートされない可能性があります。
- レポートする使用量がない場合でも、1 時間ごとに計測レコードを送信し続け、その時間にレポートする使用量がない場合は 0 の数量を記録できます。購入者によるディメンションの使用状況を 0 以上報告した後は、レコードを修正できないことに注意してください。したがって、過去 1 時間の使用状況を報告するのがベストプラクティスです。
- 公開中、AWS Marketplace 運用チームは、SaaS アプリケーションが計測レコードを正常に送信したことをテストしてから、製品の公開を許可します。通常、チームは SaaS のモックサインアップを実行し、計測レコードが受信されたことを確認します。

Note

SaaS 製品が別の方法で計測を処理する別の AWS マネージドサービスと統合されている場合 (Amazon SageMaker Ground Truth や など AWS WAF)、AWS Marketplace 計測サービスと統合する必要はありません。顧客への二重請求を避けるため、製品の計測は 1 つのシステムでのみ行う必要があります。現時点では、AWS Marketplace は新しい AWS WAF 製品を公開していないことに注意してください。

使用状況を計測するように製品を設定します。

AWS Marketplace Metering Service の BatchMeterUsage オペレーションを使用して、計測レコードを AWS に配信します。以下に留意してください。

- 当社では、BatchMeterUsage オペレーションによるバッチ処理を使用することを販売者に求めています。
- 当社では、毎時 0 分に計測リクエストを重複排除します。
 - リクエストは、製品/顧客/時間/ディメンションの組み合わせごとに重複排除されます。
 - リクエストはいつでも再試行できますが、元の数量と異なる数量で計測した場合は、元の数量で課金されます。
 - 同じ顧客、ディメンション、時間に対して複数のリクエストを送信する場合、レコードは集計されません。
- 顧客が商品を購入している場合、販売者は最大で過去 6 時間のタイムスタンプ付きの計測レコードを送信できます。顧客が登録を解除した場合、出品者は顧客が登録を解除してから 1 時間以内に計測レコードを送信する必要があります。
- BatchMeterUsage ペイロードは 1 MB を超えてはなりません。ペイロードのサイズを超えないように、BatchMeterUsage リクエストで送信する使用状況レコードの数を選択します。
- AWS Marketplace Metering Service は、「AWS 全般のリファレンス」の[AWS Marketplace 「エンドポイントとクォータ AWS リージョン」](#)に記載されているので利用できます。SaaS 計測製品では、製品のリクエスト時に、デフォルトで米国東部 (バージニア北部) リージョンが有効になっています。他のリージョンを使用する場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。詳細については、[BatchMeter 「の使用」](#)を参照してください。

コード例については、[SaaS 製品統合のコード例](#)を参照してください。

例: ホストスキャン

販売者は、コンピューティングハードウェアで既知のセキュリティ脆弱性を分析する製品を提供しているとします。顧客は、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスのスキャンを手動で開始またはスケジューリングします。製品は、スキャンを実行すると、スキャンした固有のホストの数を 1 時間ごとに集計します。この例の製品では [Hosts (ホスト)] カテゴリを使用しています。スキャンされたホストのタイプに複数のディメンションを宣言できます。たとえば、小型、中型、そして大型のホストに異なる料金を請求できます。

例: ログ分析

顧客の製品、トレンドの報告、および異常によって生成されたログを取り込む SaaS 製品を提供しているとします。顧客がログをその製品にアップロードすると、販売者は受信したデータの量をメガバイト、ギガバイト、またはテラバイト単位で計測します。毎時 10 分ごとの cron ジョブが、直前の 1 時間の顧客ごとの使用量を読み取ります。そのジョブは、バッチレポートを作成

し、BatchMeterUsage オペレーションを使用してそのレポートを AWS に送信します。この例の製品では [Data (データ)] カテゴリを使用しています。この製品は、特定の 1 時間に保存されたログデータの量を計測することもできます。その場合、この製品は、1 時間に受信したデータと 1 時間に保存された総データの 2 つのディメンションで計測できます。顧客がこのデータを削除するか、期限が切れるまで、保存されたデータの計測を続行できます。

ベンダー計測タグ付け (オプション)

ベンダー計測タグ付けは、独立系ソフトウェアベンダー (ISV) がソフトウェアの使用状況についてより詳細な情報を購入者に提供し、コスト配分を行うのに役立ちます。

購入者が使用したソフトウェア使用状況をタグ付けするには、さまざまな方法があります。1 つの方法は、コスト配分に何を期待するかを購入者にたずねます。次に、購入者のアカウントで追跡するプロパティ全体で使用量を分割できます。プロパティの例には、Account ID、Business Unit、Cost Centers、および商品に関連する他のメタデータが含まれます。これらのプロパティはタグとして購入者に公開されます。タグを使用すると、購入者は AWS 請求コンソール (<https://console.aws.amazon.com/billing/>) のタグ値別にコストを使用量に分割して表示できます。ベンダー計測タグ付けでは、報告する価格、ディメンション、合計使用量は変わりません。お客様は商品に適したカテゴリ別にコストを確認できます。

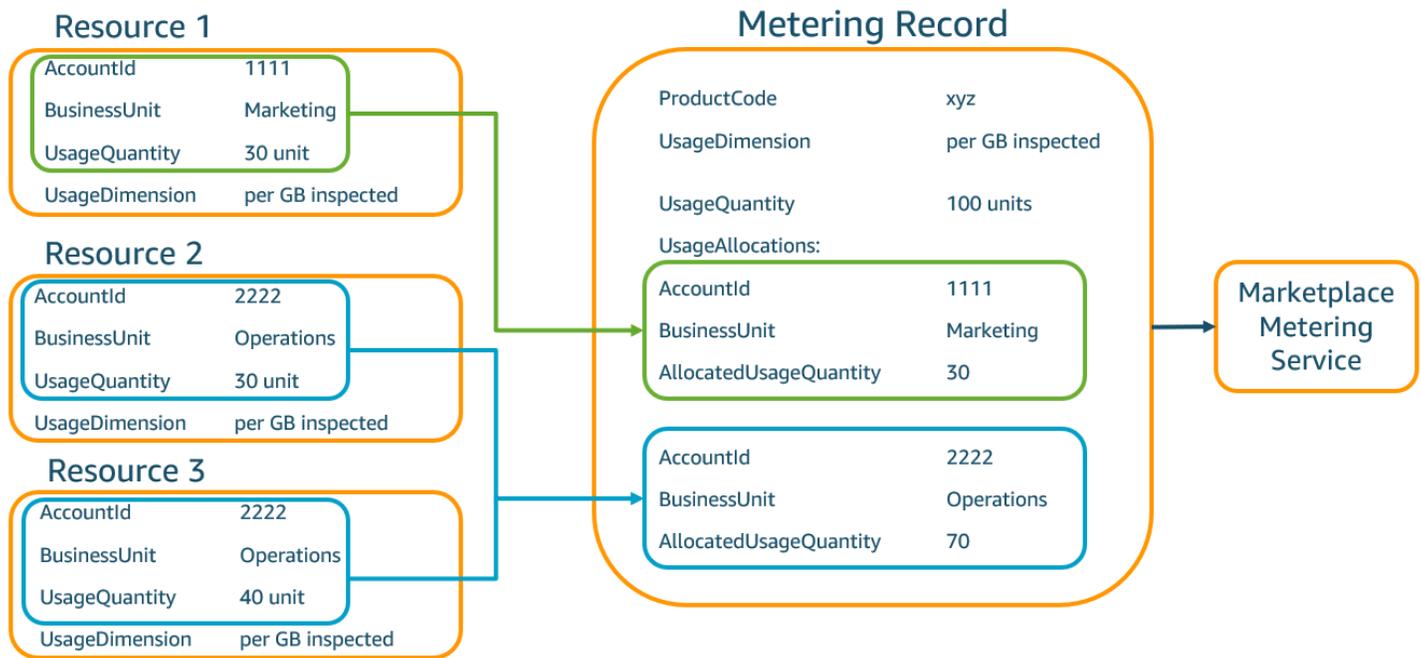
一般的なユースケースでは、購入者が 1 つの AWS アカウントを使って商品を購入します。また、購入者には同じ製品サブスクリプションに関連するユーザーが多数います。Account ID のキーを持つタグを使用して使用量の割り当てを作成し、各ユーザーに使用量を割り当てることができます。この場合、購入者は Billing and Cost Management コンソールで Account ID タグを有効にして、個々のユーザーの使用状況を分析できます。

販売者のエクスペリエンス

販売者は、すべてのリソースの使用量を集計する代わりに、同じタグセットを使用してリソースの計測レコードを集計できます。例えば、販売者は UsageAllocations の異なるバケットを含む計測レコードを作成できます。各バケットは、AccountId や BusinessUnit などの、タグのセットの UsageQuantity を表します。

以下の図では、[リソース 1] には固有の AccountId と BusinessUnit タグセットがあり、[計測レコード] に 1 つのエントリとして表示されます。

[リソース 2] と [リソース 3] にはどちらも同じ AccountId タグ、2222、および同じ BusinessUnit タグ Operations があります。その結果、これらは [計測レコード] の 1 つの UsageAllocations エントリにまとめられます。



また、販売者は、タグのないリソースを単一の UsageAllocation にまとめて、UsageAllocations のエントリの 1 つとして送信することもできます。

制限には以下が含まれます。

- タグの数 - 5
- UsageAllocations のサイズ (カーディナリティ) - 2,500
- リクエストの最大サイズ - 1 MB

検証には次の項目が含まれます。

- タグのキーと値に使用できる文字 - a-zA-Z0-9+ -=._:\@
- UsageAllocation リスト全体の最大タグ数 - 5
- 2 つの UsageAllocations タグに同じタグ (つまり、同じタグキーと値の組み合わせ) を持つことはできません。その場合は、同じ UsageAllocation を使用する必要があります。
- UsageAllocation の AllocatedUsageQuantity の合計は、総使用量である UsageQuantity と等しくなる必要があります。
- ペイロードの最大サイズは 1 MB 以下でなければなりません。これには、入力属性キー (例: UsageRecords、AllocatedUsageQuantity、タグ) が含まれます。

Note

ペイロードの制限を超えないようにするには、ビジネス要件に基づいて最大サイズを設定したサンプルリクエストオブジェクトを作成し、そのオブジェクトを JSON 文字列に変換して、サイズをバイト単位で取得します。1 回の API コールが 1 MB の制限を超えないようにしてください。例えば、1 つの UsageRecord リクエストの最大サイズが 200 KB の場合、リクエストの一部として 5 つを超える UsageRecords を送信しないでください (200 KB * 5 = 1 MB)。

購入者エクスペリエンス

次の表は、購入者が AccountId および BusinessUnit ベンダータグを有効化した後の購入者エクスペリエンスの例を示しています。

この例では、購入者は [コスト使用状況レポート] で割り当てられた使用量を確認できます。ベンダー計測タグにはプレフィックス “aws:marketplace:isv” が使用されます。購入者は、請求情報とコスト管理の、[コスト配分タグ]、[AWS生成コスト配分タグ] でそれらを有効化できます。

[コスト使用状況レポート] の最初と最後の行は、出品者が Metering Service に送信する内容に関係します ([販売者のエクスペリエンス](#) 例を参照)。

コスト使用状況レポート (簡略版)

ProductCode	購入者	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	70	2222	オペレーション
xyz	111122223 333	ネットワーク: 検査対象 (GB) あたり	30	3333	財務

ProductCode	購入者	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	20	4444	IT
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	20	5555	マーケティング
xyz	111122223333	ネットワーク: 検査対象 (GB) あたり	30	1111	マーケティング

コード例については、[使用量割り当てタグ付きの BatchMeterUsage のコード例 \(オプション\)](#) を参照してください。

使用権限の確認

製品が SaaS 契約製品である場合、製品は AWS Marketplace Entitlement Service を呼び出して、を使用して顧客のエンタイトルメントを取得します [GetEntitlements](#)。製品は、そのアカウントで後続の使用を AWS Marketplace エンタイトルメントサービスに対して検証する必要があります。例えば、顧客がアカウントに 10 人のユーザーをプロビジョニングする場合、製品はその容量に対する使用権限について AWS Marketplace エンタイトルメントサービスをチェックする必要があります。

製品に対する顧客のエンタイトルメントを確認するには、エンタイトルメントサービスの AWS Marketplace GetEntitlements オペレーションを使用します。AWS Marketplace Entitlement Service は、米国東部 (バージニア北部) リージョンでのみ利用でき、からアクセスできません `entitlement.marketplace.us-east-1.amazonaws.com`。

GetEntitlements はフィルターとして顧客識別子およびディメンションを受け付けます。ProductCode は必須パラメータです。このオペレーションは、使用権限のページ分割されたリストを返します。結果には、その使用権限が有効である最小期間を示す ExpirationDate フィールドがあります。お客様が自動更新を設定している場合、ExpirationDate フィールドの日付は更新日です。

コードの例については、「[SaaS 製品統合のコード例](#)」を参照してください。

ユーザー操作に関する使用権限の取得

次の例は、ユーザー操作に関する使用権限を取得するプロセスを理解するのに役立ちます。

例: ユーザーベースの製品

所定の顧客に対していくつかのアカウントが存在できる製品を提供しています。その顧客はダッシュボードで新しいユーザーを (たとえば、認証情報を割り当てるために) プロビジョニングできます。顧客が新しいユーザーをプロビジョニングすると、製品は GetEntitlements を呼び出して、そのキャパシティがあることを確認します。そうでない場合は、Metering Service AWS Marketplace を呼び出して、追加のユーザーに請求できます。

例: データストレージ製品

顧客が特定量のデータを暗号化形式または非暗号化形式で保存できる製品を提供しています。顧客は、製品での既存のデータ量および割り当て済みのデータ量が表示されるダッシュボードを表示できます。ダッシュボードでは、GetEntitlements を使用して割り当て量を取得します。

SaaS 製品統合チェックリスト

SaaS 製品が公開される前に、このチェックリストを使用して、必要な設定が完了していることを確認します。

カテゴリ	要件
アクセス	使用する AWS AWS Marketplace アカウントが記載された販売者登録フォームを送信しました。
アクセス	利用規約、銀行口座、および W8/W9 税金フォームを含めて、販売者登録を完了した。
アクセス	登録済みアカウントのクロス AWS Marketplace アカウントロールを設定しました。
製品	AWS Marketplace 管理ポータルでの製品リクエストフォームの記入を完了した。

カテゴリ	要件
製品	AMMP の製品作成ウィザードのメモタブでテスト用の AWS アカウント IDs を提供しました。
製品	[製品] タブで .txt 形式の EULA の URL を指定した。
製品	から製品コードと Amazon SNS トピック情報を受信しました AWS Marketplace。
製品	Amazon SNS トピックを予約購読し、Amazon SNS トピックを予約購読するための Amazon SQS キューを作成した。
請求ソリューション	SaaS サブスクリプション製品の顧客に対する 1 時間ごとの計測レコードを BatchMeterUsage オペレーションに送信できることを検証した。追加使用量に対する計測レコードを SaaS 契約製品の顧客ごとに送信できることを確認した。
請求ソリューション	検証済み SaaS AWS Marketplace 契約製品のエンタイトルメントサービスから顧客エンタイトルメントを検証できます。
請求ソリューション	テストアカウントに対して生成された請求書で、想定どおりに費用が表示されることを検証した。
請求ソリューション	無効な顧客 ID やキャンセル済みサブスクリプションなどの状況をテストした。
製品	製品リクエストを AWS Marketplace に送信して公開しました。
Registration (登録)	HTTP POST リクエストを受け入れることができる HTTPS 登録ページを実装した。

カテゴリ	要件
Registration (登録)	新規の顧客登録を受け入れることができることを検証した。
Registration (登録)	登録トークンを Cookie に保存していないことを検証した。
Registration (登録)	AWS トークン CustomerIdentifier から ProductCode とを取得 ResolveCustomer するためにを使用していることが検証されました。
Registration (登録)	検証済みです。 から受信した登録トークンを遅延 AWS なく解決できます。
Registration (登録)	Gmail などの E メールサービスのアドレスを使用した登録をブロックしていないことをテストした。
Registration (登録)	不完全な登録および複数回の登録試行を受け入れることができることをテストした。
サブスクリプション	unsubscribe-pending メッセージおよび unsubscribe-success メッセージを処理できることをテストした。
サブスクリプション	unsubscribe-pending メッセージを受信してから 1 時間以内に最終的な計測レコードが送信されることを検証した。
セキュリティ	AWS ルートアカウントに API キーがなく、強力なパスワードがあり、ハードウェア多要素認証 (MFA) デバイスに関連付けられていることが検証されました。すべての管理アクセスは、AWS Identity and Access Management (IAM) で作成された ID を通じて行われます。共有アカウントがありません。

カテゴリ	要件
セキュリティ	Amazon Elastic Compute Cloud (Amazon EC2) へのすべてのプログラムによるアクセスに IAM ロールが使用されていることを確認した。スクリプト、ヘッダー、ソースコードに認証情報がハードコードされていないことを確認した。
セキュリティ	包括的なログ記録およびログの統合が維持されていることを検証した。
セキュリティ	アプリケーションサービスと、データベースやファイルシステムへのアクセスを分離する、パブリックとプライベートのサブネットの境界が明確に定義されていることを確認した。機密データを区別し、パブリックデータとプライベートデータを分離する、明確なデータクラスが定義されていることを確認した。
セキュリティ	スケジュールされたキーローテーションを使用した、転送中および休止中のプライベートデータの暗号化が実施されていることを確認した。
セキュリティ	セキュリティインシデントツールが導入および利用されていること、およびタイムリーな調査と復旧に対応するインシデント対応演習が定期的にスケジュールされていることを確認した。
信頼性	システムが、必要に応じてスケールアップ/ダウンして需要の変化に適応できること、および負荷分散を採用して高いパフォーマンスを確保していることを確認した。システムが、必要に応じてエッジベースのキャッシュを提供できることを確認した。

カテゴリ	要件
信頼性	復旧時間と目標復旧時点が規定されていること、および災害復旧が定期的にスケジュールされていることを検証した。コンポーネントの障害が自動トリガーと通知によって自己修復されることを確認した。

レポート作成

AWS Marketplace は、サブスクライバー、財務、使用状況、税金に関するデータを含む SaaS 製品のレポートを生成します。詳細については、「[the section called “販売者レポート”](#)」を参照してください。以下の表に、SaaS 製品の財務状況がどのようにレポートされるかを示します。

レポートを行う	SaaS の内容
日別ビジネスレポート	<p>前払い契約料金は [Fees (料金)] セクションに表示されます。</p> <p>計測使用量料金は [Usage (使用量)] セクションに表示されます。</p>
月別収益レポート	<p>前払い契約料金は [Annual subscriptions (年間サブスクリプション)] セクションに表示されます。</p> <p>計測使用量料金は [Billing and revenue data (請求および収益データ)] セクションに表示されます。</p>
販売補償レポート	前払い契約料金と月別の追加使用量料金は個別の明細項目として表示されます。
顧客サブスクリプションレポート	新しい SaaS 契約は [Annual subscriptions (年間サブスクリプション)] セクションに表示されます。

レポートを行う	SaaS の内容
	新しい SaaS サブスクリプションは [Hourly/monthly subscriptions (時間単位または月単位のサブスクリプション)] セクションに表示されません。

SaaS 製品統合のコード例

次のコード例は、Software as a Service (SaaS) 製品を、製品の公開と保守に必要な AWS Marketplace APIs と統合するのに役立ちます。

トピック

- [ResolveCustomer コード例](#)
- [GetEntitlement コード例](#)
- [BatchMeterUsage コード例](#)
- [使用量割り当てタグ付きの BatchMeterUsage のコード例 \(オプション\)](#)

ResolveCustomer コード例

次のコード例は、すべての料金モデルに関連しています。Python の例によって、x-amzn-marketplace-token トークンが

CustomerIdentifier、ProductCode、CustomerAWSAccountId に交換されま

す。CustomerAWSAccountId はサブスクリプションに関連付けられた AWS アカウント ID です。このコードは、AWS Marketplace 管理ポータルからリダイレクトされたときに、登録ウェブサイト上のアプリケーションで実行されます。リダイレクトは、トークンを含む POST リクエストです。

の詳細については ResolveCustomer、「Metering Service API リファレンス [ResolveCustomer](#)」の「」を参照してください。AWS Marketplace

```
# Import AWS Python SDK and urllib.parse
import boto3
import urllib.parse as urlparse

# Resolving Customer Registration Token
formFields = urlparse.parse_qs(postBody)
regToken = formFields['x-amzn-marketplace-token'][0]
```

```
# If regToken present in POST request, exchange for customerID
if (regToken):
    marketplaceClient = boto3.client('meteringmarketplace')
    customerData = marketplaceClient.resolve_customer(RegistrationToken=regToken)
    productCode = customerData['ProductCode']
    customerID = customerData['CustomerIdentifier']
    customerAWSAccountId = customerData['CustomerAWSAccountId']

    # TODO: Store customer information
    # TODO: Validate no other accounts share the same customerID
```

レスポンスの例

```
{
  'CustomerIdentifier': 'string',
  'CustomerAWSAccountId': 'string',
  'ProductCode': 'string'
}
```

GetEntitlement コード例

次のコード例は、契約を持つ SaaS 製品および消費料金モデルを持つ SaaS 契約に関連しています。Python の例では、顧客がアクティブな使用権限を持っていることを確認します。

の詳細については [GetEntitlement](#)、[「AWS Marketplace Entitlement Service API リファレンス \[GetEntitlement\]\(#\)」](#) の「」を参照してください。

```
# Import AWS Python SDK
import boto3

marketplaceClient = boto3.client('marketplace-entitlement', region_name='us-east-1')

# Filter entitlements for a specific customerID
#
# productCode is supplied after the AWS Marketplace Ops team has published
# the product to limited
#
# customerID is obtained from the ResolveCustomer response
entitlement = marketplaceClient.get_entitlements({
    'ProductCode': 'productCode',
    'Filter' : {
```

```
        'CUSTOMER_IDENTIFIER': [
            'customerID',
        ]
    },
    'NextToken' : 'string',
    'MaxResults': 123
})

# TODO: Verify the dimension a customer is subscribed to and the quantity,
# if applicable
```

レスポンスの例

戻り値は、AWS Marketplace 管理ポータルで製品を作成したときに作成されたディメンションに対応します。

```
{
  "Entitlements": [
    {
      "CustomerIdentifier": "string",
      "Dimension": "string",
      "ExpirationDate": number,
      "ProductCode": "string",
      "Value": {
        "BooleanValue": boolean,
        "DoubleValue": number,
        "IntegerValue": number,
        "StringValue": "string"
      }
    }
  ],
  "NextToken": "string"
}
```

BatchMeterUsage コード例

次のコード例は、SaaS サブスクリプションおよび消費料金モデルを使用した契約に関連しますが、消費のない SaaS 契約製品には該当しません。Python の例では、計測レコードを に送信 AWS Marketplace して、顧客に pay-as-you-go 料金を請求します。

```
# NOTE: Your application will need to aggregate usage for the
```

```
# customer for the hour and set the quantity as seen below.
# AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited
#
# customerID is obtained from the ResolveCustomer response

# Import AWS Python SDK
import boto3

usageRecord = [
    {
        'Timestamp': datetime(2015, 1, 1),
        'CustomerIdentifier': 'customerID',
        'Dimension': 'string',
        'Quantity': 123
    }
]

marketplaceClient = boto3.client('meteringmarketplace')

response = marketplaceClient.batch_meter_usage(usageRecord, productCode)
```

の詳細についてはBatchMeterUsage、「Metering Service API リファレンス[BatchMeterUsage](#)」の「」を参照してください。AWS Marketplace

レスポンスの例

```
{
  'Results': [
    {
      'UsageRecord': {
        'Timestamp': datetime(2015, 1, 1),
        'CustomerIdentifier': 'string',
        'Dimension': 'string',
        'Quantity': 123
      },
      'MeteringRecordId': 'string',
      'Status': 'Success' | 'CustomerNotSubscribed' | 'DuplicateRecord'
    },
  ],
  'UnprocessedRecords': [
```

```
    {
      'Timestamp': datetime(2015, 1, 1),
      'CustomerIdentifier': 'string',
      'Dimension': 'string',
      'Quantity': 123
    }
  ]
}
```

使用量割り当てタグ付きの **BatchMeterUsage** のコード例 (オプション)

次のコード例は、SaaS サブスクリプションおよび消費料金モデルを使用した契約に関連しますが、追加使用量のない SaaS 契約製品には該当しません。Python の例では、適切な使用量割り当てタグを含む計測レコードを AWS Marketplace に送信して、顧客に料金を請求します pay-as-you-go。

```
# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
#       AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited
#
# customerID is obtained from the ResolveCustomer response

# Import AWS Python SDK
import boto3
import time

usageRecords = [
    {
        "Timestamp": int(time.time()),
        "CustomerIdentifier": "customerID",
        "Dimension": "Dimension1",
        "Quantity": 3,
        "UsageAllocations": [
            {
                "AllocatedUsageQuantity": 2,
                "Tags":
                    [
                        { "Key": "BusinessUnit", "Value": "IT" },
                        { "Key": "AccountId", "Value": "123456789" },
                    ]
            }
        ]
    }
]
```

```
    },
    {
        "AllocatedUsageQuantity": 1,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "Finance" },
                { "Key": "AccountId", "Value": "987654321" },
            ]
    },
]
}
]
```

```
marketplaceClient = boto3.client('meteringmarketplace')

response = marketplaceClient.batch_meter_usage(UsageRecords=usageRecords,
        ProductCode="testProduct")
```

の詳細についてはBatchMeterUsage、AWS Marketplace Metering Service 「API リファレンス [BatchMeterUsage](#)」の「」を参照してください。

レスポンスの例

```
{
  "Results": [
    {
      "Timestamp": "1634691015",
      "CustomerIdentifier": "customerID",
      "Dimension": "Dimension1",
      "Quantity": 3,
      "UsageAllocations": [
        {
          "AllocatedUsageQuantity": 2,
          "Tags":
            [
              { "Key": "BusinessUnit", "Value": "IT" },
              { "Key": "AccountId", "Value": "123456789" },
            ]
        },
        {
          "AllocatedUsageQuantity": 1,
```

```
        "Tags":
          [
            { "Key": "BusinessUnit", "Value": "Finance" },
            { "Key": "AccountId", "Value": "987654321" },
          ]
        },
      ],
    },
    "MeteringRecordId": "8fjef98ejf",
    "Status": "Success"
  },
],
"UnprocessedRecords": [
  {
    "Timestamp": "1634691015",
    "CustomerIdentifier": "customerID",
    "Dimension": "Dimension1",
    "Quantity": 3,
    "UsageAllocations": []
  }
]
```

PrivateLink での AWS の使用 AWS Marketplace

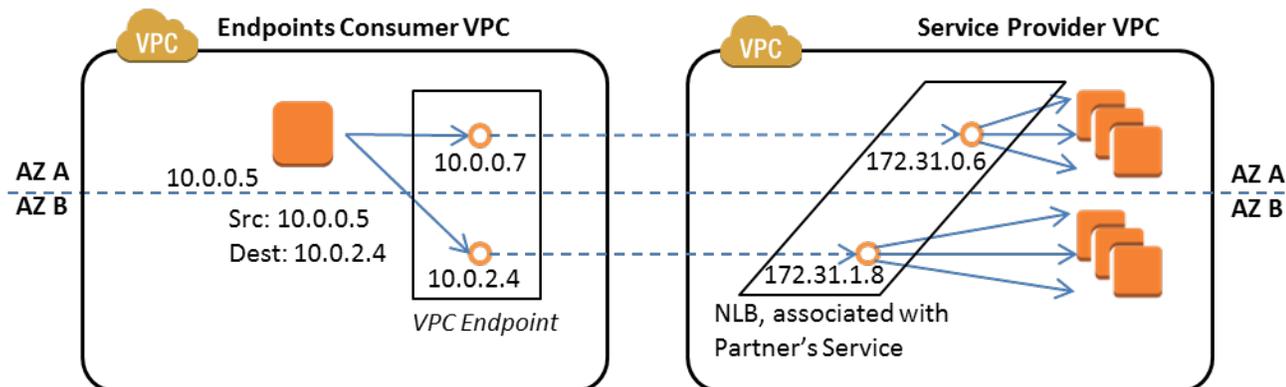
AWS Marketplace は AWS PrivateLink、Amazon ネットワークを使用して、を通じて販売する製品へのアクセスを購入者に提供するテクノロジーである をサポートします AWS Marketplace。このドキュメントでは、AWS PrivateLink テクノロジーを使用して Amazon Virtual Private Cloud (VPC) エンドポイントを介して製品を設定および配信するプロセスの概要を説明します。

このドキュメントでは、複数の AWS サービスと AWS Marketplace 環境に関する実務知識があることを前提としています。

序章

AWS Marketplace 販売者は、Amazon VPC エンドポイントを介して購入者にサービスへのアクセスを提供できます。このアプローチにより、購入者は [AWS PrivateLink](#) テクノロジーを使用して Amazon ネットワーク経由で販売者のサービスにアクセスできます。AWS Marketplace を使用してこのサービスを作成および配信する場合、購入者は でサービスを検出できます AWS Marketplace。

購入者は、VPC エンドポイントを作成するための使用可能なサービスのリストで、製品を見つけることもできます。



[VPC エンドポイント](#)は、インターネット、NAT デバイス、VPN 接続、または を介したアクセスを必要とせずに、AWS VPC と別の AWS サービスとの間にプライベート接続を作成できるようにする仮想デバイスです AWS Direct Connect。を使用してエンドポイントサービスを作成し AWS Marketplace、購入者がこのテクノロジーを使用してサービスに接続できるようにします。この接続方法は、インターネット経由ではなく、Amazon のプライベートネットワーク経由でサービスにアクセスするため、購入者にとってより安全です。

サービスを提供するリージョンごとに、サービスエンドポイントを作成することにより、リソースを新規作成するか既存のリソースを使用して VPC を設定し、サービスインスタンスと Network Load Balancer をセットアップして、サービスを Network Load Balancer に登録します。これらのステップを完了し、サービスをテストしたら、設定情報を [AWS Marketplace Seller Operations](#) チームに提供します。

AWS では、購入者が VPC エンドポイントを作成するときに使用できるプライベート DNS 名を指定することをお勧めします。

購入者は VPC エンドポイントを作成するときに、オプションとして、プライベート DNS 名を有効にすることができます。このオプションを選択すると、購入者の VPC サービスによって [プライベートホストゾーン](#) が設定されます。プライベート DNS 名を指定すると、購入者はこれを使用して販売者のサービスに接続するように VPC エンドポイントを設定できます。購入者のプライベートホストゾーンで、プライベート DNS 名 (api.example.com) は、エンドポイントサービス用にランダムに生成された DNS 名 (vpce-11111111111111111111111111111111-yyy-yyyyy.api.vpce.example.com) をポイントします。購入者の EC2 インスタンスは、複数の異なる VPC 間で同じ統一された DNS 名 (api.example.com) を呼び出します。また、パブリックとプライベートの DNS 名が同じであれば、購入者は VPC の内部または外部からサービスにアクセスする際に同じパブリック名を使用できます。

を通じてサービスを利用できるようにする方法については [AWS Marketplace](#)、[AWS Marketplace Seller Operations](#) チームにお問い合わせください。AWS Marketplace 購入者がサービスをサブスクライブして VPC エンドポイントを作成すると、サービスは AWS Marketplace Services の下に表示されます。AWS Marketplace Seller Operations チームは、VPC エンドポイントの作成時にサービスを簡単に検出できるように、ユーザーフレンドリーな DNS 名を使用します。

製品は SaaS(Software as a Service) 製品として作成されます。計測と請求は、他の AWS Marketplace SaaS 製品と同じです。

製品の設定

Amazon VPC エンドポイント経由で製品を使用できるように設定するには:

1. 既存の [Amazon VPC](#) を作成または使用します。
2. 製品の [Amazon EC2](#) インスタンス (複数可) を作成 (または既存のインスタンスを使用) します。
3. 製品を提供する各リージョンで [Network Load Balancer](#) を作成します。AWS では、あるリージョンのすべての [アベイラビリティーゾーン \(AZ\)](#) を含めることをお勧めします。
4. Amazon VPC コンソール、CLI、またはサポートされている SDK を使用して、VPC エンドポイントサービスを作成します。
5. Network Load Balancer 経由でサービスにアクセスできることを確認します。
6. わかりやすい DNS 名の [証明書を AWS Certificate Manager \(ACM\) にリクエスト](#) します。ACM は証明書を発行する前に、証明書リクエストのドメイン名の所有者または管理者を検証します。
7. `api.vpce.example.com` などのユーザーフレンドリーな DNS 名のサブドメインを、AWS Marketplace Seller Operations チームによって提供されるネームサーバーに委任します。DNS システムでは、DNS 名 (`vpce-0ac6c347a78c90f8.api.vpce.example.com` など) をパブリックに解決できるように、このサブドメインを AWS Marketplace Seller Operations チームが提供する Amazon Route 53 ネームサーバーを指すネームサーバー (NS) リソースレコードを作成する必要があります。
8. 購入者の AWS アカウントへのアクセスを許可します。

注: サポートされている SDK またはこの CLI コマンドを使用して、アカウントへのアクセスを自動化できます。`aws vpcev2 modify-vpc-endpoint-service-permissions --service-id vpce-svc-0123456789abcdef1 --add-allowed-principals arn:aws:iam::111111111111:root arn:aws:iam::222222222222:root`。

製品を に送信する AWS Marketplace

サービスを に公開するプロセス中に AWS Marketplace、AWS Marketplace Seller Operations チームと協力して作業します。PrivateLink対応製品を送信するには：

1. 次の情報を [AWS Marketplace Seller Operations](#) チームに E メールで送信します。
 - a. エンドポイントと、エンドポイントの作成に使用された AWS アカウント。エンドポイントは次のようになります: `com.amazonaws.vpce.us-east-1.vpce-svc-0daa010345a21646`
 - b. ユーザーが使いやすいサービスの DNS 名。これは、AWS Marketplace 購入者が製品にアクセスするために使用する DNS 名です。
 - c. 証明書のリクエストに使用した AWS アカウントと、購入者が VPC エンドポイントへのアクセスに使用するプライベート DNS 名。

Seller AWS Marketplace Operations チームは、登録するサービスに使用する会社の ID と DNS 名 (`api.vpce.example.com` など) を検証します。検証後、DNS 名はデフォルトのベースエンドポイントの DNS 名よりも優先されます。

VPC エンドポイントへの購入者のアクセス

AWS Marketplace VPC エンドポイントを作成する購入者は、次のような状況でサービスを検出できません。

- このページで前述した販売者のプロセスに従って、製品を新規作成したか、既存の製品を使用した場合。
- 購入者がサービスをサブスクライブした場合。
- 許可された AWS アカウントのリストに購入者のアカウントを追加しました。

購入者が VPC エンドポイントを作成する場合、オプションとして、プライベートホストゾーンを VPC と関連付けることができます。ホストゾーンにはサービスのデフォルトのプライベート DNS 名 (など) のレコードセットが含まれており、VPC のエンドポイントネットワークインターフェイスのプライベート IP アドレスに解決されます。

AWS Marketplace サービスを含む購入者がホストするエンドポイントは、すべてのアカウントにアクセス許可 (「*」アクセス許可) を付与できます。ただし、この方法を使用すると、サービス名で検索しない限り、Describe 呼び出しやコンソールにサービスが表示されません。サービスを Describe コールに表示するには、購入者の AWS アカウントをサービスによって許可リストに明示的に追加する必要があります。

サービスにアクセスする場合、購入者は次の操作を行います。

1. でサービスを検出してサブスクライブします AWS Marketplace。
2. AWS Command Line Interface (AWS CLI)、API、または Amazon VPC コンソールを使用してサービスを検出し、使用するサブネットと AZs します。エンドポイントは、サブネットに Elastic Network Interface として表示されます。ローカル IP アドレス、リージョンおよびゾーンの DNS 名がエンドポイントに割り当てられます。

クライアント側の DNS 名	名前
リージョン別	Vpce<0dc9a211a78c90f8>.api.vpce.example.com
IAD2 (1a)	us-east-1a-Vpce<0dc9a211a78c90f8>.api.vpce.example.com
IAD2 (1b)	us-east-1b-Vpce<0dc9a211a78c90f8>.api.vpce.example.com

デフォルトのプライベート DNS 名を指定し、購入者が VPC エンドポイントの作成時に [プライベート DNS 名を有効にする] (プライベートホストゾーンを関連付ける) を選択すると、サービスに接続するためのリージョンのデフォルトのプライベート DNS 名が購入者に表示されます。

名前	エイリアス	[Alias Hosted Zone ID]	(注意)
api.example.com	vpce<0dc9a211a78c90f8>.api.vpce.example.com	Z00AABBCCDD	IAD1 IAD2

付録: チェックリスト

次のチェックリストを使用して、AWS Marketplace Seller Operations チームに送信する前に、製品を設定してテストしていることを確認します。

製品の作成チェックリスト

- VPC を作成 (または既存を使用) し、設定します。
- VPC 内にネットワークロードバランサーを作成して設定します。
- VPC エンドポイントサービスを作成して、ネットワークロードバランサーにサービスを登録します。
- VPC エンドポイントの設定に使用した AWS アカウント ID を AWS Marketplace Seller Operations チームに提供します。
- デフォルトのエンドポイントサービス名 (com.amazonaws.vpce.us-east-1.vpce-svc-0bbb070044a2164 など) を AWS Marketplace Seller Operations チームに提供します。
- ランダムに生成されたサービス DNS 名を上書きするには、ユーザーの使いやすいサービス DNS 名 (必須) を指定します。ユーザーの使いやすいサービス DNS 名に使用されているサブドメインの ACM から SSL 証明書をリクエストします。これらの証明書と、それらを Seller Operations チームにリクエストするために使用した AWS アカウント ID AWS Marketplace を提供します。
- 推奨: プライベート DNS 名を提供します。
- AWS PrivateLink テクノロジーを使用してサービスに接続するオプションを AWS Marketplace 購入者に通知し、許可するプロセスを作成します。購入者の AWS アカウント ID を許可されたアカウントのリストに追加します。IDs

製品のテスト

- サービスが設定済みで、検出可能であることを確認します。
- サービスが Network Load Balancer を介して検出可能であることを確認します。
- 購入者が VPC エンドポイントを作成し、サービスにアクセスできることを確認します。サービスのセットアップに使用した AWS アカウントではない、所有しているアカウントを使用します。

プロフェッショナルサービス製品

販売者は、AWS Marketplace 購入者にプロフェッショナルサービスを提供できます。プロフェッショナルサービスには、AWS MarketplaceのAWSサービスおよび製品の使用方法に関して、評価、移行、サポート、管理、および他の人をトレーニングするサービスが含まれています。販売者は、提供するサービスを説明する製品オファーを作成し、顧客と交渉して契約条項を作成し、AWS Marketplaceを通じてサービスのカスタムオファーを作成します。

Note

独立系ソフトウェアベンダー (ISV)、チャネルパートナー、またはコンサルティングパートナーとして、チャネルパートナーのプライベートオファーを使用してプロフェッショナルサービス製品を再販することを別のパートナーに許可することもできます。詳細については、「[ISV としてのチャネルパートナーの再販オポチュニティの作成](#)」を参照してください。

購入者は、カテゴリでプロフェッショナルサービスを選択し、配信方法でプロフェッショナルサービスを選択し、パブリッシャー、料金モデル、料金単位で検索を絞り込むことで、AWS Marketplace カタログでプロフェッショナルサービス製品を検索できます。AWS 請求に含まれるサービスに対して課金されます。などのツールを使用して、支払い AWS Cost Explorer を一元化し、コストを管理できます。

プロフェッショナルサービス製品の詳細については、以下を参照してください。

- [プロフェッショナルサービス製品の開始方法](#)
- [プロフェッショナルサービス製品の詳細を提供する](#)
- [プロフェッショナルサービス製品の要件](#)
- [ISV としてのチャネルパートナーの再販機会の作成](#)

以下の動画では、AWS Marketplaceのプロフェッショナルサービス製品の管理について説明しています。

[AWS Marketplace のプロフェッショナルサービス製品の管理。](#)

ヘルプの利用

プロフェッショナルサービス製品に関するサポートについては、の事業開発パートナー AWS Marketplace または [AWS Marketplace Seller Operations](#) チームにお問い合わせください。

プロフェッショナルサービス製品の開始方法

このトピックでは、プロフェッショナルサービス製品の使用を開始する方法を説明し、最初の製品を作成する手順と、それを顧客に提供する方法について説明します。製品定義により、提供するサービスについて、またそれらのサービスのために会社を選択する必要がある理由について、顧客に指示されます。AWS Marketplace これにより、顧客はお客様に連絡できるようになります。契約に合意してから、固定料金でサービスを購入できるプライベートオファーを作成します。

トピック

- [前提条件](#)
- [プロフェッショナルサービス製品の作成](#)
- [プライベートオファーの作成](#)
- [製品情報の編集](#)
- [製品の料金の編集](#)
- [製品の可視性の編集](#)
- [プロフェッショナルサービス製品の削除](#)

以下の動画では、AWS Marketplaceのプロフェッショナルサービス製品の表示について説明しています。

前提条件

でプロフェッショナルサービスを販売するには AWS Marketplace、以下の前提条件を満たす必要があります。

- AWS Marketplace 管理ポータルにアクセスできます。これは、販売者として登録し、で販売する製品を管理するために使用するツールです AWS Marketplace。へのアクセスの詳細については、AWS Marketplace 管理ポータル「」を参照してください [AWS Marketplace 販売者のポリシーとアクセス許可](#)。
- AWS Marketplace 販売者として登録し、税金と銀行情報を送信します。販売者になる方法については、「[販売者としての開始方法](#)」を参照してください。

- のサービスまたは 内の少なくとも 1 つのパブリック製品に関連するプロフェッショナル AWS サービス製品を提供する必要があります AWS Marketplace。あなたの製品がそれらの製品を直接サポートするか、それらの製品のサブスクリプションを促進するサービスを提供している必要があります。

Note

製品は、評価、実装、マネージドサービス、プレミアムサポート、トレーニングのいずれかの主要なカテゴリに出品されている必要があります。

プロフェッショナルサービス製品のガイドラインの詳細については、「[プロフェッショナルサービス製品の要件](#)」を参照してください。

プロフェッショナルサービス製品の作成

次の手順では、AWS Marketplace 管理ポータルで新しいプロフェッショナルサービス製品を作成する方法について説明しています。

プロフェッショナルサービス製品を作成するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューから [プロフェッショナルサービス] を選択します。このページには、既に作成済みのすべてのプロフェッショナルサービス製品と、それらの製品の作成または変更のために送信したリクエストが表示されます。
3. [プロフェッショナルサービス製品] タブで、[プロフェッショナルサービス製品の作成] を選択します。
4. [製品の作成] ページで、製品の情報を入力し、[送信] を選択します。入力する必要がある詳細については、「[プロフェッショナルサービス製品の詳細の提供](#)」を参照してください。
5. (オプション) の「製品」メニューから AWS Marketplace 管理ポータル「プロフェッショナルサービス」を選択し、「リクエスト」タブを選択します。製品リクエストに正しい [製品のタイトル] が表示され、[リクエストの状態] が [レビュー中] になっていることを確認します。数分以内に製品が限定プレビューモードで作成されるはずですが。

Note

[プロフェッショナルサービス] ページの [リクエスト] タブに戻って、いつでもリクエストの状態を確認できます。作成プロセスでエラーが発生した場合は、ここに表示されます。リクエストを選択すると、リクエストの詳細を確認したり、エラーを修正したりできます。

製品が最初に公開された時点では、あなたの AWS アカウント (製品の作成に使用したアカウント) のみがアクセスできます。プロフェッショナルサービスページから製品を表示する場合は、[表示](#) を選択して AWS Marketplace、AWS Marketplace 購入者向けの [表示](#) に表示される製品の詳細を表示できます。この詳細リストは、プライベートオファーを拡張しない限り、AWS Marketplace 他のユーザーは利用できません。

製品を一般公開する方法については、「[製品の可視性の編集](#)」を参照してください。

プライベートオファーの作成

潜在的な購入者が [表示](#) で製品を表示する場合 AWS Marketplace、直接購入することはできません。サブスクライブしようとする、[プライベートオファー](#) をリクエストするようにリダイレクトされます。は、顧客がプライベートオファーをリクエストしたことを通知する E メールメッセージを AWS Marketplace 販売者アカウントのルートユーザーの E メールアドレス AWS Marketplace に送信します。次の手順では、このリクエストに応答する方法を説明します。

Note

を通じて 250,000 USD を超えるプライベートオファーを作成する場合 AWS Marketplace 管理ポータル、追加の承認が必要になる場合があります。詳細については、ビジネス開発担当者にお問い合わせるか、[表示](#) で AWS Marketplace ビジネスオペレーションチームに詳細を記載した E [メールメッセージを送信してください mpcustdesk@amazon.com](mailto:mpcustdesk@amazon.com) AWS Marketplace。

プロフェッショナルサービス製品のプライベートオファーを作成するには

1. リクエストに関して疑問がある場合は、顧客に連絡して解決してください。AWS Marketplace でプライベートオファーを作成する前に、オファー条件に同意します。購入者は製品を購入する義務はないため、オファーを作成する前に同意しておくのは理にかなっています。
2. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
3. メニューから [オファー] を選択し、[プライベートオファーの作成] を選択します。

4. [プライベートオファーの作成] ページで、プライベートオファーを作成する製品を選択します。現在入手可能な製品のみが含まれます。
5. プライベートオファーを延長する [購入者アカウント ID] を入力します。1つのプライベートオファーに最大 25 のアカウントを含めることができます。購入者がオファーのリクエスト機能を使用した場合、受信したメールメッセージには、リクエストするアカウントの購入者アカウント ID が含まれます。
6. 購入者に製品の分割払いをできるようにするかどうかを選択します。通常、短期契約は 1 回払いで支払われます。長期契約には分割払いのオプションがある場合がありますが、これは購入者との契約の一部です。[次へ] を選択します。
7. 以下の情報を含む、オファーの詳細を入力します。
 - [カスタムオファー名] - あなたと顧客がオファーを認識できるような詳細情報を入力します。会社名または製品名と製品の説明を含めます。名前、電話番号、住所など、個人を特定できる情報は含めないでください。
 - [契約終了日] - 同意したサービスの終了日。例えば、1 年間のサポートを提供する場合は、サービスが利用可能になる日から 1 年後の日付を入力します。
 - [製品ディメンション] - 提供しているサービスの料金と単位は次のとおりです。
 - 一括払いオファー - 各ディメンションとそれに関連する料金を記載できます (例えば、シルバー、ゴールド、プラチナというディメンションを設定できます)。購入者は希望に応じて選択して支払うことができます。
 - 支払いスケジュールを含むオファー - 1 つのディメンションを選択し、各支払いの金額と日付を記載した支払いスケジュールを提供する必要があります。

 Note

0 USD オファーを作成する場合は、確認のため [0 USD 料金を有効にする] を選択する必要があります。この予防措置は、誤って無料オファーを作成することを防ぐのに役立ちます。

- [サービス契約] - 顧客とのサービス契約を定義するドキュメント。アップロードするドキュメント (テキスト形式または PDF 形式) は 1 つの PDF 文書にまとめられるため、内容を理解するためにファイル名が不要であることを確認してください。
- [オファーの有効期限日] - オファーの有効期限が切れる日付です。これにより購入者がオファーを承諾するまでの期間が決まりますが、プロフェッショナルサービスがいつ利用可能になるかとは関係ありません。オファーの作成後にオファーの有効期限を延長できます。

8. オプションの編集が終わったら、[次へ] を選択します。
9. [オファーを確認] ページで、オファーの詳細が正しいことを確認し、[オファーを作成] を選択します。

Note

オファーが公開されるまで時間がかかる場合があります。公開されたオファーは、[オファーを管理] ページで確認できます。(まだ承認されていない) オファーを編集する必要がある場合は、そのページから編集できます。

10. オファーが公開され、[プライベートオファーの管理] ページに表示されたら、そのオファーの [アクション] メニューから [オファー URL をコピー] を選択し、購入者に E メールメッセージで送信して承認してもらいます。

製品情報の編集

次の手順では、AWS Marketplace 管理ポータル の既存のプロフェッショナルサービス製品の製品情報を編集する方法を説明します。

製品情報を編集するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューから [プロフェッショナルサービス] を選択します。このページには、既に作成済みのすべてのプロフェッショナルサービス製品と、それらの製品の作成または変更のために発行済みのリクエストが表示されます。
3. 編集する既存の製品を選択します。[変更をリクエスト] メニューから、[製品情報を更新] を選択します。
4. 詳細情報を変更します。編集できるフィールドの詳細については、「[プロフェッショナルサービス製品の詳細の提供](#)」を参照してください。
5. [送信] を選択してリクエストを作成します。
6. (オプション) プロフェッショナルサービスページのリクエスト タブにまだない場合は、の製品メニューから AWS Marketplace 管理ポータルプロフェッショナルサービス を選択し、リクエスト タブを選択します。リクエストに正しい [製品のタイトル] が表示され、[リクエストの状態] が [レビュー中] になっていることを確認します。数分以内に、リクエストした変更が製品に反映されます。エラーがある場合は、こちらで確認し、エラーを修正して編集内容を再送信することができます。

製品の料金の編集

次の手順では、AWS Marketplace 管理ポータル の既存のプロフェッショナルサービス製品の料金情報を編集する方法を説明します。

製品の料金を編集するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューから [プロフェッショナルサービス] を選択します。このページには、既に作成済みのすべてのプロフェッショナルサービス製品と、それらの製品の作成または変更のために送信したリクエストが表示されます。
3. 編集する既存の製品を選択してから、[変更をリクエスト] メニューから [料金設定ディメンションを更新] を選択します。

Note

新しい料金ディメンションは、からのみ追加できます AWS Marketplace 管理ポータル。以前に作成したディメンションを変更または削除するには、[AWS Marketplace Seller Operations チーム](#)に連絡してリクエストを伝えてください。リクエストには、製品 ID と、変更または削除を希望するディメンションの詳細を記載してください。

4. 必要な新しい料金ディメンションを追加します。料金フィールドの詳細については、「[プロフェッショナルサービス製品の詳細の提供](#)」を参照してください。
5. [送信] を選択してリクエストを作成します。
6. (オプション) の「製品」メニューから AWS Marketplace 管理ポータル「プロフェッショナルサービス」を選択し、「リクエスト」タブを選択します。リクエストに正しい [製品のタイトル] が表示され、[リクエストの状態] が [レビュー中] になっていることを確認します。数分以内に、リクエストした変更が製品に反映されます。エラーがある場合は、こちらで確認し、エラーを修正して編集内容を再送信することができます。

製品の可視性の編集

デフォルトでは、製品は公開範囲が限定され、新しい製品はあなたのアカウントにのみ表示されます。他のテストアカウントを追加、または製品を AWS Marketplace で公開することができます。次の手順では、AWS Marketplace 管理ポータル の既存のプロフェッショナルサービス製品の可視性を編集する方法を説明します。

製品の可視性を編集するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューから [プロフェッショナルサービス] を選択します。このページには、すでに作成したすべてのプロフェッショナルサービス製品が表示されます。
3. 可視性を編集または更新する既存の製品を選択します。[変更をリクエスト] メニューから、[製品の可視性の更新] を選択します。
4. 製品の可視性を更新するオプションを選択し、送信を選択してレビューリクエストを送信します。
5. [リクエスト] タブの [リクエストステータス] が [レビュー中] と表示されていることを確認します。リクエストが完了すると、ステータスが [成功] になります。ステータスが「失敗」の場合は、リクエスト名を選択して、「問題が見つかりました」を表示します。

Note

製品を公開 AWS Marketplace カタログに表示するには、AWS Marketplace Seller Operations チームによる製品レビューを行い、製品が製品ガイドラインを満たしていることを確認する必要があります (「」を参照[プロフェッショナルサービス製品の要件](#))。リクエストが完了するには数日かかる場合があります。

プロフェッショナルサービス製品の削除

次の手順では、AWS Marketplace 管理ポータルで既存のプロフェッショナルサービス製品を削除する方法について説明しています。

Note

プロフェッショナルサービス製品を削除しても、アクティブなプライベートオファーには影響しません。

製品を削除するには

1. ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#) にサインインします。
2. [製品] メニューから [プロフェッショナルサービス] を選択します。このページには、すでに作成したすべてのプロフェッショナルサービス製品が表示されます。

3. 削除する既存の製品を選択します。[変更をリクエスト]メニューから、[製品の可視性の更新]を選択します。
4. 製品の可視性を更新する可視性オプションとして「制限あり」を選択し、「送信」を選択します。
5. [リクエスト]タブの[リクエストステータス]が[レビュー中]と表示されていることを確認します。リクエストが完了すると、ステータスが[成功]になります。

Note

リクエストが完了するには数日かかる場合があります。アクティブなオファーを持つ製品は、最後のアクティブなサブスクリプションまたは契約が完了するまで制限付き状態に移行され、その後から削除されます AWS Marketplace。制限付きステータスは、既存のユーザーが製品を引き続き使用できることを意味します。ただし、製品は一般には公開されなくなり、新規ユーザーも利用できなくなります。

プロフェッショナルサービス製品の詳細の提供

プロフェッショナルサービス製品を公開するときは AWS Marketplace、製品メタデータを提供する必要があります。このトピックでは、製品の詳細を準備または編集するときに役立つ情報について説明します。

Note

プロフェッショナルサービス製品のガイドラインと要件については、「[プロフェッショナルサービス製品の要件](#)」を参照してください。

トピック

- [製品の説明](#)
- [追加リソース](#)
- [サポート情報](#)
- [料金ディメンション](#)
- [製品の可視性](#)

製品の説明

製品詳細の製品説明セクションは、製品の中核です。潜在的な購入者が購入を決定できるように、製品について説明しています。製品詳細のこのセクションには、以下のデータが含まれます。

- [製品のタイトル] - 製品の名前。これは製品を識別するために使用され、製品ページと検索結果に表示されます。製品に対して意味のある名前を選択します。内で一意である必要があります AWS Marketplace。
- SKU – (オプション) で製品を追跡するために使用されます AWS Marketplace。この情報はあなた自身が使用するもので、購入者には見えません。
- 簡単な説明 – 製品カタログの製品タイトルの下に表示される AWS Marketplace 製品の簡潔な説明。
- [詳しい説明] - 購入者に製品の詳細を説明する、長い形式の説明です。製品の機能、利点、使用方法、および製品に固有のその他情報を一覧表示します。情報を理解して容易に目を通せるようにするため、用意されている書式を使用してください。
- [製品のロゴ] - このフィールドは、あなたの製品を表すイメージファイルを指す公開 S3 URL です。ファイルは、.png、.jpg、.gif 形式で、背景は透明または白、5 MB 未満、幅と高さは 110 ~ 10,000 ピクセルである必要があります。ロゴは製品送信時にアップロードされ、に保存されます AWS Marketplace。URL の内容を変更しても、送信 AWS Marketplace 後に でロゴは変更されません。

Note

指定する S3 URL は公開されている必要があります。これは、ファイルが保存されている S3 バケットのプロパティです。詳細については、「Amazon Simple Storage Service コンソールユーザーガイド」の「[S3 バケットのパブリックアクセス設定を編集する方法](#)」を参照してください。

- [ハイライト] - 製品の主な機能や差別化要因を説明する、1~3 つの短い特徴です。少なくとも 1 つのハイライトが必要です。
- [製品カテゴリ] - 提供するサービスの種類です。少なくとも 1 つ、最大 3 つのカテゴリを選択する必要があります。選択できるカテゴリは多数ありますが、プロフェッショナルサービス製品には以下のうち少なくとも 1 つが含まれている必要があります。

評価

顧客の現在の運用環境を評価して、その組織に適したソリューションを見つけます。

実装

サードパーティー製ソフトウェアの設定、セットアップ、デプロイを支援します。

Premium Support

顧客のニーズに合わせて設計された、エキスパートによるガイダンスやサポートを受けることができます。

Managed Services

お客様に代わって End-to-end 環境管理を行います。

トレーニング

顧客の従業員がベストプラクティスを学ぶのに役立つ、エキスパートによるカスタマイズされたワークショップ、プログラム、教育ツールを提供します。

- [検索結果のキーワード] - 購入者が製品の検索に使用できるキーワードを 3 つまで提供します。キーワードは、カンマ区切りのリストで最大 250 文字まで指定できます。
- 関連製品 – オプション — サービス AWS Marketplace が または サポートのいずれかと連携するから少なくとも 1 つのパブリック製品を含めます。AWS Marketplace は、製品の詳細ページまたはそれらの製品の関連製品に表示する製品を選択するときに、これらの製品を入力として使用します。

追加リソース

製品詳細の [こちらをご確認ください] セクションには、顧客に役立つように作成したリソースへのリンクを記載できます。これは、オンラインで保存される 1~3 つのダウンロード可能なリソースのオプションセットです。リソースの例としては、製品情報シート、ホワイトペーパー、製品マニュアルなどがあります。リソースごとに、リソースの名前と URL を指定します。

サポート情報

このセクションはフォーマットされたテキストフィールドで、サービスに提供するサポートについて説明できます。

顧客は、サービスの使用、トラブルシューティング、返金のリクエスト (該当する場合) などの問題に関するサポートを期待します。サポートの説明には、顧客が期待できるサポートレベルに関する記述を含める必要があります。購入前の質問と購入後の問題の両方に関するサポートの詳細を含めることを検討してください。

料金ディメンション

プロフェッショナルサービスの料金ディメンションは、あなたが提供するパッケージです。例えば、シルバー、ゴールド、プラチナのサポートを提供する場合があります。または、10 時間、20 時間、または 50 時間のコンサルティングを提供することもできます。提供するディメンション (少なくとも 1 つ、最大 24 個) ごとに、名前と説明を指定します。購入者と直接やり取りして製品のプライベートオファーを作成する場合は、これらのディメンションの実際の料金を設定します。

Note

料金ディメンションの使用方法和料金の設定方法については、「[プライベートオファーの作成](#)」を参照してください。

製品の可視性

リリースされた製品は、自分のアカウント AWS Marketplace のみ、テストアカウントの小さなセット、またはすべての AWS アカウントで表示できます。デフォルトでは、製品はプライベートリリースで公開されます。製品の表示設定を変更するには、「[製品の可視性の編集](#)」を参照してください。

プロフェッショナルサービス製品の要件

AWS Marketplace は、のすべての製品およびサービスの要件を維持します AWS Marketplace。これらの要件は、顧客に安全、安心、および信頼できる、整理されたデジタルカタログを提供するのに役立ちます。販売者には、特定の製品のニーズを満たすために、必要に応じて追加の規制やプロトコルの導入を検討するよう奨励しています。

AWS Marketplace は、送信時にすべての製品とその関連メタデータを確認し、現在の AWS Marketplace 要件を満たしているか、超えているかを確認します。当社のセキュリティ要件を満たすため、私たちはこれらの要件を見直して調整します。さらに、は製品 AWS Marketplace を継続的に見直し、これらの要件に対する変更を満たしていることを確認します。製品がコンプライアンスに違反した場合、AWS Marketplace は、製品の更新についてお客様に連絡します。場合によっては、問題が解決されるまで、新しい受信者が製品を一時的に利用できないことがあります。

製品設定ガイドライン

すべてのプロフェッショナルサービス製品は、次の製品設定ガイドラインに従う必要があります。

- すべての料金ディメンションは、AWS Marketplaceを使用して提供および請求される実際のサービスに関連するものでなければなりません。
- 製品は、評価、実装、マネージドサービス、プレミアムサポート、トレーニングのいずれかのカテゴリに出品されている必要があります。
- 必要なプロフェッショナルサービスカテゴリ以外にも、提供するサービスに適合する他の適切なカテゴリを選択して、製品を正しく分類する必要があります。
- 製品のロゴは、AWS ロゴや無関係なサードパーティーのロゴと混同するように設計しないでください。
- 製品の詳細には、攻撃的または不適切な内容を含めないでください。これらは、<https://aws.amazon.com/aup/> で入手できる AWS 利用規定ポリシーに準拠している必要があります。
- プロフェッショナルサービス製品は、 にリストされているサードパーティーソフトウェア製品に関連するサービスを直接サポートまたは提供するか、お客様が の導入 AWS Marketplace または管理に関連する特定の成果を達成できるように支援する必要があります AWS クラウド。

顧客情報の要件

すべてのプロフェッショナルサービス製品は、次の顧客情報の要件に従う必要があります。

- プロフェッショナルサービス製品は、 に記載されているディメンションで完全に請求される必要があります AWS Marketplace。
- クレジットカードや銀行口座情報など、 のプロフェッショナルサービス製品リストの顧客支払い情報を収集 AWS Marketplace することは許可されていません。
- プロフェッショナルサービス製品 AWS と組み合わせて から提供されるサブスクライバーまたはサブスクライバー候補情報は、 AWS Marketplace 販売者向けの利用規約に従ってのみ使用する必要があります。

製品使用ガイドライン

すべてのプロフェッショナルサービス製品は、次の製品使用ガイドラインに従う必要があります。

- プロフェッショナルサービスのリストを通じて顧客から連絡があった場合は、2 営業日以内にその顧客に連絡する必要があります。顧客がプライベートオファーを受け入れると、プライベートオファーに説明がない限り、2 日以内にその顧客に連絡するか、次のステップを伝える必要があります。

- 顧客は、サービスの使用、トラブルシューティング、返金のリクエスト (該当する場合) などの問題について、簡単にサポートを受けることができる必要があります。サポートの問い合わせオプションは、フルフィルメントランディングページで指定する必要があります。サポートの説明には、顧客が期待できるサポートレベルに関する記述を含める必要があります。
- 製品の概要には、明確な価値提案、主な機能、詳細なドキュメントへのリンク、提供されるサービスの購入前および購入後のサポートの明確な定義を含める必要があります。
- 製品には明確でわかりやすいサービス条件が必要です。

アーキテクチャガイドライン

すべてのプロフェッショナルサービス製品は、次のアーキテクチャガイドラインに従う必要があります。

- プロフェッショナルサービス製品は、AWS サービスまたは少なくとも 1 つのパブリック AWS Marketplace 製品 (最大 4 つ) に関連する必要があります。この製品では、関連製品のサービスを直接提供するか、関連製品のサブスクライバーを増やすサービスを提供します。
- 提供されるサービスのために顧客のインフラストラクチャにリソースを追加する必要がある場合は、次のガイドラインに従う必要があります。
 - AWS Security Token Service または AWS Identity and Access Management (IAM) を使用するなど、安全な方法でリソースをプロビジョニングします。
 - プロビジョニングされたすべての AWS サービスの説明、IAM ポリシーステートメント、IAM ロールまたはユーザーがカスタマーアカウントでどのようにデプロイおよび使用されるかなど、追加のドキュメントを提供します。
 - 製品の説明に、顧客が AWS Marketplace トランザクションとは別に追加の AWS インフラストラクチャコストが発生した場合、追加のインフラストラクチャ料金を支払う責任があることを説明する通知を含めます。
 - 製品によってエージェントがデプロイされる場合は、そのエージェントを AWS アカウントでデプロイする方法の説明について顧客に提供する必要があります。

プロフェッショナルサービス製品の料金

プロフェッショナルサービス製品には、以下の製品料金モデルを設定できます。

- [プライベートオファー] - 販売者のプライベートオファーでは、複数年契約とカスタム期間契約のオプションがあります。複数年契約とカスタム期間契約の詳細については、[プライベートオファーの準備](#)および[分割プラン](#)を参照してください。

1つの製品につき1つの料金設定ができます。

AWS Marketplace 製品の料金の詳細については、「」を参照してください [製品の料金](#)。

データ製品

AWS Data Exchange は、AWS お客様が でファイルベースのデータセットを安全に交換できるようにするサービスです AWS クラウド。プロバイダーとして、AWS Data Exchange は、データ配信、エンタイトルメント、課金テクノロジーを構築して維持する必要を排除します。AWS Data Exchange のプロバイダーには、AWS 顧客にリーチし、既存の顧客にサブスクリプションをより効率的に付与するための、安全で透過的で信頼性の高いチャネルがあります。AWS Data Exchange プロバイダーになるためのプロセスでは、適格性を判断するためのステップがいくつか必要になります。

データ製品には、次の部分があります。

- 製品の詳細 - この情報は、サブスクライバーになる可能性があるユーザーが製品の内容を理解するのに役立ちます。これには、名前、説明 (簡単な説明と詳しい説明の両方)、ロゴ画像、およびサポート連絡先情報が含まれます。製品の詳細はプロバイダーによって記入されます。
- 製品のオファー - AWS Data Exchange で製品を利用できるようにするには、プロバイダーがパブリックオファーを定義する必要があります。これには、価格と期間、データサブスクリプション契約、返金ポリシー、カスタムオファーの作成オプションが含まれます。
- データセット - 製品には、1つ、または複数のデータセットを含めることができます。データセットは、ファイルベースのデータコンテンツの動的なセットです。データセットは動的で、リビジョンを使用してバージョン管理されます。各リビジョンには複数のアセットを含めることができます。

資格要件を含む詳細については、[AWS Data Exchange ユーザーガイドの「AWS Data Exchange でのデータ製品の提供」](#)を参照してください。

掲載のための製品の送信

製品を AWS Marketplace で使用可能にするには、製品送信プロセスを使用します。製品は非常にシンプルにすることができます。1 つの料金構造を持つ単一の Amazon Machine Image (AMI) がその例です。または、複数の AMIs、AWS CloudFormation テンプレート、複雑な料金オプションと支払いスケジュールなど、製品は非常に複雑になる可能性があります。製品提供を定義し、次の 2 つの方法のいずれか AWS Marketplace 管理ポータル で を通じて送信します。

- [製品] タブの使用 - 比較的複雑度が低い製品の場合、[製品] タブを使用して、リクエストを完全に定義し送信します。
- [アセット] タブの使用 - 複雑度が高く、より多くの定義を必要とする製品の場合、製品積載フォーム (PLF) をダウンロードし、製品の詳細を追加して、[ファイルのアップロード] オプションを使用して完成したフォームをアップロードします。

Note

データ製品プロバイダーは、AWS Data Exchange コンソールを使用して製品を公開する必要があります。詳細については、「AWS Data Exchange ユーザーガイド」の「[新製品の公開](#)」を参照してください。

最初は [製品] タブを使用して開始し、どちらの方法を使用するかを決定することをお勧めします。次の表では、リクエストの送信に使用する設定と方法が示されています。最初の列は製品の料金モデルです。他の 3 つの列では、製品が顧客にデプロイされる方法が示されています。

料金モデル	単一ノード AMI を使用して開始された製品	で起動される製品 AWS CloudFormation	Software as a Service (SaaS) として開始された製品
Bring-Your-Own-License (BYOL)	[製品] タブ	[アセット] タブ	
空き	[製品] タブ	[アセット] タブ	
時間単位払い	[製品] タブ	[アセット] タブ	

料金モデル	単一ノード AMI を使用して開始された製品	で起動される製品 AWS CloudFormation	Software as a Service (SaaS) として開始された製品
年間契約で時間単位での支払い	[製品] タブ	[アセット] タブ	
月額	[製品] タブ	[アセット] タブ	
月間で毎時	[アセット] タブ	[アセット] タブ	
有料使用料 (AWS Marketplace Metering Service)	[製品] タブ	[アセット] タブ	
契約料金	[製品] タブ		
SaaS サブスクリプション			[製品] タブ
SaaS 契約			[製品] タブ
SaaS レガシー			[アセット] タブ

製品を個別に送信することも、製品積載フォームを使用して複数の製品を送信したり、製品のアップデートを同時に実行したりすることもできます。[製品] タブを使用して同時に複数の製品を送信することはできません。どの製品をどの方法で送信するかが不明な場合は、[製品] タブを使用して開始することをお勧めします。送信で問題が発生した場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

[製品] タブの使用

[製品] タブにアクセスするには、AWS Marketplace 管理ポータルにログインします。[製品] タブで、管理する製品のタイプに応じて、[サーバー]、[SaaS]、または [機械学習] のいずれかを選択します。現在のすべての製品を含む、その製品タイプのダッシュボードが表示されます。[リクエスト] タブを選択すると、ダッシュボードには未処理のリクエストと、完了済みのリクエスト履歴が表示されます。新しい製品リクエストの作成を開始したら、進行中の作業を保存し、必要に応じていくつかの異なるセッションでリクエストを作成することができます。

製品リクエストを送信する準備ができると、リクエストは AWS Marketplace チームにより確認されます。リクエストしている製品タイプの製品ページで、リクエストのステータスをモニタリングできます。新しい製品の場合、リクエストの公開が承認されると、限定掲載の URL が届きます。この URL を使用して送信をプレビューし、承認します。製品オファーは、送信を承認するまで公開されません。既存の製品に対する更新をリクエストした場合、変更を確認し承認する必要なく、更新が公開されます。これには、バージョンの追加または削除、メタデータの変更などが含まれます。

リクエストのステータスは [リクエスト] タブで追跡します。ステータスは次のいずれかになります。

- [下書き] - リクエストプロセスを開始しましたが、リクエストを送信していません。
- [送信済み] - リクエストを完了および送信して確認中です。
- アクションが必要 - AWS Marketplace チームはリクエストを確認し、詳細情報が必要です。
- 承認が必要 - AWS Marketplace チームは製品の限定リスト URL を作成しました。AWS Marketplace が公開する前に URL を確認して、承認または却下する必要があります。承認すると、サイトの公開中にステータスが公開保留中に変ります。却下すると、ステータスは下書きに戻り、リクエストを変更することができます。
- 公開保留中 - リクエストのモックアップを承認し、製品を AWS Marketplace 公開しています。
- [期限切れ] - リクエストプロセスを開始しましたが、6 か月以内に完了しなかったため、要求が期限切れになりました。

送信済みというステータスのエントリがある場合は、送信を取り消すことができます。下書きというステータスのエントリがある場合は、リクエストを取り消すことができます。これにより、最初からやり直すことができます。下書きのエントリを削除すると、エントリはリクエスト履歴タブに移動します。

に製品を追加するには AWS GovCloud (US) AWS リージョン、[アクティブな AWS GovCloud \(米国\) アカウントがあり](#)、エクスポートコントロールの要件を含む要件に準拠 AWS GovCloud (US) している必要があります。

会社と製品のロゴ要件

AWS Marketplaceをブラウジングした際のユーザーエクスペリエンスに統一感を持たせるため、顧客の企業ロゴおよび製品ロゴは次の AWS Marketplace のガイドラインに準拠する必要があります。

製品ロゴ仕様 - 製品のロゴイメージは、背景は透過または白、120~640 ピクセルのサイズで比率は 1:1 または 2:1 (幅) である必要があります。

企業ロゴ仕様 - 企業ロゴイメージは、背景が透過で、220 x 220 ピクセルのサイズ、すべての辺の内側に 10 ピクセルの余白を持たせることができます。

有料の再パッケージ版ソフトウェアを送信するための要件

再パッケージされたオープンソースソフトウェア (例えば、オープンソースの AMI や有料サポート付きのコンテナ製品)、または自分以外のベンダーが元々作成したソフトウェア (Windows オペレーティングシステムで AMI を再販するなど) の有料リストを送信する場合は、送信前に次の要件を満たす必要があります。

- 製品のタイトルには、再パッケージ化による付加価値を示す必要があります。製品のタイトルの例: 強化済み <製品>、パッケージが追加された <製品>、<製品 2> 上の <製品 1>
- 製品のタイトルには、ドキュメントでサポートされていない他の言語を含めないでください。例えば、製品のタイトルに認定、オリジナル、無料といった言葉を使用することはできません。ただし、提供する製品詳細でこれらが裏付けられている場合を除きます。
- 商品の簡単な説明には、商品の料金を要約した明確な説明を含める必要があります。簡単な説明は、この商品には...の料金がかかりますというフレーズで始める必要があります。例えば、製品に販売者からのサポート料金が含まれている場合、製品説明に「この製品には販売者サポートの料金がかかります」と明記する必要があります。
- 製品のロゴは、販売者登録時に使用した会社のロゴと同じでなければなりません。製品のロゴが会社のロゴと異なる可能性があるのは、正式なソフトウェアロゴを使用する場合のみで、元のソフトウェアベンダーから明示的な許可を受ける必要があります。明示的な許可を得た場合は、変更リクエストのメモセクション (または製品ロードフォームを使用する場合は [ファイルのアップロード] ページの [簡単な説明を入力] フィールド) にそのドキュメントへのリンクを含める必要があります。
- AMI 製品の場合、AMI 名を元の製品から再利用することはできません。AMI 名は販売者名で始まり、[販売者名] [name-given-to-ami] の形式に従う必要があります。

有料表示が自社製ではないスタンドアロンソフトウェア製品に関するもので、製品に知的財産が追加されていない (例えば、追加のソフトウェアライブラリのバンドルや特別な設定の追加など) 場合は、前述の要件に加えて、次の要件も満たす必要があります。

- 製品のタイトルには、販売者名 (前述の付加価値と共に) を含める必要があります。販売者名とは、販売者登録時に使用する名前です。例えば、<販売者> によるメンテナンスサポートがある <製品> などです。

- 製品の詳細な説明の最初の行は、これは再パッケージされたソフトウェア製品で、追加料金がかかります...というフレーズで始まる必要があります。(または、オープンソースの場合、これは再パッケージされたオープンソースソフトウェア製品で、追加料金がかかります...)。また、詳細な説明には、課金対象を要約した明確な説明と、それらの機能を説明する追加の詳細を記載する必要があります。例えば、追加サポートの料金を請求するオープンソース製品の長い説明は、これは再パッケージされたオープンソースソフトウェア製品であり、{SLA の詳細} のサポートには追加料金が適用されますのように始まる場合があります。

ハードウェアコンポーネントを使用する製品の要件

ハードウェア製品の販売は、では許可されていません AWS Marketplace。ハードウェアコンポーネント (SIM カード、スマートデバイス、IoT デバイス、センサーなど) を必要とするソフトウェア製品を送信する場合は、次の要件を満たす必要があります。

- ハードウェアコンポーネントを で販売することはできません AWS Marketplace。
- ハードウェアコンポーネントのコストを製品の出品価格に含めることはできません。
- リストの製品概要セクションには、次のステートメントを含める必要があります。このリストで必要となる可能性のあるハードウェアは、個別に取得する必要があります。詳細については、製品の詳細を確認してください。

AWS CloudFormationが発表した製品 (無料または有料) または使用量ベースの有料 AMI 製品

製品ロードフォーム (PLF) を使用して、AWS Marketplace 顧客が AWS CloudFormation テンプレートを使用して起動する製品を送信します。PLF は、AWS Marketplace 管理ポータル (AMMP) を通じて利用できます。

製品の送信

1. [AMMP](#) から、製品の製品積載フォーム (PLF) をダウンロードします。
2. 製品情報 (タイトル、説明、ハイライト)、技術情報 (AMI_ID、リージョン、インスタンスタイプ、OS)、および料金詳細 (料金モデル、無料トライアル) をなどの製品定義を追加します。
3. スプレッドシートの指示テーブルの指示に従って PLF を送信してください。

AWS Marketplace チームは、ポリシーとセキュリティのコンプライアンス、ソフトウェアの脆弱性、製品の使いやすさについて製品を確認します。リクエストについて質問や問題がある場合、AWS Marketplace チームは E メールメッセージを通じてお客様に連絡し、リクエストについて話し合います。承認されると、製品のページのモックアップが作成されます。ページを確認したら、モックアップを承認または却下します。承認されると、ページが AWS Marketplace に追加されます。

製品の更新

製品ロードフォーム (PLF) を使用して作成した製品の場合は、PLF を使用して製品に変更を加えます。完了した元の PLF を変更することができます。利用できない場合は、新しい PLF から開始することもできます。[製品] タブを使用するのと同様に、新しいバージョンの追加や既存のバージョンの削除をしたり、さらに料金、インスタンスタイプ、リージョンの可用性、メタデータを更新することができます。更新を行うには、新製品の場合と同じ方法で、更新された製品を準備します。製品アップデートの準備が整ったら、以下のステップに従います。

1. 既存の PLF を使用するか、[アセット] タブの、[AWS Marketplace 管理ポータル](#) から [ファイルのアップロード] を選択します。[製品積載フォームおよび販売者ガイド] で、製品の PLF をダウンロードできます。
2. PLF で製品送信を更新します。
3. [AWS Marketplace 管理ポータル](#) の [アセット] タブで、[ファイルのアップロード] をクリックします。
4. [ファイルのアップロード] ページで、更新した PLF と AWS CloudFormation テンプレートをアップロードします。ファイルアップローダーは、安全な転送メカニズムと送信されたファイルの履歴を提供します。アップローダーは、リクエストの処理を開始するように AWS Marketplace チームに自動的に通知します。送信の説明を追加します (新しいバージョンの追加、価格の変更、メタデータの変更など)。

製品の送信は、ポリシーとセキュリティのコンプライアンス、ソフトウェアの脆弱性、および製品のユーザビリティについて確認されます。リクエストについて質問や問題がある場合、AWS Marketplace チームは E メールメッセージでお客様に連絡します。既存の製品ページに対する更新は、追加の確認を受けずに直接処理されリリースされます。

製品の変更と更新

販売者はいつでも製品の変更を送信することができ、前述のように処理されます。ただし、90 日または 120 日ごとに、または変更が保留中の場合にのみ変更を加えることができます。例としては、

料金変更 AWS リージョン やインスタンスタイプの変更などがあります。一般的な変更は次のとおりです。

- 新バージョン - ソフトウェアの新バージョン、パッチまたはアップデートのロールアウト。お客様のリクエストに応じて、新しいバージョンの可用性について AWS Marketplace コンテンツをサブスクライブしているお客様に通知したり、お客様に代わってアップグレード手順を送信したりできます。
- メタデータの変更 - 製品情報の変更 (説明、URL、および使用方法)。
- 料金の変更 - 料金設定額の変更。リクエストが完了した後に、現在のお客様への通知が送信されます。通知が送信されると、料金変更は 90 日間の期間の翌月の 1 日に有効になります。例えば、3 月 16 日に変更を行うと、90 日後の 6 月 16 日ごろになりますが、料金変更は翌月の初日に行われます。実際の変更日は 7 月 1 日です。
- 料金モデルの変更 - 料金モデルの変更 (例えば、毎時、無料、年間で毎時)。すべての料金モデルの変更がサポートされているわけではなく、モデルを変更するリクエストはすべて AWS Marketplace チームで確認および承認されなければなりません。無料モデルから有料モデルへの変更は、既存のお客様に大きな影響を与えます。もう 1 つの選択肢は、追加機能を備えた新しい製品を提案し、現在のお客様に移行を促すことです。
- リージョンまたはインスタンスの変更 - インスタンスのタイプやリージョンの追加または削除。
- 製品のテイクダウン - から製品ページを削除 AWS Marketplace して、新規顧客がサブスクライブしないようにします。リクエストが完了した後に、現在のお客様への通知が送信されます。

タイミングと心構え

リクエストをできるだけ早く処理するよう努めていますが、リクエストには複数の反復処理と、販売者と AWS Marketplace チームによるレビューが必要になる場合があります。プロセスを完了するまでの時間については、以下のガイダンスを参考にしてください。

- 通常、合計リクエスト時間はカレンダー時間の 2~4 週間です。複数の繰り返しと製品メタデータおよびソフトウェアの調整により、リクエストや製品が複雑なほど時間がかかることがあります。
- Amazon では、リクエストの優先順位をつけるために、計画されているイベントまたはリリースの少なくとも 45 日前に、完了した製品リクエストと AMI が必要です。

リクエストに関するご質問がございましたら、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

への AMIs の送信 AWS Marketplace

に構築および送信されるすべての AMIs AWS Marketplace は、すべての製品ポリシーに従う必要があります。送信する前に、AMI の最終チェックをいくつかお勧めします。

- システムから、すべてのデフォルトのパスワード、認証キー、キーペア、セキュリティキー、またはその他の認証情報など、すべてのユーザー認証情報を削除します。
- ルートログインが無効になっている、またはロックされていることを確認します。sudo アクセスアカウントのみが許可されます。
- リージョンに AWS GovCloud (US) デプロイする AMI を送信する場合は、[アクティブな AWS GovCloud アカウントを持ち](#)、該当するエクスポートコントロール要件を含む [AWS GovCloud 要件](#) に同意する必要があります。

AMI セルフサービススキャン

AWS Marketplace 管理ポータル内ではセルフサービス AMI スキャンを利用できます。この機能を使用すると、AMI のスキャンを開始し、スキャン結果を通常 1 時間以内に迅速に受信でき、1 か所で明確なフィードバックを得ることができます。

この新しいセルフサービススキャンで AMI を共有してスキャンするには、次のようにします。

1. <https://aws.amazon.com/marketplace/management/manage-products/> に移動します。
2. 共有する AMI を選択します。
3. スキャン結果を表示します。

AMI が正常にスキャンされたら、現在のプロセスに従って製品ロードフォーム (PLF) [をアップロード](#) して AWS Marketplace Seller Operations チームに送信できます。問題がある場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

AMI をセルフサービススキャンリストに含めるには、AMI が us-east-1 (バージニア北部) リージョンにあり、AWS Marketplace 販売者アカウントによって所有されている必要があります。他のアカウントにへのアクセスを許可する必要がある場合は AWS Marketplace 管理ポータル、それらのアカウントを販売者として登録する必要があります。詳細については、「[販売者登録プロセス](#)」を参照してください。

AMI クローン作成と製品コード割り当て

AMI が送信されると、 は、ソフトウェアが利用可能であることを示すリージョンごとにクローンされた AMIs AWS Marketplace を作成します。このクローン作成および発行プロセス中に、 AWS Marketplace はクローン AMI に製品コードをアタッチします。製品コードは、アクセスの制御と使用量の測定の両方に使用されます。すべての送信は、この AMI クローン作成プロセスを経なければなりません。

最終チェックリスト

製品の公開が遅れるのを防ぐには、製品リクエストを送信する前にこのチェックリストを使用してください。

製品の使用

- 本稼働の準備完了。
- 時間またはその他の制限による製品の使用を制限しません。
- 1-Click 履行体験との互換性。
- 製品を使用するために必要なものはすべて、クライアントアプリケーションを含むソフトウェアに含まれます。
- デフォルトのユーザーはランダム化されたパスワードを使用し、初期ユーザーの作成では、インスタンス ID などのインスタンスに固有の値を使用してインスタンスを使用する権限が購入者に付与されていることを確認する必要があります。

無料製品にするか、有料製品にするか

- 製品を使用するために追加のライセンスは必要ありません。
- 有料の再パッケージソフトウェアは、 を満たしています AWS Marketplace [有料の再パッケージ版ソフトウェアを送信するための要件](#)。
- 購入者は、製品を使用するために個人を特定できる情報 (E メールアドレスなど) を提供する必要はありません。

AMI の準備

- ハードウェア仮想マシン (HVM) の仮想化と 64 ビットアーキテクチャを活用します。
- 既知の脆弱性、マルウェア、またはウイルスは含まれていません

- 購入者には、AMI へのオペレーティングシステムレベルの管理アクセス権があります。
- AMI セルフサービススキャンを使用して AMI を実行します。

Windows AMI の場合

- 「Amazon EC2 ユーザーガイド」の「[EC2Configサービスを使用したWindowsインスタンスの設定](#)」で説明されているように、最新バージョンの `Ec2ConfigService` を使用します。Amazon EC2
- `Ec2SetPassword`、`Ec2WindowsActivate`、および `Ec2HandleUserData` プラグインは、「Amazon EC2 [ユーザーガイド](#)」の「[EC2Configサービスを使用したWindowsインスタンスの設定](#)」で説明されているように有効になっています。Amazon EC2
- ゲストアカウントまたはリモートデスクトップのユーザーアカウントが存在しないことを確認します

Linux AMI の場合

- ルートログインはロックされ、無効になっています。
- 承認されたキー、デフォルトのパスワード、またはその他の認証情報が含まれません。
- すべての必須フィールドが入力されています。
- すべての値が指定された文字制限内にあります。
- すべての URL がエラーなしで読み込まれます。
- 製品イメージは、少なくとも 110 ピクセル幅で、1:1 ~ 2:1 の比率です。
- 料金は、有効なすべてのインスタンスタイプ (料金モデルが `hourly`、`hourly_monthly`、および `hourly_annual`) で指定されます。
- 月単位料金設定が指定されています (`hourly_monthly` および `monthly pricing` モデルの場合)。

自動 AMI 構築に関する質問やコメントがある場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

製品のマーケティング

製品の成功に貢献するには、 の認識を高め AWS Marketplace、 の製品ページに直接トラフィックを誘導します AWS Marketplace。以下は、 AWS Marketplaceに出品した製品の販売に役立つ情報とサポートを提供します。

180 日間の GTM アカデミー

180 日間の GTM ワークショップはすべての AWS Marketplace 販売者が利用でき、セルフサービス go-to-market (GTM) リソースを提供し、でのサービスの需要生成キャンペーンを構築、アクティブ化、追跡するのに役立ちます AWS Marketplace。次のようにできます。

- リスティングの需要を迅速に追跡できます。
- と AWS Marketplace 価値提案をメッセージングに統合することで、マーケティングの投資収益率を向上させ AWS、カスタマーメッセージングを強化します。
- [AWS Marketplace 販売促進プログラムガイドプログラムガイド](#)の達成に向けて、またはその範囲内で進めることができます。

180 日間の GTM アカデミーには、[AWS Marketplace 管理ポータル](#)の [Marketplace リソース] セクションからアクセスできます。

製品が入手可能になったことの発表

での製品の販売状況を広く発表することをお勧めします AWS Marketplace。これは、プレスリリース、ツイート、ブログ、その他の任意のメディアチャネル経由で行うことができます。含まれる可能性のあるサンプルテキストと、商標の使用に関するガイドラインと指示、およびプレスリリースの発行について説明しました。

公開する前に、ブログ、ツイート、その他の非プレスリリースの発表を確認し、AWS メッセージングやブランドガイドライン、音声との一貫性を確保します。レビューのリクエストを AWS アカウントマネージャーに送信します。レビューが完了するまでに、最大で 10 営業日かかります。ツイート、ブログ、またはプレスリリースを投稿したらお知らせください。その可視性を高めるために再掲載できるよう最善を尽くします。

AWS Marketplace メッセージング

顧客とのコミュニケーションでは、AWS Marketplaceを使用して製品を購入する目的、目標、および利点について説明することができます。AWS Marketplaceを参照するときは、次のメッセージを使用します。

とは AWS Marketplace

AWS Marketplace は、お客様が で実行されるソフトウェアとサービスを簡単に検索、比較、すぐに使用を開始できるようにするオンラインストアです。AWS の訪問者 AWS Marketplace は 1-Click デプロイを使用して、事前設定されたソフトウェアをすばやく起動し、使用した分に対してのみ時間または月単位で料金を支払うことができます。は請求と支払い AWS を処理し、ソフトウェア料金は顧客の AWS 請求書に表示されます。

顧客が でショッピングする理由 AWS Marketplace

ソフトウェアの検索とデプロイは難しい場合があります。データベースやアプリケーションサーバー、IoT ソリューション、デベロッパーツール、ビジネスアプリケーションなどのソフトウェアインフラストラクチャなど、商用および無料の IT およびビジネスソフトウェアを幅広く AWS Marketplace 提供しています。の AWS Marketplace お客様は、オプションの比較、レビューの閲覧、必要なソフトウェアをすばやく見つけることができます。その後、1-Click または AWS Marketplace 管理ポータルを使用して、独自の Amazon Elastic Compute Cloud インスタンスにデプロイできます。

ソフトウェア価格は Web サイトに明確に掲載されており、既にアマゾン ウェブ サービスに登録されている支払い方法で、お客様はほとんどのソフトウェアを即座に購入することができます。ソフトウェア料金は、AWS インフラストラクチャ料金と同じ月額料金に表示されます。

ソフトウェアまたは SaaS 販売者が で販売される理由 AWS Marketplace

AWS では AWS Marketplace、 で実行されるサービスを持つ Software and Software as a Service (SaaS) 販売者は、顧客認識の向上、デプロイの簡素化、自動請求の恩恵を受けることができます。

AWS Marketplace は、個々のソフトウェアデベロッパーから大企業まで、数十万の顧客に製品を公開することで、 で実行されるソフトウェアおよびサービスのソフトウェアおよび SaaS 販売者が新しい AWS 顧客 AWS を見つけるのに役立ちます。

を販売 AWS Marketplace することで、独立系ソフトウェアベンダー (ISVs、コストのかかるコード変更を行わずに、ソフトウェアの時間単位の請求を追加できます。Amazon マシンイメージ (AMI)

を にアップロード AWS して時間単位のコストを提供するだけです。請求は によって管理され AWS Marketplace、販売者は使用量の計測、顧客アカウントの管理、支払いの処理を行う責任を解放されるため、ソフトウェア開発者は優れたソフトウェアの構築に集中する時間が増えます。

さらに、お客様にとっては、ソフトウェアの事前構成済みイメージを簡単にデプロイして、新規お客様のオンボーディングを簡素化することができます。

のレビュー AWS Marketplace

AWS Marketplace は、顧客が製品のレビューを送信できるようにします。また、独立した製品レビューをキュレートする business-to-business マーケットプレイスである G2 などのシンジケートレビューワーが、シンジケートレビューを に統合する機能も提供します AWS Marketplace。

AWS Marketplace カスタマーレビューは、購入者向けユーザーガイドに記載されているレビューガイドラインを満たす必要があります。レビュー送信は、レビューがレビュー基準を満たしていることを確認するために AWS Marketplace レビューされるまでにリリースされません。レビューガイドラインについて詳しくは、「[製品レビュー](#)」を参照してください。シンジケートレビュー組織は独自のプロセスを使用してレビューを検証し、リリース AWS Marketplace 前に によってレビューされません。製品の配信レビューが商品レビューガイドラインを満たしていないと思われる場合や、商品のレビューに好ましくないコンテンツが含まれていると思われる場合は、[AWS Marketplace Seller Operations](#) にお問い合わせください。

レビュー担当者は、5 つの星評価システムに基づいて製品の星評価を提供することもできます。評価は、製品の全体的な星評価を算出するために平均されます。シンジケートレビューには星の評価を含めることもできますが、シンジケートレビューの星の評価は、AWS 顧客の星の評価では平均化されません。

製品レビュー機能に関するその他の重要なポイントを以下に示します。

- から製品レビューを削除することはできません AWS Marketplace。ただし、コメントが製品レビューを管理するレビュー基準を満たしている限り、任意のレビューにコメントを残すことができます。
- レビューがレビューのガイドラインを満たしていないか、好ましくないコンテンツが含まれていると思われる場合は、[AWS Marketplace Seller Operations](#) にお問い合わせになり、懸念事項を説明してください。
- AWS で製品を検索するお客様は、評価、検証済みレビュー、外部ソースレビューに基づいて結果を検索およびフィルタリング AWS Marketplace できます。 の AWS お客様は、検索結果に AWS 顧客評価とともに外部ソース評価が表示されます。

- 製品のシンジケートレビューは AWS Marketplace、無料で自動的に追加されます。レビューは自動的に追加されるため、シンジケートレビューを追加するためにリクエストを送信する必要はありません。
- 製品のシンジケートレビューがない場合は、シンジケートレビュー担当者に連絡し、そのプロセスに従って製品のレビューを受けることができます。たとえば G2 では、そのウェブサイトアクセスし、製品ページに移動してレビュープロセスを開始できます。

へのリンク AWS Marketplace

企業のウェブで、製品について説明し、促進する存在感を出すことができます。製品は で実行でき AWS、 を使用して購入できることを強調することをお勧めします AWS Marketplace。お客様がソフトウェアを発見してデプロイするプロセスを簡略化するため、お客様の製品を結びつけるための手順について説明しました。

AWS Marketplace ロゴの使用

AWS Marketplace ロゴは、ソフトウェアが で AWS 実行され、 で利用できることを顧客に簡単に伝える方法です AWS Marketplace。でソフトウェアを昇格させる場合は AWS Marketplace、アーカイブフォルダ (.zip ファイル) をダウンロードします。このフォルダには、複数の色処理とファイル形式が含まれています。AWS 商標ガイドラインに基づく義務についてご質問がある場合は、[AWS 「商標ガイドラインとライセンス条項」](#)を参照してください。

で製品に直接リンクする AWS Marketplace

ウェブサイトにディープリンクを含めるか、リダクション AWS Marketplace することで、 の製品情報ページに顧客を直接送信できます。ブラウザベースのリンクには、次のリンク構造の例を使用します。

```
https://aws.amazon.com/marketplace/pp/ASIN
```

URL の ASIN セグメントを製品の ASIN に置き換えます。

Example

```
https://aws.amazon.com/marketplace/pp/B00635Y2IW
```

aws.amazon.com/marketplace でアプリケーションを検索すると、URL に ASIN が表示されます。または、アカウントマネージャーに相談して ASIN を見つけることができます。

Note

リンクを使用する前にテストして、お客様が正しいページに移動していることを確認してください。

プレスリリース

任意のチャネル AWS Marketplace を通じて、の製品の可用性を発表することをお勧めします。ただし、公開または発表を行う前に、Amazon が参照するすべてのプレスリリースを確認してサインオフ AWS Marketplace する必要があります。発表することをお勧めしますが、AWS Marketplace 販売者との共同プレスリリースはサポートできません。からの引用を含むプレスリリースは、case-by-case ベースでサポートされます AWS。見積りは、にリストされている新しい製品やサービスを発表したり、を使用する顧客リファレンスを含む AWS Marketplace など、いくつかの条件を満たす必要があります AWS Marketplace。

すべてのプレスリリースは、お客様が作成しなければなりません。次の見出しをお勧めします: [製品名を挿入] が で利用可能になりました AWS Marketplace。整合性を保つために、このドキュメントのメッセージを使用します。

プレスリリースは、以下のようにする必要があります:

- 発表が Amazon.com にどのように関連しているかを明確かつ正確に説明する
- 顧客とのロールを明確に AWS する
- お客様を重視し、お客様の利益を高める

プレスリリースは、以下のことは禁止です:

- 関係を表す際、パートナー、パートナーシップ、アライアンス、「協業」、「提携」などの用語を使用する。合意、連携、関係をお勧めする。
- 事前合意なしに、アマゾン ウェブ サービス エグゼクティブからの引用を含めません。
- 販売予測を含めるか、企業共通のウェブサイトを参照する場合を除き、出品者ごとに「.com」を使用する。

- Amazon.com の「アソシエイト」としてお客様の組織を参照する。これは、オンラインアフィリエイトプログラムの Amazon Associates と混同する可能性があるためです。
- Amazon.com に関する専有情報を開示するか、弊社の株式ティッカーシンボルを参照する。

プレスリリースをテキスト形式でアカウントマネージャーに送信して、レビューしてもらいます。さらに、商標を使用する前に、[Amazon Web Services の商標ガイドライン](#)を確認してください。AWS Marketplace 商標に固有のガイドラインは、以下のセクションに記載されています。

AWS Marketplace 商標使用ガイドライン

本ガイドラインは、Amazon.com, Inc. および/またはその関連会社 (Amazon) によって事前に承認された資料における AWS Marketplace ロゴと商標 (それぞれ「商標」と総称して「商標」) の使用に適用されます。本ガイドラインはどのようなときにも厳格に厳守していただくものとし、本ガイドラインに違反して商標を使用すると、商標の使用に関連するライセンスは自動的に解除されます。

1. お客様は、Amazon が明示的に許可する目的でのみ、本商標を使用することができます。また、お客様は、(i) お客様による本商標の使用に関して、Amazon と up-to-date のすべての契約 (総称して「契約」) の最新バージョンを遵守すること、(ii) 本ガイドライン up-to-date の最新バージョンを遵守すること、および (iii) Amazon が随時発行することがある、本商標の使用に適用されるその他の条件またはポリシーを遵守する必要があります。
2. 弊社はお客様が使用できる承認された商標イメージを提供します。お客様は、以下を含みますがこれらに限定されない、商標の割合、色、フォントの変更、商標の追加や削除など、どのような方法によっても商標を変更することはできません。
3. 本契約で特別に許可された商標を使用する以外に、Amazon による後援または裏書を意味するような方法で商標を使用することはできません。
4. Amazon、その製品またはサービスを誹謗するために商標を使用することはできません。また商標の Amazon の信用をおとしめたり、損なったり、傷つけたりする方法を使用することはできません。
5. 商標は、商標の各面とその他のビジュアル、グラフィックまたはテキストの要素間に適度な間隔を置いて、単独に表示されなければなりません。どのような場合でも、商標は、商標の可読性または表示を妨害するような背景に置かれてはなりません。
6. AWS Marketplace AWS Marketplace 「ロゴは Amazon.com, Inc. またはその関連会社の商標です。

- お客様は、商標に対するすべての権利は Amazon の独占的財産であり、商標の使用を通じて構築されたすべての信用は、Amazon のメリットのために効力が生ずることを認めます。商標に関する Amazon の権利または所有権と矛盾するどのような措置も講じません。

Amazon は、単独の裁量で行使可能な権利を有し、これらのガイドラインや承認された商標をいつでも変更し、アクセス許可なく、または本ガイドラインに準拠しない使用に対しては適切な措置を講じる権利を留保します。このガイドラインについてご不明な点がございましたら、trademarks@amazon.com までお問い合わせください。または、書面で以下までお問い合わせください。

Amazon.com, Inc., Attention: Trademarks

PO Box 81226

Seattle, WA 98108-1226

AWS Marketplace イベントの販売者通知

AWS Marketplace は、E メール、Amazon EventBridge イベント、Amazon Simple Notification Service (Amazon SNS) トピックを通じてタイムリーな通知を提供します。

トピック

- [AWS Marketplace イベントの E メール通知](#)
- [Amazon EventBridge イベント](#)
- [AWS Marketplace イベントの Amazon Simple Notification Service 通知](#)

AWS Marketplace イベントの E メール通知

AWS Marketplace は、ルートユーザーアカウントを使用して自動 E メールメッセージを送信します。自動 E メールメッセージは、に関連付けられた E メールアドレスに送信されます AWS アカウント。これらの E メールメッセージは、のイベントを可視化 AWS Marketplace し、自動的に送信されます。

Note

通知用のカスタム E メールエイリアスを追加できます。詳細については、「[the section called “E メールアドレスの追加または更新”](#)」を参照してください。

AWS Marketplace は、で行われたオファーと契約のトランザクションを検証する E メール通知を送信します AWS Marketplace。通知は、購入者のサブスクリプションが正常に履行されたことに基づいて、リアルタイムで送信されます。

購入者がオファーを受け入れると、販売者は E メール通知を受け取ります。パブリックサブスクリプションの通知は、購入者と独立系ソフトウェアベンダー (ISV) に送信されます。チャンネルパートナーのプライベートオファーのサブスクリプションでは、購入者、ISV、チャンネルパートナーに送信されます。

ISV またはチャンネルパートナーに送信される E メール通知には、以下の詳細が含まれます。

- 購入日、時刻、およびタイムゾーン
- 顧客 AWS アカウント ID

- 製品名
- 製品の識別子
- オフアー名
- オフアーの識別子
- 契約の識別子
- サービス開始日
- サービス終了日
- 購入金額 (契約およびチャネルパートナー向け)

Note

特定の E メールプロバイダー (Google や Yahoo など) は、AWS Marketplace の通知 E メールを除外する場合があります。から通知を受信していない場合 AWS Marketplace、またはスパムフォルダに表示されている場合は、E メール設定を調整します。例えば、[「Google グループの手順」](#)または[「Yahoo の手順」](#)を参照してください。

以下のトピックでは、E メール通知がサポートするイベントタイプと、通知の管理方法について説明します。

イベントタイプ

以下のイベントタイプは、すべての製品および価格タイプの E メール通知でサポートされています。

- 購入者がプロフェッショナルサービス製品をリクエストした
- 定期スキヤンの脆弱性または定期スキヤンのリマインダー
- 再販機会が作成、更新、または期限切れになった
- 新規または更新されたプライベートオフアーが公開された
- 新規または更新されたチャネルオフアーが公開された
- 購入者と販売者へオフアーの受理を知らせる E メール通知

通知を管理する

以下のトピックでは、イベントの E メール通知を管理する方法について説明します。

E メールアドレスの追加または更新

AWS Marketplace 管理ポータルを使用して、カスタム E メール通知に最大 10 件の E メールアドレスを追加できます。

E メールアドレスを追加または更新するには

1. [AWS Marketplace 管理ポータル](#) にサインインします。
2. [設定] から [通知] タブを選択します。
3. [カスタム通知用 E メール] で [メールアドレスを追加] を選択します。
4. [受信者の詳細] では、[E メールアドレス] フィールドにカスタム E メールアドレスを入力します。
5. (オプション) [新しい受信者を追加] を選択し、別の E メールアドレスを追加します (合計 10 個まで)。
6. [送信] を選択します。

通知から受信者登録を解除する

E メールアドレスを削除すると、受信者はカスタム E メール通知の購読を解除できます。

イベント通知から受信者登録を解除するには

1. [AWS Marketplace 管理ポータル](#) にサインインします。
2. [設定] から [通知] タブを選択します。
3. [カスタム通知用 E メール] で [メールアドレスを更新] を選択します。
4. [受信者の詳細] で [削除] を選択して E メールアドレスを削除します。
5. [送信] を選択します。

受信者はカスタムイベントの E メール通知を受信しなくなります。

Note

E メール内のリンクを使用して登録を解除することもできます。

Amazon EventBridge イベント

AWS Marketplace は EventBridge、以前は Amazon CloudWatch Events と呼ばれていた Amazon と統合されています。EventBridge は、アプリケーションをさまざまなソースのデータに接続するために使用できるイベントバスサービスです。詳細については、[「Amazon ユーザーガイド EventBridge」](#)を参照してください。

販売者は、オファーの作成 AWS Marketplace 時 からイベントを受け取ります。イベントには、ID、有効期限、製品の詳細などの詳細が含まれます。

トピック

- [AWS Marketplace Catalog API Amazon EventBridge イベント](#)

AWS Marketplace Catalog API Amazon EventBridge イベント

このトピックでは、EventBridge コンソールの Marketplace Catalog サービスでのイベントに関する詳細情報を提供します。

販売者によるアクション	受信したイベント	関連トピック
独立系ソフトウェアベンダー (ISV) がオファーを作成し、オファーを購入可能にする	Offer Released	the section called “新しいオファーのイベント”
ISV の製品をチャネルパートナーが使用してオファーを作成する	Offer Released	the section called “新しいオファーのイベント”
チャネルパートナーがオファーを作成する	Offer Released	the section called “新しいオファーのイベント”
変更セットが成功する	Change Set Succeeded	the section called “変更セットのイベント”
変更セットが失敗する	Change Set Failed	the section called “変更セットのイベント”

販売者によるアクション	受信したイベント	関連トピック
変更セットがキャンセルされる	Change Set Cancelled	the section called “変更セットのイベント”
ISV の製品でセキュリティの脆弱性が検出されました	Products Security Report Created	the section called “セキュリティ概要レポートのイベント”

新しいオファアのイベント

販売者がオファアを作成して、オファアを購入できるようにすると、購入者は次の詳細タイプのイベントを受け取ります: Offer Released。

Note

EventBridge ルールの作成については、[「Amazon ユーザーガイド」の「Amazon EventBridge ルール」](#)を参照してください。 EventBridge

ISV が作成した新しいオファアのイベント本文の例を以下に示します。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Offer/offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
      "arn": "arn:aws:catalog:us-east-1:123456789012:Offer/offer-1234567890123",
      "name": "Offer Name",

```

```
    "expirationDate": "2025-08-26T00:00:00Z"
  },
  "product": {
    "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
    "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
    "title": "Product Title"
  },
  "manufacturer": {
    "accountId": "123456789012",
    "name": "Manufacturer Account Name"
  },
  "sellerOfRecord": {
    "accountId": "123456789012",
    "name": "Seller Account Name"
  },
  "targetedBuyerAccountIds": [
    "999988887777",
    "111122223333"
  ]
}
}
```

以下は、ISV の製品をチャネルパートナーが使用してオファーを作成した場合のイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:987654321098:AWSMarketplace/Offer/
offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
```

```
    "arn": "arn:aws:catalog:us-east-1:987654321098:Offer/offer-1234567890123",
    "name": "Offer Name",
    "expirationDate": "2025-08-26T00:00:00Z"
  },
  "product": {
    "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
    "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
    "title": "Product Title"
  },
  "manufacturer": {
    "accountId": "123456789012",
    "name": "Manufacturer Account Name"
  },
  "sellerOfRecord": {
    "accountId": "987654321098",
    "name": "Seller Account Name"
  },
  "targetedBuyerAccountIds": ["999988887777", "111122223333"],
}
}
```

以下は、チャネルパートナーがオファーを作成した場合のイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "987654321098",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:987654321098:AWSMarketplace/Offer/
offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
      "arn": "arn:aws:catalog:us-east-1:987654321098:Offer/offer-1234567890123",

```

```
    "name": "Offer Name",
    "expirationDate": "2025-08-26T00:00:00Z"
  },
  "product": {
    "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
    "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
    "title": "Product Title"
  },
  "manufacturer": {
    "accountId": "123456789012",
    "name": "Manufacturer Account Name"
  },
  "sellerOfRecord": {
    "accountId": "987654321098",
    "name": "Seller Account Name"
  },
  "targetedBuyerAccountIds": ["999988887777", "111122223333"],
}
}
```

変更セットのイベント

変更セットが完了すると、販売者、チャネルパートナー、プライベートマーケットプレイス管理者はイベントを受信することができます。AWS Marketplace Catalog API は、変更セットが完了し、ステータスが成功、失敗、またはキャンセルされたときにイベントを送信します。これらのイベントのソースは `aws.marketplacecatalog` で、詳細タイプの値は `Change Set Succeeded`、`Change Set Failed`、`Change Set Cancelled` のいずれかになります。

Note

変更セットの詳細については、「AWS Marketplace Catalog API リファレンス」の「[変更セットの使用](#)」を参照してください。

各イベントには、変更セット ID、変更セット名、イベント詳細タイプ、失敗コード (失敗したリクエストの場合)、リクエストの開始時刻と終了時刻などの変更リクエストの詳細が含まれます。これにより、`DescribeChangeSet` アクションを継続的にクエリしたり、変更リクエストのステータス `AWS Marketplace 管理ポータル` を確認したりすることなく、変更セットをモニタリングできます。

Note

EventBridge ルールの作成については、[「Amazon ユーザーガイド」の「Amazon EventBridge ルール」](#)を参照してください。 EventBridge

以下は、Change Set Succeeded 詳細タイプのイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Succeeded",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/76yesvf8y165pa4f98td2crtg"
  ],
  "detail": {
    "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "Catalog": "AWSMarketplace",
    "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
    "ChangeSetName": "Create my product",
    "StartTime": "2018-02-27T13:45:22Z",
    "EndTime": "2018-02-27T14:55:22Z"
  }
}
```

以下は、Change Set Failed 詳細タイプのイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Failed",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": [
```

```
"arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/76yesvf8y165pa4f98td2crtg"
],
"detail": {
  "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
  "Catalog": "AWSMarketplace",
  "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
  "ChangeSetName": "Create my product",
  "StartTime": "2018-02-27T13:45:22Z",
  "EndTime": "2018-02-27T14:55:22Z",
  "FailureCode": "CLIENT_ERROR"
}
}
```

以下は、Change Set Cancelled 詳細タイプのイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Cancelled",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/76yesvf8y165pa4f98td2crtg"
  ],
  "detail": {
    "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "Catalog": "AWSMarketplace",
    "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
    "ChangeSetName": "Create my product",
    "StartTime": "2018-02-27T13:45:22Z",
    "EndTime": "2018-02-27T14:55:22Z"
  }
}
```

セキュリティ概要レポートのイベント

販売者の製品でセキュリティの脆弱性が検出されると、概要レポートイベントと、未解決の製品問題の定期的なリマインダーを受け取ることができます。これらのイベントのソースは `aws.marketplacecatalog`、詳細タイプは `Products Security Report Created` です。

各イベントには、検出された問題のある製品とバージョンの数、影響を受ける最新バージョンの数、およびこれらの製品またはバージョンの一時的な制限を防ぐために解決が必要な日付の概要が含まれます。

Note

EventBridge ルールの作成については、[「Amazon ユーザーガイド」の「Amazon EventBridge ルール」](#)を参照してください。EventBridge セキュリティイベントの管理の詳細については、ブログのブログ記事の[「製品カタログのセキュリティを向上させる方法 AWS MarketplaceAWS」](#)を参照してください。

以下は、Products Security Report Created 詳細タイプのイベント本文の例です。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Products Security Report Created",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2023-10-31T00:00:00Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "numberOfProductsWithIssues": 1,
    "numberOfVersionsWithIssues": 1,
    "numberOfLatestVersionsWithIssues": 1,
    "newIssuesFound": true,
    "upcomingResolutionDueDate": "2023-12-01T00:00:00Z",
    "requestId": "533fa17d-3e97-5051-bcaf-1fae45fb3f8b"
  }
}
```

AWS Marketplace イベントの Amazon Simple Notification Service 通知

AWS Marketplace は、以下の製品タイプの購入者のサブスクリプションおよび契約エンタイトルメントの変更について、Amazon Simple Notification Service (Amazon SNS) を通じて通知を送信することもできます。

- [Software as a service \(SaaS\) 製品](#)
- [Amazon マシンイメージ \(AMI\) 製品](#)
- [コンテナ製品](#)

出品者レポート、データフィード、ダッシュボード

AWS Marketplace には、製品の販売に関する情報を収集および分析するための以下のツールが用意されています。

- 自動的に作成され、登録されたすべての AWS Marketplace 販売者が利用できる [レポート](#)。
- [API](#)。これらのレポートの各セクションにアクセスできます。
- [データフィード](#)。レポートに示された取引の顧客情報を識別するための追加の顧客情報を提供します。
- [Amazon QuickSight が提供するダッシュボード](#) には、財務データへのアクセスと分析に役立つグラフ、グラフ、インサイトが含まれています。

AWS Marketplace は、レポート、データフィード、ダッシュボードにできるだけ多くのデータを提供し、以下に従います。

- 顧客データの保護に関する Amazon の標準と規定。
- 購入者が AWS Marketplace で製品を購入する際に承諾する利用規約。販売者は、購入者のデータを安全に管理し、購入者からリクエストがあり次第、データを削除することが契約上義務付けられています。

販売者配送データフィードサービス

AWS Marketplace は、構造化された up-to-date 製品および顧客情報を AWS Marketplace システムから販売者 Amazon S3 バケットに送信し、販売者が所有するビジネスインテリジェンスツール間で ETL (抽出、変換、ロード) するためのメカニズムとしてデータフィードを提供します。

取引データは双時構造で配信および追加されるため、販売者は両方のタイムスタンプ付きの 2 つのタイムラインに沿ってデータを保存し、クエリを実行できます。

- 有効時間: 現実世界で事実が発生した日時 (「知ったこと」)
- システム時間: その事実がデータベースに記録された日時 (「知ったとき」)

データフィードは、前日の 24 時間分のデータを含む前日の更新の後、毎日午後 4 時 (太平洋標準時) (UTC 午前零時) に配信されます。更新は、顧客サブスクライブ、請求、または AWS 支払いによって定義できます。

このセクションでは、データフィードの概要を示し、データフィードにアクセスして使用方法について説明します。以降のセクションでは、各データフィードについて説明します。

データフィードの保存と構造

データフィードは、カンマ区切り値 (CSV) ファイルを収集し、これを指定先の暗号化された Amazon S3 バケットに配信します。CSV ファイルの特性は次のとおりです。

- [4180 標準](#)に準拠しています。
- 文字エンコーディングは UTF-8 (BOM なし) です。
- カンマは、値間の区切り文字として使用されます。
- フィールドは二重引用符でエスケープ
- \n は改行文字です。
- 日付は UTC タイムゾーンで報告され、日時形式は ISO 8601 に従い、精度は 1 秒以内です。
- すべての *_period_start_date 値および *_period_end_date 値は包括的です。つまり、23:59:59 は任意の日の最後のタイムスタンプです。
- すべての金銭情報フィールドの先頭に通貨フィールドが付きます。
- 金銭情報フィールドでは、小数点の区切り文字としてピリオド (.) を使用し、3 桁の区切り文字としてカンマ (,) を使用しません。

データフィードは次のように生成されて保存されます。

- データフィードは 1 日以内に生成され、前日の 24 時間のデータが含まれます。
- Amazon S3 バケットの場合、データフィードは次の形式を使用して月別に整理されます。

bucket-name/data-feed-name_version/year=YYYY/month=MM/data.csv

- 毎日のデータフィードが生成されると、当月の既存の CSV ファイルに追加されます。新しい月が始まると、データフィードごとに新しい CSV ファイルが生成されます。
- データフィードの情報は、2010 年 1 月 1 日から 2020 年 4 月 30 日 (当日を含む) にバックフィルされ、year=2010/month=01 サブフォルダの [CSV ファイル](#)で確認できます。

特定のデータフィードの当月のファイルに列ヘッダーのみが含まれ、データが含まれていない場合があります。これは、当月のフィードに新しいエントリがなかったことを意味します。製品フィードなど、更新頻度が低いデータフィードで発生する場合があります。このような場合、データはバックフィルされたフォルダで利用できます。

- Amazon S3 では、[Amazon S3 ライフサイクルポリシー](#)を作成して、バケットにファイルを保持する期間を管理できます。
- 暗号化された S3 バケットにデータが配信されたときに通知するように Amazon SNS を設定できます。通知を設定する方法の詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[Amazon SNS の開始方法](#)」を参照してください。

データの履歴化

各データフィールドには、データの履歴を示す列があります。valid_to を除き、これらの列はすべてのデータフィールドに共通です。これらは共通の履歴スキーマとして含まれており、データのクエリに役立ちます。

列名	説明
valid_from	主キーの値が他のフィールドの値に関連して有効である最初の日付。
valid_to	この列は 住所 データフィールドにのみ表示され、常に空白です。
insert_date	レコードがデータフィールドに挿入された日付。
update_date	レコードが最後に更新された日付。
delete_date	この列は常に空白です。

これらの列の例を次に示します。

valid_from	valid_to	insert_date	update_date	delete_date
2018-12-12T02:00:00Z		2018-12-12T02:00:00Z	2018-12-12T02:00:00Z	
2019-03-29T03:00:00Z		2019-03-29T03:00:00Z	2019-03-29T03:00:00Z	
2019-03-29T03:00:00Z		2019-03-29T03:00:00Z	2019-04-28T03:00:00Z	

valid_from と update_date フィールドが一緒になって双時データモデルを形成します。valid_from フィールドは、その名のとおり、アイテムがいつ有効になるかがわかります。項目が編集された場合、フィールドにはそれぞれ、update_date が異なり、valid_from の日付が同じレコードが複数含まれている可能性があります。例えば、ある項目の現在の値を調べるには、最新の valid_from の日付があるレコードのリストから最新の update_date のレコードを検索します。

上の例では、レコードは元々 2018 年 12 月 12 日に作成されたものです。その後、2019 年 3 月 29 日に変更されました (レコード内のアドレスが変更された場合など)。その後、2019 年 4 月 28 日に、アドレスの変更が修正されました (valid_from は変更されず、update_date は変更されました)。アドレスを修正すると (まれなイベント)、レコードが元の valid_from の日付から遡って変更されるため、そのフィールドは変更されませんでした。最新の valid_from レコードを検索するクエリでは 2 つのレコードが返され、最新の update_date から現在の実際のレコードが得られます。

データフィードへのアクセス

データフィードにアクセスするには、暗号化された Amazon S3 バケットにデータフィードを受信するように環境を設定する必要があります。は、設定を簡素化するために使用できる [AWS CloudFormation テンプレート](#) AWS Marketplace を提供します。

AWS CloudFormation テンプレートを使用してデータフィードを受信するように環境を設定するには

1. ウェブブラウザを開いて [AWS Marketplace 管理ポータル](#) にサインインし、「[顧客データストレージの設定](#)」に進みます。
2. テンプレートを使用してリソースを作成する AWS CloudFormation を選択して、別のウィンドウの AWS CloudFormation コンソールでテンプレートを開きます。
3. テンプレートで以下を指定し、[Next (次へ)] を選択します。
 - スタック名 - データフィードへのアクセスを有効にするために作成するリソースのコレクション。
 - Amazon S3 バケット名 - データフィードを保存する先のバケット。
 - (オプション) Amazon SNS トピック名 - AWS が新しいデータを Amazon S3 バケットに配信したときに、通知を受信するためのトピック。
4. [Review (確認)] ページで入力内容を確認し、[Create stack (スタックの作成)] を選択します。これにより、CloudFormation ステータスと詳細を含む新しいページが開きます。

- リソース タブから、次のリソースの Amazon リソースネーム (ARNs) ページのフィールドにコピーします。CloudFormation AWS Marketplace <https://aws.amazon.com/marketplace/management/reports/data-feed-configuration>
 - データフィードを保存する先の Amazon S3 バケット
 - AWS KMS Amazon S3 バケットを暗号化するための キー
 - (オプション) が新しいデータを Amazon S3 バケットに配信するときに通知を受信するための Amazon SNS Amazon S3トピック AWS
- [Set up customer data storage (顧客データストレージの設定)] ページで、[Submit (送信)] を選択します。
- (オプション) CloudFormation テンプレートによって作成されたポリシーを編集します。詳細については、「[データフィードポリシー](#)」を参照してください。

これで、データフィードにサブスクライブしました。次回にデータフィードが生成されると、データにアクセスできます。

AWS CloudFormation テンプレートの詳細については、「[ユーザーガイド](#)」の [AWS CloudFormation 「テンプレート」の使用](#) を参照してください。

データフィードポリシー

Amazon S3 バケットが CloudFormation テンプレートによって作成されると、そのバケット、AWS KMS キー、および Amazon SNS トピックにアタッチされたアクセス用のポリシーが作成されます。このポリシーにより、AWS Marketplace レポートサービスはデータフィード情報をバケットと SNS トピックに書き込むことができます。各ポリシーには次のようなセクションがあります (この例は Amazon S3 バケットのものです)。

```
{
  "Sid": "AwsMarketplaceDataFeedsAccess",
  "Effect": "Allow",
  "Principal": {
    "Service": "reports.marketplace.amazonaws.com"
  },
  "Action": [
    "s3:ListBucket",
    "s3:GetObject",
    "s3:PutObject",
    "s3:GetEncryptionConfiguration",
    "s3:GetBucketAcl",
```

```
        "s3:PutObjectAcl"
    ],
    "Resource": [
        "arn:aws:s3:::datafeed-bucket",
        "arn:aws:s3:::datafeed-bucket/*"
    ]
},
```

このポリシーでは、は AWS Marketplace を使用して Amazon S3 バケットにデータをプッシュするサービスプリンシパル `reports.marketplace.amazonaws.com` です。 **datafeed-bucket** は、CloudFormation テンプレートで指定したバケットです。

AWS Marketplace レポートサービスが Amazon S3 AWS KMS または Amazon SNS を呼び出すと、バケットに書き込む予定のデータの ARN が提供されます。バケットに書き込まれるデータが、ユーザーに代わって書き込まれるデータだけにするには、ポリシーの条件に `aws:SourceArn` を指定します。次の例では、*account-id* を自分の AWS アカウントの ID に置き換える必要があります。

```
{
  "Sid": "AwsMarketplaceDataFeedsAccess",
  "Effect": "Allow",
  "Principal": {
    "Service": "reports.marketplace.amazonaws.com"
  },
  "Action": [
    "s3:ListBucket",
    "s3:GetObject",
    "s3:PutObject",
    "s3:GetEncryptionConfiguration",
    "s3:GetBucketAcl",
    "s3:PutObjectAcl"
  ],
  "Resource": [
    "arn:aws:s3:::datafeed-test-bucket",
    "arn:aws:s3:::datafeed-test-bucket/*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:SourceAccount": "account-id",
      "aws:SourceArn": ["arn:aws:marketplace::account-id:AWSMarketplace/SellerDataSubscription/DataFeeds_V1",
```

```
        "arn:aws:marketplace::account-id:AWSMarketplace/  
SellerDataSubscription/Example-Report"]  
    }  
},
```

データフィードからのサブスクリプションの解除

ウェブブラウザを開き、[AWS Marketplace 管理ポータル](#)にサインインします。次に、[お問い合わせページ](#)に移動して、サブスクリプション解除リクエストを AWS Marketplace Seller Operations チームに送信します。受信解除のリクエストの処理が完了するのに、最大で 10 営業日かかる場合があります。

データフィードの使用

Amazon S3 バケットのデータが使用可能な場合、次の方法でデータフィードを使用できます。

- [データフィードへのアクセス](#) で作成した Amazon S3 バケットから .CSV ファイルをダウンロードして、スプレッドシートでデータを表示できるようにします。
- ETL (抽出、変換、ロード)、SQL クエリ、ビジネス分析ツールを使用して、データを収集して分析します。

AWS サービスを使用して、データを収集および分析したり、.CSV ベースのデータセットの分析を実行できるサードパーティーツールを使用したりできます。

例: AWS サービスを使用したデータの収集と分析

次の手順では、データフィードの送信先を Amazon S3 バケットとするように環境を設定済みであり、バケットにデータフィードが含まれていることを前提としています。

データフィードからデータを収集して分析するには

1. [AWS Glue コンソール](#)から、データフィードを格納する Amazon S3 バケットに接続する[クローラーを作成](#)し、必要なデータを抽出して、AWS Glue Data Catalogにメタデータテーブルを作成します。

の詳細については AWS Glue、「[AWS Glue デベロッパーガイド](#)」を参照してください。

2. [Athena コンソール](#)から、[AWS Glue Data Catalogのデータに対して SQL クエリを実行](#)します。

Athena の詳細については、「[Amazon Athena ユーザーガイド](#)」を参照してください。

3. [Amazon QuickSight コンソール](#) から [分析](#) を作成し、データの [ビジュアル](#) を作成します。

Amazon の詳細については QuickSight、「Amazon [ユーザーガイド QuickSight](#)」を参照してください。

AWS サービスを使用してデータフィード内のデータを収集および分析する方法の詳細な例については、AWS Marketplace ブログの「[Using Seller Data Feed Delivery Service, Amazon Athena, and Amazon QuickSight to create seller reports](#)」を参照してください。

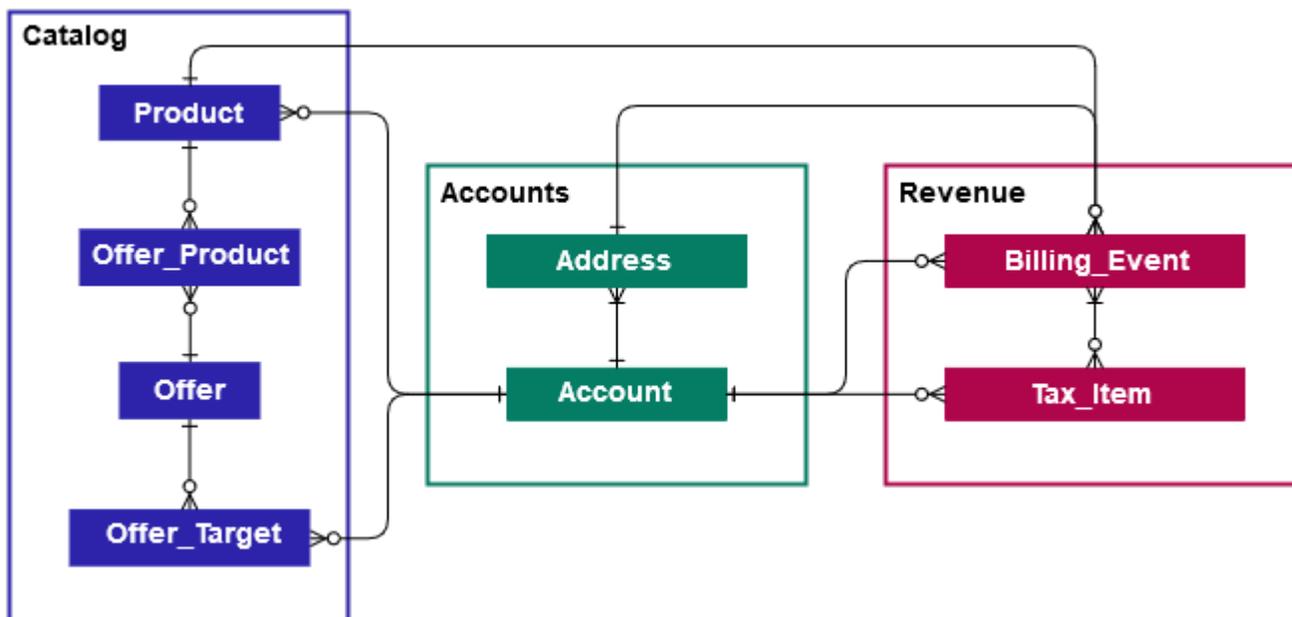
データフィードテーブルの概要

AWS Marketplace 提供されるデータフィードは、結合してクエリのコンテキストを増やすことができる一連のテーブルです。

データフィードには、次の 3 つの一般的なドメイン、つまり関心のあるカテゴリがあります。

- カタログ - アカウント内の製品やオファーに関する情報が含まれます。
- アカウント - で製品を提供または購入するアカウントに関する情報 AWS Marketplace が含まれます (チャンネルパートナーや購入者など、お客様自身のアカウントまたはお客様が協力する相手のアカウント)。
- 収益 - 請求、支払い、税金に関する情報が含まれます。

次の図は、各ドメインのテーブルと、それぞれのテーブル間の関係を示しています。この図は、カタログ、アカウント、収益の各ドメインとその中のテーブルを示しています。



以下のセクションでは、各ドメインのエンティティ関係 (ER) 図を示します。各 ER 図には、テーブルと各テーブル内のフィールド、およびテーブルの結合に使用できるフィールドが示されています。

Note

このセクションの ER 図には、すべてのデータフィールドに共通するフィールドは含まれていません。共通フィールドの詳細については、「[データフィールドの保存と構造](#)」を参照してください。

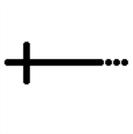
次の表では、ER 図に示されている記号について説明します。

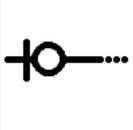
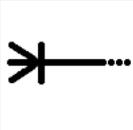
記号	説明
PK	プライマリキー - テーブルのプライマリキー。valid_from および update_date フィールドと一緒に使用すると、一意になります。これらのフィールドと一緒に使用する方法の詳細については、「 データの履歴化 」を参照してください。複数のフィールドがプライマリキーとしてマークされている場合、それらの

記号	説明
	フィールドが一緒になってプライマリキーになります。
	<p>外部キー - 別のテーブルのプライマリキーを表すフィールド。テーブル内で必ずしも一意である必要はありません。</p> <div data-bbox="829 512 1507 827" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"> <p>Note</p> <p>現在のテーブルのレコードに対応するレコードが外部テーブルにない場合、外部キーが空白になることがあります。</p> </div>
	<p>代替キー - テーブル内のキーとして使用できるキー。プライマリキーと同じ一意性のルールに従います。</p>
	<p>コネクタ - フィールド間の線は接続を表し、テーブルを結合するために使用できる 2 つのフィールドです。線の端は接続の種類を表します。この例では、one-to-many 接続を表します。</p>

コネクタタイプ

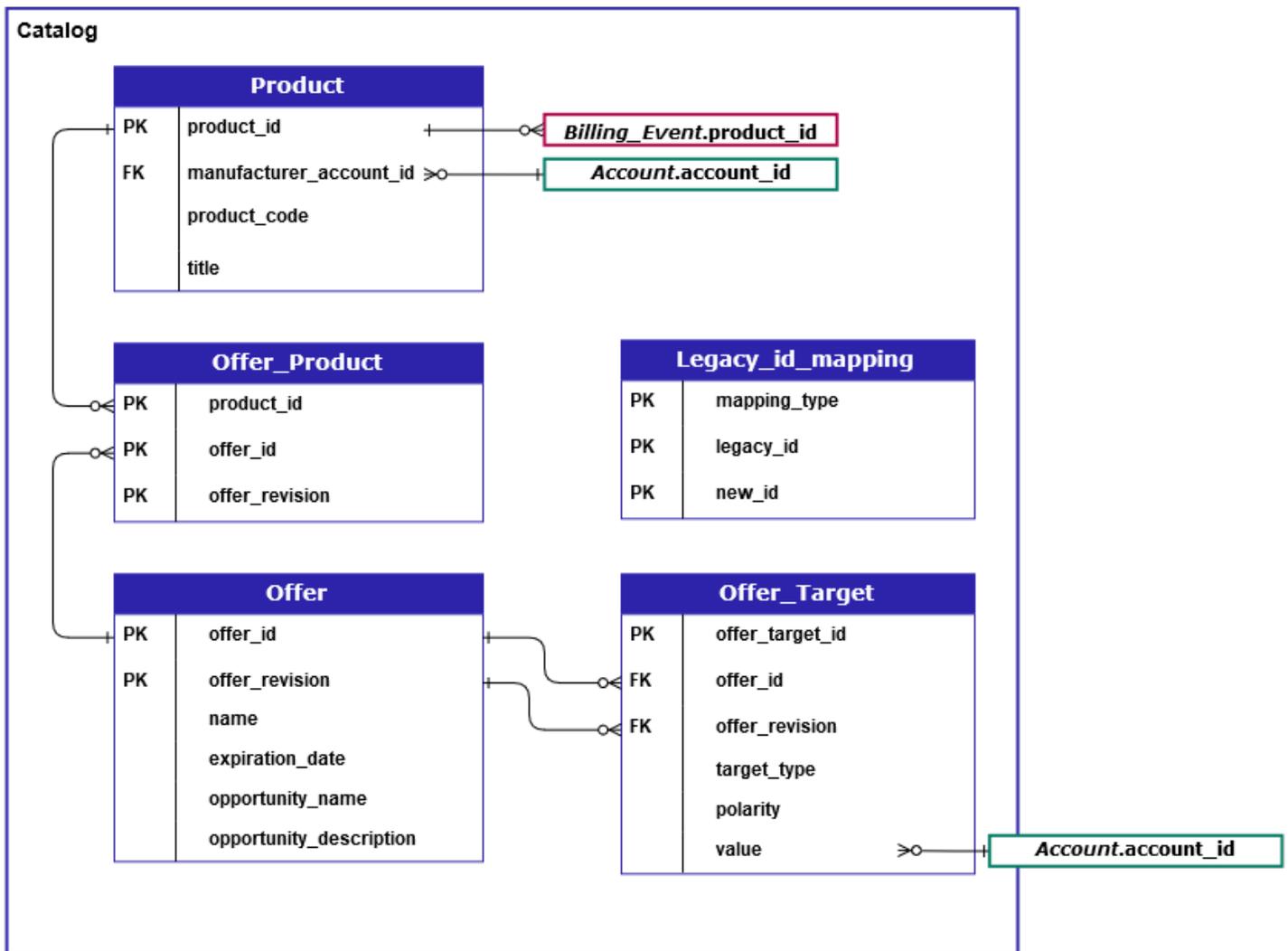
次のテーブルは、各コネクタが持つことができる端部のタイプを示しています。

コネクタタイプ	説明
	<p>1 対 n - この端があるコネクタは、結合のこちら側に値が 1 つだけある結合を表します。</p>

コネクタタイプ	説明
	<p>0 または 1 対 n - この端のコネクタは、結合のこちら側に値が 0 または 1 つある結合を表します。</p>
	<p>0 以上対 n - この端のコネクタは、結合のこちら側に 0 個、1 個、または多数の値を持つ結合を表します。</p>
	<p>1 以上対 n - この端があるコネクタは、結合のこちら側に 1 つまたは複数の値がある結合を表します。</p>

カタログ関連テーブル

次の図は、カタログドメイン内のテーブルとテーブル内のフィールド間の関係を示しています。



Product、Offer_Product、Offer、Offer_Target、Legacy_id_mapping_テーブルはカタログドメインにあります。

Offer_Target テーブルにはターゲットの account_id の値フィールドが含まれますが、target_type 値が account である場合に限りです。

Legacy_id_mapping テーブルは現在のデータには使用されません。

Note

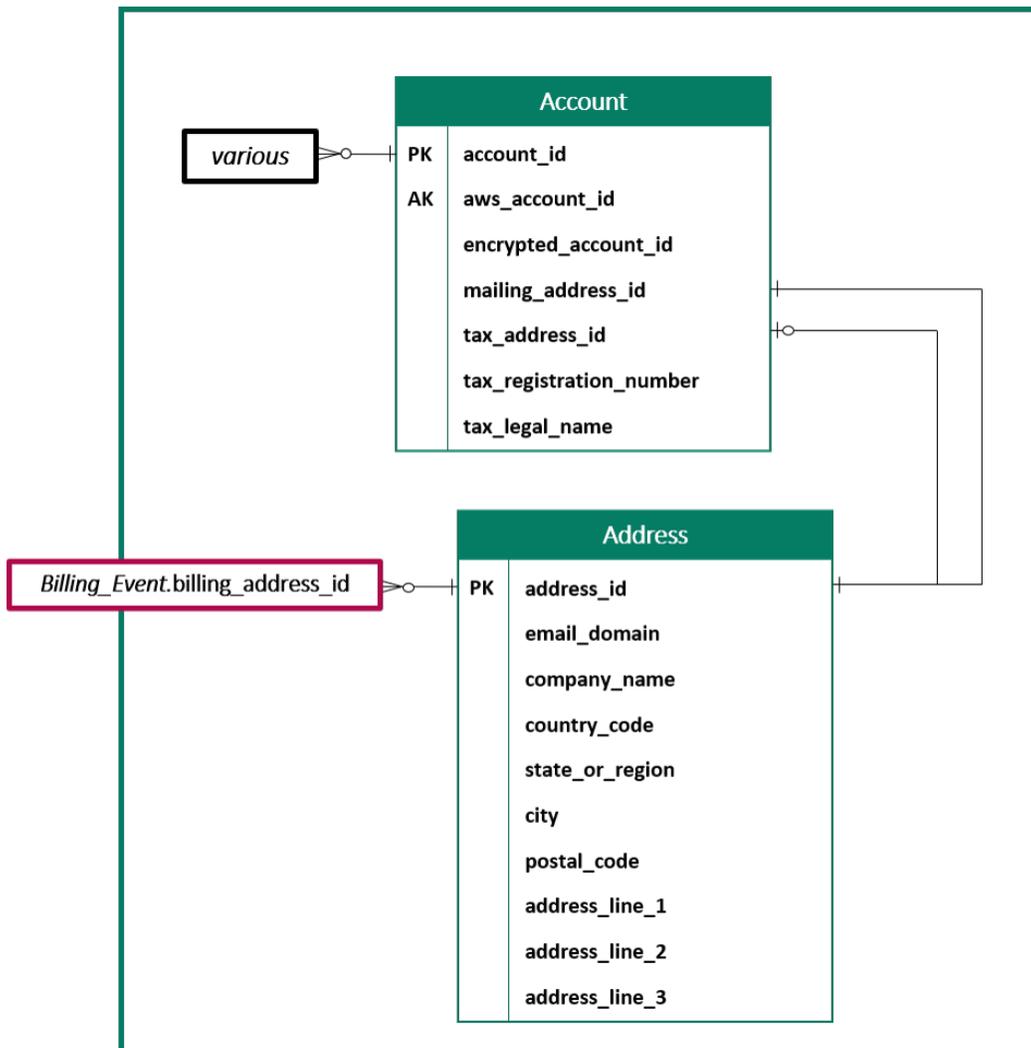
テーブル内の各フィールドの説明や作成できる結合など、これらのテーブルの詳細については、以下のトピックを参照してください。

- [製品データフィード](#)
- [オファー製品データフィード](#)

- [オファーデータフィード](#)
- [オファーターゲットデータフィード](#)
- [レガシーマッピングデータフィード](#)

勘定科目関連テーブル

次の図は、アカウントドメイン内のとのテーブル間の関係、Account および Address テーブル内のフィールドを示しています。



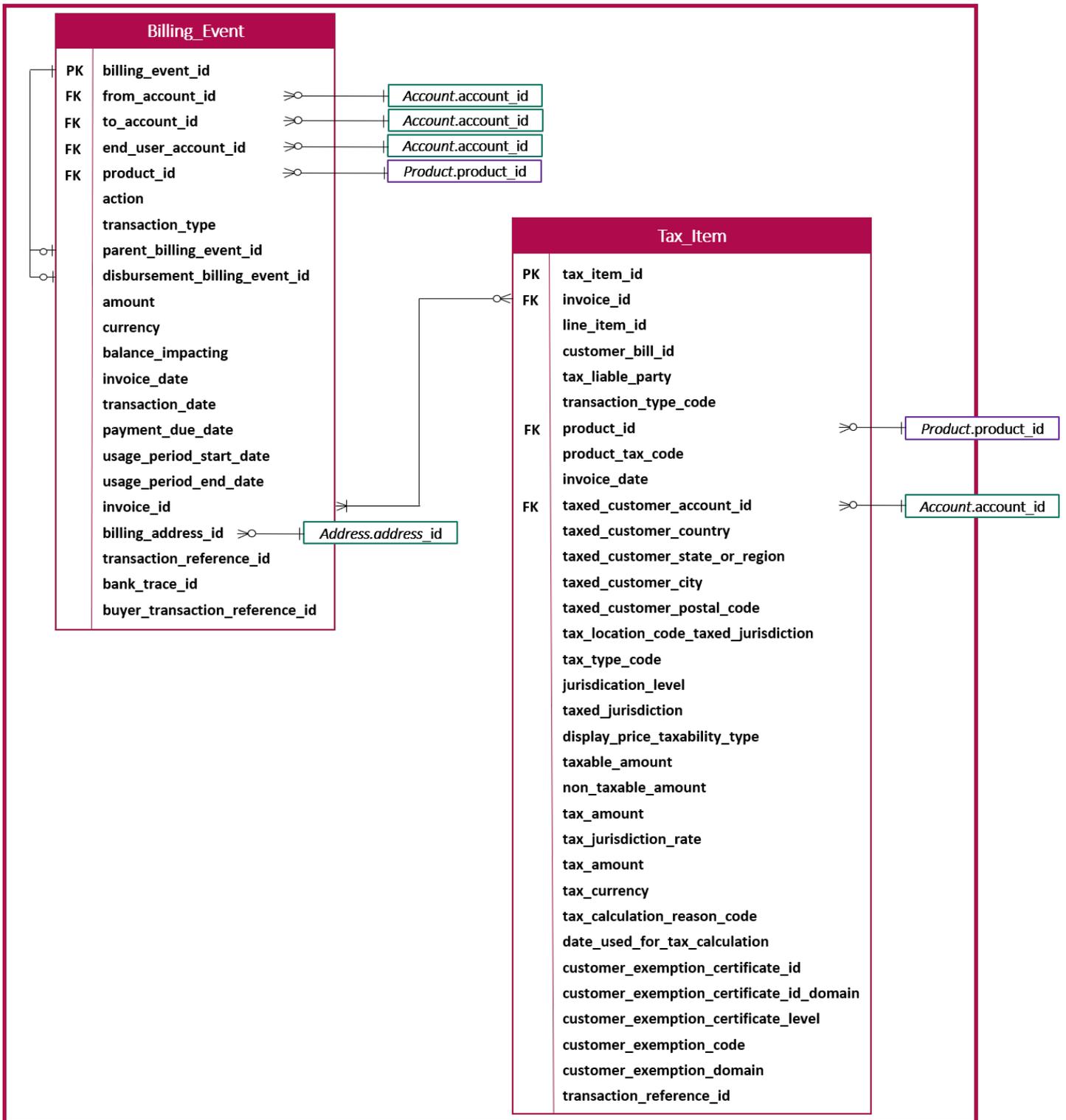
Note

テーブル内の各フィールドの説明や作成できる結合など、これらのテーブルの詳細については、以下のトピックを参照してください。

- [アカウントデータフィールド](#)
- [住所データフィールド](#)

収益関連テーブル

次の図は、収益ドメイン内のとのテーブル間の関係、Billing_Event および Tax_Item テーブル内のフィールドを示しています。Billing_Event テーブルには、支払いと請求イベントに関する情報が含まれています。



Note

テーブル内の各フィールドの説明や作成できる結合など、これらのテーブルの詳細については、以下のトピックを参照してください。

- [請求イベントデータフィード](#)
- [税品目データフィード](#)

データフィードクエリの例

このセクションでは、が提供するデータフィードを使用した複雑なクエリの例を示します AWS Marketplace。これらの例は、AWS Marketplace 管理ポータルから得られる [販売者レポート](#) と似ています。これらのクエリをカスタマイズして、必要な他のレポートを作成できます。

例 1: 製品別の支払い

製品ごとに支払われた金額を確認するには、次のようなクエリを実行できます。この例は、販売者レポートとして得られる [支払いレポート](#) と同等です。ただし、このサンプルを使用して独自のクエリを作成し、必要なレポートを正確に取得するようにカスタマイズできます。

このサンプルクエリのセットは相互に積み上げられて、支払いを含む製品詳細の最終リストを作成します。特定の時点での製品情報を取得する方法も示します。クエリ内のコメントには、クエリの実行内容や、データをさまざまな表示にするためにクエリを変更する方法が説明されています。

Note

このクエリを実行するときは、取り込まれたデータが 2 つの時間軸 (valid_from 列と update 列) を使用していることを前提としています。詳細については、「[データフィードの保存と構造](#)」を参照してください。

```
-- Get all the products and keep the latest product_id, valid_from tuple
with products_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
```

```
        ROW_NUMBER() OVER (PARTITION BY product_id, valid_from
                           ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
    from
        productfeed_v1
)
where
    -- A product_id can appear multiple times with the same
    -- valid_from date but with a different update_date column,
    -- making it effectively bi-temporal. By only taking the most
    -- recent tuple, we are converting to a uni-temporal model.
    row_num = 1
),

-- Gets the latest revision of a product
-- A product can have multiple revisions where some of the
-- columns, like the title, can change.
-- For the purpose of the disbursement report, we want
-- to get the latest revision of a product
products_with_latest_version as (
    select
        *
    from
        (
            select
                *,
                ROW_NUMBER() OVER (PARTITION BY product_id
                                   ORDER BY from_iso8601_timestamp(valid_from) desc)
                as row_num_latest_version
            from
                products_with_uni_temporal_data
        )
    where
        row_num_latest_version = 1
),

-- Get all the accounts and keep the latest account_id, valid_from tuple
accounts_with_uni_temporal_data as (
    select
        *
    from
        (
            select
                *,
```

```
        ROW_NUMBER() OVER (PARTITION BY account_id, valid_from ORDER BY
from_iso8601_timestamp(update_date) desc) as row_num
    from
        accountfeed_v1
    )
    where
        -- An account_id can appear multiple times with the same
        -- valid_from date but with a different update_date column,
        -- making it effectively bi-temporal. By only taking the most
        -- recent tuple, we are converting to a uni-temporal model.
        row_num = 1
    ),

-- Gets the latest revision of an account
-- An account can have multiple revisions where some of the
-- columns, like the mailing_address_id, can change.
-- For the purpose of the disbursement report, we want
-- to get the latest revision of a product
accounts_with_latest_version as (
    select
        *
    from
        (
            select
                *,
                ROW_NUMBER() OVER (PARTITION BY account_id
                ORDER BY from_iso8601_timestamp(valid_from) desc)
                as row_num_latest_version
            from
                accounts_with_uni_temporal_data
        )
    where
        row_num_latest_version = 1
    ),

-- Get all the billing events and keep the
-- latest billing_event_id, valid_from tuple:
billing_events_with_uni_temporal_data as (
    select
        *
    from (
        select
            billing_event_id,
            from_iso8601_timestamp(valid_from) as valid_from,
```

```
    from_iso8601_timestamp(update_date) as update_date,
    from_iso8601_timestamp(invoice_date) as invoice_date,
    transaction_type,
    transaction_reference_id,
    product_id,
    disbursement_billing_event_id,
    action,
    from_account_id,
    to_account_id,
    end_user_account_id,
    CAST(amount as decimal(20, 10)) invoice_amount,
    bank_trace_id,
    ROW_NUMBER() OVER (PARTITION BY billing_event_id, valid_from
        ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
from
    billingeventfeed_v1
)
where row_num = 1
),

-- Get all the disbursements
-- The billing events data is immutable.
-- It is not required to use time windows based on the
-- valid_from column to get the most recent billing event
disbursement_events as (
    select
        billing_events_raw.billing_event_id as disbursement_id,
        billing_events_raw.invoice_date as disbursement_date,
        billing_events_raw.bank_trace_id
    from
        billing_events_with_uni_temporal_data billing_events_raw
    where
        -- Only interested in disbursements, so filter out
        -- non-disbursements by selecting transaction type
        -- to be DISBURSEMENT:
        billing_events_raw.transaction_type = 'DISBURSEMENT'
        -- Select a time period, you can adjust the dates
        -- below if need be. For billing events use the
        -- invoice date as the point in time of the
        -- disbursement being initiated:
        and billing_events_raw.invoice_date >=
            from_iso8601_timestamp('2020-10-01T00:00:00Z')
        and billing_events_raw.invoice_date <
```

```
        from_iso8601_timestamp('2020-11-01T00:00:00Z')
    ),

-- Get the invoices along with the line items that
-- are part of the above filtered disbursements
disbursed_line_items as (
    select
        line_items.transaction_reference_id,
        line_items.product_id,
        line_items.transaction_type,
        (case
            -- Get the payer of the invoice from any
            -- transaction type that is not AWS and
            -- not BALANCE_ADJUSTMENT.
            -- For AWS and BALANCE_ADJUSTMENT, the billing
            -- event feed will show the "AWS Marketplace"
            -- account as the receiver of the funds and the
            -- seller as the payer. Filter those out.
            when line_items.transaction_type
                not like '%AWS%' and transaction_type
                not like 'BALANCE_ADJUSTMENT'
            then line_items.from_account_id
        end) as payer_account_id,
        line_items.end_user_account_id,
        invoice_amount,
        disbursements.disbursement_date,
        disbursements.disbursement_id,
        disbursements.bank_trace_id
    from
        billing_events_with_uni_temporal_data line_items
        -- Each disbursed line item is linked to the parent
        -- disbursement via the disbursement_billing_event_id
    join disbursement_events disbursements
        on disbursements.disbursement_id
        = line_items.disbursement_billing_event_id
    where
        -- we are interested only in the invoice line
        -- items that are DISBURSED
        line_items.action = 'DISBURSED'
),

-- An invoice can contain multiple line items
-- Create a pivot table to calculate the different
-- amounts that are part of an invoice.
```

```
-- The new row is aggregated at
-- transaction_reference_id - end_user_account_id level
invoice_amounts_aggregated as (
  select
    transaction_reference_id,
    product_id,
    -- a given disbursement id should have the
    -- same disbursement_date
    max(disbursement_date) as disbursement_date,
    -- Build a pivot table in order to provide all the
    -- data related to a transaction in a single row.
    -- Note that the amounts are negated. This is because
    -- when an invoice is generated, we give you the
    -- positive amounts and the disbursement event
    -- negates the amounts
    sum(case when transaction_type = 'SELLER_REV_SHARE'
      then -invoice_amount else 0 end) as seller_rev_share,
    sum(case when transaction_type = 'AWS_REV_SHARE'
      then -invoice_amount else 0 end) as aws_rev_share,
    sum(case when transaction_type = 'SELLER_REV_SHARE_REFUND'
      then -invoice_amount else 0 end) as seller_rev_refund,
    sum(case when transaction_type = 'AWS_REV_SHARE_REFUND'
      then -invoice_amount else 0 end) as aws_rev_refund,
    sum(case when transaction_type = 'SELLER_REV_SHARE_CREDIT'
      then -invoice_amount else 0 end) as seller_rev_credit,
    sum(case when transaction_type = 'AWS_REV_SHARE_CREDIT'
      then -invoice_amount else 0 end) as aws_rev_credit,
    sum(case when transaction_type = 'SELLER_TAX_SHARE'
      then -invoice_amount else 0 end) as seller_tax_share,
    sum(case when transaction_type = 'SELLER_TAX_SHARE_REFUND'
      then -invoice_amount else 0 end) as seller_tax_refund,
    -- This is the account that pays the invoice:
    max(payer_account_id) as payer_account_id,
    -- This is the account that subscribed to the product:
    end_user_account_id as customer_account_id,
    bank_trace_id
  from
    disbursed_line_items
  group by
    transaction_reference_id,
    product_id,
    disbursement_id,
    -- There might be a different end-user for the same
    -- transaction reference id. Distributed licenses
```

```
-- is an example
end_user_account_id,
bank_trace_id
),

disbursed_amount_by_product as (
  select
    products.title as ProductTitle,
    products.product_code as ProductCode,
    -- We are rounding the sums using 2 decimal precision
    -- Note that the rounding method might differ
    -- between SQL implementations.
    -- The disbursement seller report is using
    -- RoundingMode.HALF_UP. This might create
    -- discrepancies between this SQL output
    -- and the disbursement seller report
    round(invoice_amounts.seller_rev_share, 2) as SellerRev,
    round(invoice_amounts.aws_rev_share, 2) as AWSRefFee,
    round(invoice_amounts.seller_rev_refund, 2) as SellerRevRefund,
    round(invoice_amounts.aws_rev_refund, 2) as AWSRefFeeRefund,
    round(invoice_amounts.seller_rev_credit, 2) as SellerRevCredit,
    round(invoice_amounts.aws_rev_credit, 2) as AWSRefFeeCredit,
    (
      round(invoice_amounts.seller_rev_share, 2) +
      round(invoice_amounts.aws_rev_share, 2) +
      round(invoice_amounts.seller_rev_refund, 2) +
      round(invoice_amounts.aws_rev_refund, 2) +
      round(invoice_amounts.seller_rev_credit, 2) +
      round(invoice_amounts.aws_rev_credit, 2)
    ) as NetAmount,
    invoice_amounts.transaction_reference_id
      as TransactionReferenceID,
    round(invoice_amounts.seller_tax_share, 2)
      as SellerSalesTax,
    round(invoice_amounts.seller_tax_refund, 2)
      as SellerSalesTaxRefund,
    payer_info.aws_account_id
      as PayerAwsAccountId,
    customer_info.aws_account_id
      as EndCustomerAwsAccountId,
    invoice_amounts.disbursement_date
      as DisbursementDate,
    invoice_amounts.bank_trace_id
      as BankTraceId
```

```
from
  invoice_amounts_aggregated invoice_amounts
  join products_with_latest_version products
    on products.product_id = invoice_amounts.product_id
  left join accounts_with_latest_version payer_info
    on payer_info.account_id = invoice_amounts.payer_account_id
  left join accounts_with_latest_version customer_info
    on customer_info.account_id = invoice_amounts.customer_account_id
)

select * from disbursed_amount_by_product;
```

例 2: 販売補償レポート

顧客別の請求済み収益を確認するには、次のようなクエリを実行します。この例は、販売者レポートとして得られる [販売補償レポート](#) と同等です。ただし、このサンプルを使用して独自のクエリを作成し、必要なレポートを正確に取得するようにカスタマイズできます。

これはサンプルクエリのセットです。これらのクエリを相互に組み合わせて、ソフトウェアの使用に対して各顧客に請求される合計金額を含む顧客詳細の最終リストを作成します。クエリ内のコメントには、クエリの実行内容や、データをさまざまな表示にするためにクエリを変更する方法が説明されています。

Note

このクエリを実行するときは、取り込まれたデータが 2 つの時間軸 (valid_from 列と update 列) を使用していることを前提としています。詳細については、「[データフィールドの保存と構造](#)」を参照してください。

```
-- Gets all the products and keeps the latest product_id,
-- valid_from tuple.
with products_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY product_id, valid_from
                           ORDER BY from_iso8601_timestamp(update_date) desc)
```

```
        as row_num
    from
        productfeed_v1
    )
where
    -- A product_id can appear multiple times with the same
    -- valid_from date but with a different update_date column,
    -- making it effectively bi-temporal. By only taking the most
    -- recent tuple, we are converting to a uni-temporal model.
    row_num = 1
),

-- Gets the latest revision of a product
-- A product can have multiple revisions where some of the
-- columns, like the title, can change.
-- For the purpose of the sales compensation report, we want
-- to get the latest revision of a product
products_with_latest_revision as (
    select
        *
    from
        (
            select
                *,
                ROW_NUMBER() OVER (PARTITION BY product_id ORDER BY
from_iso8601_timestamp(valid_from) desc) as row_num_latest_revision
            from
                products_with_uni_temporal_data
        )
    where
        row_num_latest_revision = 1
),

-- Gets all the addresses and keeps the latest address_id,
-- aws_account_id, and valid_from combination.
-- We're transitioning from a bi-temporal data model to an
-- uni-temporal data_model
piifeed_with_uni_temporal_data as (
    select
        *
    from
        (
            select
                *,
```

```
        ROW_NUMBER() OVER (
            PARTITION BY address_id, aws_account_id, valid_from
            ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
    from
        piifeed
)
where
    -- An address_id can appear multiple times with the same
    -- valid_from date but with a different update_date column.
    -- We are only interested in the most recent.
    row_num = 1
),

-- Gets the latest revision of an address.
-- An address_id can have multiple revisions where some of
-- the columns can change.
-- For the purpose of the sales compensation report, we want to
-- get the latest revision of an address + account_id pair.
pii_with_latest_revision as (
    select
        *
    from
        (
            select
                *,
                ROW_NUMBER() OVER (PARTITION BY address_id, aws_account_id
                    ORDER BY from_iso8601_timestamp(valid_from) desc)
                    as row_num_latest_revision
            from
                piifeed_with_uni_temporal_data
        )
    where
        row_num_latest_revision = 1
),

-- Gets all the accounts and keeps the latest
-- account_id, valid_from tuple.
-- We're transitioning from a bi-temporal data
-- model to an uni-temporal data_model.
accounts_with_uni_temporal_data as (
    select
        *
    from
```

```
(
  select
    *,
    ROW_NUMBER() OVER (PARTITION BY account_id, valid_from
      ORDER BY from_iso8601_timestamp(update_date) desc)
      as row_num
  from
    accountfeed_v1
)
where
  -- An account_id can appear multiple times with the same
  -- valid_from date but with a different update_date column.
  -- We are only interested in the most recent tuple.
  row_num = 1
),

-- Gets all the historical dates for an account
-- An account can have multiple revisions where some of the
-- columns like the mailing_address_id can change.
accounts_with_history as (
  select
    *,
    -- This interval's begin_date
    case
      when
        -- First record for a given account_id
        lag(valid_from, 1) over (partition by account_id
          order by from_iso8601_timestamp(valid_from) asc) is null
      then
        -- 'force' begin_date a bit earlier because of different
        -- data propagation times. We'll subtract one day as one
        -- hour is not sufficient
        from_iso8601_timestamp(valid_from) - INTERVAL '1' DAY
      else
        -- not the first line -> return the real date
        from_iso8601_timestamp(valid_from)
    end as begin_date,
    -- This interval's end date.
    COALESCE(
      LEAD(from_iso8601_timestamp(valid_from), 1)
        OVER (partition by account_id
          ORDER BY from_iso8601_timestamp(valid_from)),
      from_iso8601_timestamp('9999-01-01T00:00:00Z')
    ) as end_date
```

```
from
  accounts_with_uni_temporal_data
),

-- Gets all the billing events and keeps the latest
-- billing_event_id, valid_from tuple.
-- We're transitioning from a bi-temporal data
-- model to an uni-temporal data_model.
billing_events_with_uni_temporal_data as (
  select
    *
  from (
    select
      billing_event_id,
      from_iso8601_timestamp(valid_from) as valid_from,
      from_iso8601_timestamp(update_date) as update_date,
      from_iso8601_timestamp(invoice_date) as invoice_date,
      transaction_type,
      transaction_reference_id,
      product_id,
      disbursement_billing_event_id,
      action,
      currency,
      from_account_id,
      to_account_id,
      end_user_account_id,
      -- convert an empty billing address to null. This will
      -- later be used in a COALESCE call
      case
        when billing_address_id <> '' then billing_address_id else null
      end as billing_address_id,
      CAST(amount as decimal(20, 10)) invoice_amount,
      ROW_NUMBER() OVER (PARTITION BY billing_event_id, valid_from
        ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
    from
      billingeventfeed_v1
  where
    -- The Sales Compensation Report does not contain BALANCE
    -- ADJUSTMENTS, so we filter them out here
    transaction_type <> 'BALANCE_ADJUSTMENT'
    -- Keep only the transactions that will affect any
    -- future disbursed amounts.
    and balance_impacting = '1'
```

```
    )
  where row_num = 1
),

-- Gets the billing address for all DISBURSED invoices. This
-- will be the address of the payer when the invoice was paid.
-- NOTE: For legal reasons, for CPP0 transactions, the
-- manufacturer will not see the payer's billing address id
billing_addresses_for_disbursed_invoices as (
  select
    billing_events_raw.transaction_reference_id,
    billing_events_raw.billing_address_id,
    billing_events_raw.from_account_id
  from
    billing_events_with_uni_temporal_data billing_events_raw
  where
    -- the disbursed items will contain the billing address id
    billing_events_raw.action = 'DISBURSED'
    -- we only want to get the billing address id for the
    -- transaction line items where the seller is the receiver
    -- of the amount
    and billing_events_raw.transaction_type like 'SELLER_%'
  group by
    billing_events_raw.transaction_reference_id,
    billing_events_raw.billing_address_id,
    billing_events_raw.from_account_id
),

-- An invoice can contain multiple line items.
-- We create a pivot table to calculate the different amounts
-- that are part of an invoice.
-- The new row is aggregated at
-- transaction_reference_id - end_user_account_id level
invoiced_and_forgiven_transactions as (
  select
    transaction_reference_id,
    product_id,
    -- A transaction will have the same invoice date for all
    -- of its line items (transaction types)
    max(invoice_date) as invoice_date,
    -- A transaction will have the same billing_address_id
    -- for all of its line items. Remember that the billing event
    -- is uni temporal and we retrieved only the latest valid_from item
    max(billing_address_id) as billing_address_id,
```

```
-- A transaction will have the same currency for all
-- of its line items
max(currency) as currency,
-- We're building a pivot table in order to provide all the
-- data related to a transaction in a single row
sum(case when transaction_type = 'SELLER_REV_SHARE'
      then invoice_amount else 0 end) as seller_rev_share,
sum(case when transaction_type = 'AWS_REV_SHARE'
      then invoice_amount else 0 end) as aws_rev_share,
sum(case when transaction_type = 'SELLER_REV_SHARE_REFUND'
      then invoice_amount else 0 end) as seller_rev_refund,
sum(case when transaction_type = 'AWS_REV_SHARE_REFUND'
      then invoice_amount else 0 end) as aws_rev_refund,
sum(case when transaction_type = 'SELLER_REV_SHARE_CREDIT'
      then invoice_amount else 0 end) as seller_rev_credit,
sum(case when transaction_type = 'AWS_REV_SHARE_CREDIT'
      then invoice_amount else 0 end) as aws_rev_credit,
sum(case when transaction_type = 'SELLER_TAX_SHARE'
      then invoice_amount else 0 end) as seller_tax_share,
sum(case when transaction_type = 'SELLER_TAX_SHARE_REFUND'
      then invoice_amount else 0 end) as seller_tax_refund,
-- this is the account that pays the invoice.
max(case
  -- Get the payer of the invoice from any transaction type
  -- that is not AWS and not BALANCE_ADJUSTMENT.
  -- For AWS and BALANCE_ADJUSTMENT, the billing event feed
  -- will show the "AWS Marketplace" account as the
  -- receiver of the funds and the seller as the payer. We
  -- are not interested in this information here.
  when
    transaction_type not like '%AWS%'
    and transaction_type not like 'BALANCE_ADJUSTMENT'
  then from_account_id
  end) as payer_account_id,
-- this is the account that subscribed to your product
end_user_account_id as customer_account_id
from
  billing_events_with_uni_temporal_data
where
  -- Get invoiced or forgiven items. Disbursements are
  -- not part of the sales compensation report
  action in ('INVOICED', 'FORGIVEN')
group by
  transaction_reference_id,
```

```

    product_id,
    -- There might be a different end-user for the same
    -- transaction reference id. Distributed licenses
    -- is an example.
    end_user_account_id
),

invoiced_items_with_product_and_billing_address as (
  select
    invoice_amounts.*,
    products.product_code,
    products.title,
    payer_info.aws_account_id as payer_aws_account_id,
    payer_info.account_id as payer_reference_id,
    customer_info.aws_account_id as end_user_aws_account_id,
    (
      invoice_amounts.seller_rev_share +
      invoice_amounts.aws_rev_share +
      invoice_amounts.seller_rev_refund +
      invoice_amounts.aws_rev_refund +
      invoice_amounts.seller_rev_credit +
      invoice_amounts.aws_rev_credit +
      invoice_amounts.seller_tax_share +
      invoice_amounts.seller_tax_refund
    ) as seller_net_revenue,
    -- Try to get the billing address from the DISBURSED event
    -- (if any). If there is no DISBURSEMENT, get the billing
    -- address from the INVOICED item. If still no billing address,
    -- then default to getting the mailing address of the payer.
    coalesce(billing_add.billing_address_id,
             invoice_amounts.billing_address_id,
             payer_info.mailing_address_id)
             as final_billing_address_id
  from
    invoiced_and_forgiven_transactions invoice_amounts
  join products_with_latest_revision products
    on products.product_id = invoice_amounts.product_id
  left join accounts_with_history payer_info
    on payer_info.account_id = invoice_amounts.payer_account_id
    -- Get the Payer Information at the time of invoice creation
    and payer_info.begin_date <= invoice_amounts.invoice_date
    and invoice_amounts.invoice_date < payer_info.end_date
  left join accounts_with_history customer_info
    on customer_info.account_id = invoice_amounts.customer_account_id

```

```
-- Get the End User Information at the time of invoice creation
and customer_info.begin_date <= invoice_amounts.invoice_date
and invoice_amounts.invoice_date < customer_info.end_date
left join billing_addresses_for_disbursed_invoices billing_add
on billing_add.transaction_reference_id =
    invoice_amounts.transaction_reference_id
and billing_add.from_account_id =
    invoice_amounts.payer_account_id
),

invoices_with_full_address as (
  select
    payer_aws_account_id as "Customer AWS Account Number",
    pii_data.country as "Country",
    pii_data.state_or_region as "State",
    pii_data.city as "City",
    pii_data.postal_code as "Zip Code",
    pii_data.email_domain as "Email Domain",
    product_code as "Product Code",
    title as "Product Title",
    seller_rev_share as "Gross Revenue",
    aws_rev_share as "AWS Revenue Share",
    seller_rev_refund as "Gross Refunds",
    aws_rev_refund as "AWS Refunds Share",
    seller_net_revenue as "Net Revenue",
    currency as "Currency",
    date_format(invoice_date, '%Y-%m') as "AR Period",
    transaction_reference_id as "Transaction Reference ID",
    payer_reference_id as "Payer Reference ID",
    end_user_aws_account_id as "End Customer AWS Account ID"
  from
    invoiced_items_with_product_and_billing_address invoice_amounts
  left join pii_with_latest_revision pii_data
    on pii_data.aws_account_id = invoice_amounts.payer_aws_account_id
    and pii_data.address_id = invoice_amounts.final_billing_address_id
  -- Filter out FORGIVEN and Field Demonstration Pricing transactions
  where seller_net_revenue <> 0
)

select * from invoices_with_full_address;
```

データフィード

AWS Marketplace は、販売者が製品の販売に関する情報を収集および分析するのに役立つ多数のデータフィードを提供します。データフィードは、登録されたすべての AWS Marketplace 販売者が利用できます。データフィードは 1 日以内に生成されるため、最新のデータを利用できます。

以降のセクションでは、それぞれのデータフィードについて説明します。

トピック

- [アカウントデータフィード](#)
- [住所データフィード](#)
- [請求イベントデータフィード](#)
- [レガシーマッピングデータフィード](#)
- [オファーデータフィード](#)
- [オファー製品データフィード](#)
- [オファーターゲットデータフィード](#)
- [製品データフィード](#)
- [税品目データフィード](#)

アカウントデータフィード

このデータフィードは、ユーザーが操作するすべてのアカウント (ユーザー自身、使用しているチャネルパートナー、購入者、支払者、およびすべての課税対象アカウント) に関する情報を提供します。

アカウントデータはイミュータブルであり、バージョン番号は関連付けられません。フィールドに対する変更は付け加えられるため、このデータフィードには同じ `account_id` 値と、異なる `valid_from` 値を持つ複数の行が存在する場合があります。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

アカウントデータフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィードの列の名前と説明を示しています。

列名	説明
<code>account_id</code>	アカウントのグローバル一意識別子 (GUID)。

列名	説明
	Product、Offer_Target、Billing_Event、Tax_Item およびデータフィードのフィールドへの結合に使用できます。結合に使用できるフィールドについては、これらのデータフィードを参照してください。
aws_account_id	販売者の AWS アカウントのアカウント AWS 番号。AWS パーティションごとに一意です。
encrypted_account_id	アプリケーションの購入者別の一意の暗号化された ID。の値は Metering Service AWS Marketplace によって encrypted_account_id 使用され、例えば、CustomerIdentifier ResolveCustomer アクションによって返される の値として使用されます。
mailing_address_id	このアカウントの郵送先住所の参照。
tax_address_id	このアカウントの税宛先住所の参照。
tax_registration_number	米国以外のアカウントの場合、このアカウントの税登録番号。
tax_legal_name	米国以外のアカウントの場合、法人名。これは、税金請求書で使用される名前です。

アカウントデータフィードの例

アカウントデータフィードの例は次のとおりです。見やすいように、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

account_id	aws_account_id	encrypted_account_id	mailing_address_id	tax_address_id	tax_registration_number	tax_legal_name
xk0CSmiAm 6PQ4QqEog 9iiaochlz uPlkMfba7 a1oDIZ	444456660 000	Zf7oMzheG WpH	25o3k46eN 6eViOfFii qtxwX8e3k aOiPalUio fjyFa3			
7nyo5jwTR oPlyX81vx 9ji04eEwT urO1Ff8bi Qi88W8	555567679 999	373vuQUqm Q8v	5oJ6vTjSz MrrF2gvh2 Vj9HfqiM8 00MuLEHmy FY5Lr42s8	5oJ6vTjSz MrrF2gvh2 Vj9HfqiM8 00MuLEHmy FY5Lr42s8	SE8239350 83345	
VleGa2t9j 3MuxioH9w c8lsndXXC gGCGUreeX riocM5	737399988 88	8SPxAYmi8 MwX	NLUc5UeiM IGFTTrDWCo ftDPhDUF1 oaSd8xgl5 QM8Db7	V5NhBYBiY ogwy0WMhn dGU4AfMgg muoTC2j7P m8ZKKNnyT	DE4695580 25	

住所データフィード

Important

AWS Marketplace は、2023 年 12 月に販売者配信データフィードサービスを使用して配信される PIIFeed を廃止します。 <https://docs.aws.amazon.com/marketplace/latest/userguide/data-feed-service.html> アドレスデータのニーズには AddressFeed_V1 データフィードを使用します。質問がある場合やサポートが必要な場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

このデータフィードは、ユーザーが操作するすべてのアカウント (ユーザー自身、使用しているチャネルパートナー、購入者、支払者、およびすべての課税対象アカウント) に関する連絡先情報を提供します。新しい取引が発生するたびに、その取引の顧客住所がスキャンされ、その住所がデータフィードにない場合は、新しいエントリがデータフィードファイルに追加されます。

住所データはイミュータブルです。

住所データフィールドは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィールドの列の名前と説明を示しています。

列名	説明
address_id	住所の一意のキー。 billing_address_id フィールドの Billing_Event データフィールド、mailing_address_id または tax_address_id フィールドの Account データフィールドからの結合に使用できます。
email_domain	このアカウントの登録済み E メールアドレスのドメイン。
company_name	このアカウントの登録済み会社名。
country_code	この住所の登録済み ISO 3166 alpha-2 国コード。
state_or_region	この住所の登録済み都道府県またはリージョン。
city	この住所の登録済み市区町村。
postal_code	この住所の登録済み郵便番号。
address_line_1	この住所の登録済み行 (1 行目)。
address_line_2	この住所の登録済み行 (2 行目)。
address_line_3	この住所の登録済み行 (3 行目)。

住所データフィールドの例

住所データフィールドの例を次に示します。この情報は、データフィールドの単一のテーブルに表示されます。見やすいように、ここではデータを2つのテーブルに表示し、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

address_id	email_domain	company_name	country_code	state_or_region	city	postal_code
V5NhBYBiYogwy0WMhndGU4AfMggmuoTC2j7Pm8ZKKNNyT	a.com	Mateo Jackson's Company	DE		Hamburg	67568
G68xdbkZQDVVHzfBGw6yf5yos0A6NiSVWHmT5ViLjf	b.com	Mary Major's Company	米国	OH	Dayton	57684
NLUc5UeiMIGFTTrDWCoftDPhDUF1oaSd8xgl5QM8Db7	c.com	Our Seller	米国	NY	New York	89475

address_line_1	address_line_2	address_line_3
	19th Floor	

請求イベントデータフィード

このデータフィードは、請求や支払いなど、請求イベントに関する情報を提供します。

たとえば、このデータフィードを使用して、購入者への請求日や請求内容を確認できます。[サンプルの SQL クエリ](#)を使用して、このデータフィードのデータを分析することもできます。

このデータフィードは、登録販売者としての請求イベントに関連する情報を示します。チャネルパートナーを介して締結された契約の場合、このデータフィードはメーカーと登録販売者との請求イベントに関する情報を示します。

請求イベントデータフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

請求イベントデータはイミュータブルです。

以下の表では、データフィードの列の名前と説明を示しています。

列名	説明
billing_event_id	請求イベントの識別子。この ID は、販売者の環境内で一意です。
from_account_id	請求イベントを開始したアカウント。transaction_type が SELLER_REV_SHARE である場合は、購入者の支払いアカウントです。これは、 アカウント データフィードへの外部キーです。 account_id フィールドの Account データフィードへの結合に使用できます。
to_account_id	製品の取引金額を受け取るアカウント。これは、アカウントデータフィードへの外部キーです。 account_id フィールドの Account データフィードへの結合に使用できます。
end_user_account_id	製品を使用するアカウント。このアカウントは、購入者アカウントや支払者アカウントとは異なる場合があります。

列名	説明
product_id	<p>account_id フィールドの Account データフィールドへの結合に使用できます。</p> <p>製品の識別子。これは 製品 データフィールドへの外部キーです。</p> <p>product_id フィールドの Product データフィールドへの結合に使用できます。</p>
アクション	<p>このイベントのアクションのタイプ。可能な値は以下のとおりです。</p> <ul style="list-style-type: none">• INVOICED - 購入者に該当金額を請求しました。• FORGIVEN - 購入者に該当金額を請求し、AWS がその請求を取り消しました。• DISBURSED - 販売者に該当金額が支払われました。これは、1 か月分の請求書をまとめるか、オンデマンド支払いとすることができます。

列名	説明
transaction_type	<p>取引のタイプ。例については、「課税シナリオ」を参照してください。可能な値は以下のとおりです。</p> <ul style="list-style-type: none"> • SELLER_REV_SHARE - 正の金額。販売者が購入者との契約で設定した料金です。 • SELLER_TAX_SHARE - 正の金額。販売者が支払う税金として SELLER_REV_SHARE に追加される金額です。 • AWS_REV_SHARE - 負の金額。出品手数料です。 • AWS_TAX_SHARE - 正の金額。これは、に加えて AWS 徴収される税金の金額です SELLER_REV_SHARE 。この金額は販売者の残高には影響しません。この金額は支払い対象ではなく、購入者に請求される税額が販売者に代わって当局に支払われることを販売者に知らせるために提供されます。 • <i>transaction_type</i> _REFUND - 購入者からリクエストされた返金額。 • <i>transaction_type</i> _CREDIT - 購入者に AWS クレジットを付与する金額。 • BALANCE_ADJUSTMENT - 請求の問題を解決 AWS するために によって行われた調整。 • DISBURSEMENT - action の値が DISBURSED で、balance_impacting の値が 1 の場合、これは販売者に支払われる金額です。action の値が INVOICED である場合、このレコードは parent_billing_event_id レコードを全面的または部分的に無効にします。この場合、関連する支払い disbursement_bill

列名	説明
	<p>ng_event_id が表示され、balance_impacting の値は 0 です。</p> <ul style="list-style-type: none">• DISBURSEMENT_FAILURE - 取引を無効にします。

列名	説明
parent_billing_event_id	<p>broker_id の値が AWS_INC で、action の値が DISBURSED または FORGIVEN で、transaction_type の値が DISBURSEMENT の場合、parent_billing_event_id はこの請求イベントを開始した元の billing_event_id の金額を指します。action に別の値がある場合、このフィールドは null です。</p> <p>broker_id の値が AWS_EUROPE の場合、parent_billing_event_id は次のシナリオで請求イベントを開始した元の billing_event_id を指します。</p> <ul style="list-style-type: none"> • action の値は FORGIVEN または INVOICED で、transaction_type の値は、AWS_REV_SHARE、AWS_REV_SHARE_REFUND、または SELLER_REV_SHARE_REFUND です。 • action の値は DISBURSED で、transaction_type の値は ANY (DISBURSEMENT_FAILURE を除く) です。 • transaction_type の値は、AWS_TAX_SHARE、AWS_TAX_SHARE_REFUND、SELLER_TAX_SHARE、または SELLER_TAX_SHARE_REFUND です。 • action の値は DISBURSED で、transaction_type の値は DISBURSEMENT_FAILURE です。 <p>broker_id の値が AWS_EUROPE の場合、parent_billing_event_id は次の</p>

列名	説明
	<p>シナリオで前回成功した支払い請求イベントの元の <code>billing_event_id</code> を指します。</p> <ul style="list-style-type: none"> <code>action</code> の値は <code>DISBURSED</code> で、<code>transaction_type</code> の値は <code>DISBURSEMENT_FAILURE</code> です。 <p><code>broker_id</code> の値が <code>AWS_EUROPE</code> の場合、残りのシナリオではこのフィールドは <code>null</code> になります。</p>
<code>disbursement_billing_event_id</code>	<p><code>action</code> の値が <code>DISBURSED</code> であり、さらに次のいずれかに該当する場合の関連支払額。</p> <ul style="list-style-type: none"> <code>transaction_type like ('SELLER%')</code> <code>transaction_type like ('AWS%')</code> <p>他のすべてのシナリオでは、この値は <code>null</code> です。</p>
<code>amount</code>	請求イベントの金額。
<code>currency</code>	ISO 639 通貨コード。
<code>balance_impacting</code>	<p>販売者の支払いを計算する際に該当金額を考慮するかどうか。0 の値は、該当金額が情報提供のために表示されており、残高には影響しないことを示します。1 の値は、該当金額が販売者の支払いを決定する際に考慮されることを示します。</p>
<code>invoice_date</code>	請求書が作成された日付。
<code>payment_due_date</code>	<code>action</code> の値が <code>INVOICED</code> である場合は、請求書の支払期限日です。

列名	説明
usage_period_start_date	レコードの期間の開始日。
usage_period_end_date	レコードの期間の終了日。
invoice_id	AWS 請求書 ID。
billing_address_id	住所データフィールドにおける支払人の請求連絡先の参照。 address_id フィールドの Address データフィールドへの結合に使用できます。
transaction_reference_id	以下のレポートのデータを相互参照できる識別子。 <ul style="list-style-type: none">• 支払いレポート• 月別請求済み収益レポート• 販売補償レポート• 米国の消費税と使用税レポート
bank_trace_id	支払い取引 (transaction_type = DISBURSEMENT で、アクションはDISBURSED) の場合、銀行によって割り当てられたトレース ID。トレース ID は、銀行が提供する販売者銀行からのレポートと関連付けるために使用できます。

列名	説明
broker_id	<p>取引を促進した事業者の識別子。可能な値は以下のとおりです。</p> <ul style="list-style-type: none"> • AWS_INC <ul style="list-style-type: none"> - AWS、Inc. (米国を拠点とする) の識別子。 • AWS_EUROPE <ul style="list-style-type: none"> - Amazon Web Services EMEA SARL (ルクセンブルク拠点) の識別子。 • NULL <ul style="list-style-type: none"> - 明示的な broker_id のない以前の取引は、AWS_INC によって円滑に進められました。
buyer_transaction_reference_id	<p>GROUP BY または SQL で作成されたウィンドウ関数を使用して、請求フィールド内のすべての関連レコードをグループ化する識別子。これらの関連レコードには、購入者請求書、販売者請求書、および出品料金にかかる付加価値税 (VAT) が含まれる場合があります。</p>

課税シナリオ

購入者と販売者の国および州に適用されている課税モデルによって、税金の徴収と支払の方法が決まります。考えられるシナリオは次のとおりです。

- 税金は `BUYER_TAX_SHARE` によって収集され、送金されます AWS。この場合、`transaction_type` は `AWS_TAX_SHARE` です。
- 税金は、`SELLER_TAX_SHARE` によって収集され AWS、販売者に支払い、販売者が税務機関に送金されます。この場合、`transaction_type` は `SELLER_TAX_SHARE` です。
- 税金は `BUYER_TAX_SHARE` によって収集されません AWS。販売者が税金を計算して税務当局に納税する必要があります。このような場合、税金計算を実行 AWS Marketplace したり、税金情報を受け取ったりしません。販売者が収益分配金から納税します。

請求イベントデータフィールドの例

このセクションでは、請求時と 1 か月後の請求イベントデータ期間の例を示します。このセクションのすべての表について、次の点に注意してください。

- データフィールドの `billing_event_id` 値は 40 文字の英数字文字列です。ここでは、見やすいように 2 文字の文字列として表示しています。
- この情報は、データフィールドの単一のテーブルに表示されます。見やすいように、ここではデータを複数のテーブルに表示し、一部の列を省略しています。

このセクションの例では、以下を前提とします。

- Arnav は購入者です。
 - アカウント ID は 737399998888 です。
 - 居住国のフランスでは、マーケットプレイスファシリテーター法が適用されます。詳細については、「[アマゾン ウェブ サービス 税務ヘルプ](#)」を参照してください。
 - `prod-o4grxfafcx` を購入し、この製品の月間使用料として 120.60 USD を請求されました。
 - 当月内に請求額を支払いました。
- Jane はメーカーです。
 - アカウント ID は 111122223333 です。
- Paulo は登録販売者です。
 - アカウント ID は 777788889999 です。
 - 居住しているカンザス州は、マーケットプレイスファシリテーター法の対象ではありません。

登録販売者の請求イベントデータフィールド

Paulo は、登録販売者として購入者の Arnav に請求を行います。

次の表は、Paulo が Arnav に請求した際のデータフィールドの関連情報を示しています。

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I0	737399998888	777788889999	737399998888	prod-o4grxfafcx	INVOICED	SELLER_RE V_SHARE

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I1	737399998888	AWS	737399998888	prod-o4grxfafcxxxx	INVOICED	AWS_TAX_SHARE
I2	777788889999	111122223333	737399998888	prod-o4grxfafcxxxx	INVOICED	SELLER_REV_SHARE
I3	777788889999	AWS	737399998888	prod-o4grxfafcxxxx	INVOICED	AWS_REV_SHARE

parent_billing_event_id	disbursement_billing_event_id	amount	currency	invoice_date	invoice_id
		100	USD	2018-12-31T00:00:00Z	781216640
		20.6	USD	2018-12-31T00:00:00Z	781216640
		-80	USD	2018-12-31T00:04:07Z	788576665
		-0.2	USD	2018-12-31T00:04:07Z	788576665

次の表は、Arnav が請求額を支払った後の月末における Paulo のデータフィードの関連情報を示しています。

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I10	737399998888	777788889999	737399998888		DISBURSED	SELLER_REV_SHARE

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I12	777788889999	111122223333	737399998888		DISBURSED	SELLER_REV_SHARE
I13	777788889999	AWS	737399998888	prod-o4grxfafcxxxx	DISBURSED	AWS_REV_SHARE
I14	AWS	777788889999			DISBURSED	DISBURSEMENT

parent_billing_event_id	disbursement_billing_event_id	amount	currency	invoice_date	invoice_id
I0	I14	-100	USD	2018-12-31T00:00:00Z	781216640
I2	I14	80	USD	2018-12-31T00:04:07Z	788576665
I3	I14	0.2	USD	2018-12-31T00:04:07Z	788576665
		19.8	USD		

メーカーの請求イベントデータフィード

次の表は、Paulo が Arnav に請求した際の Jane のデータフィードの関連情報を示しています。

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I5	777788889999	111122223333		prod-o4grxfafcxxxx	INVOICED	SELLER_REV_SHARE

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
16	77778888999	11112222333		prod-o4grxfafcxxxx	INVOICED	SELLER_TAX_SHARE
17	11112222333	AWS		prod-o4grxfafcxxxx	INVOICED	AWS_REV_SHARE

parent_billing_event_id	disbursement_billing_event_id	amount	currency	invoice_date	invoice_id
		73.5		2018-12-3 1T00:04:07Z	788576665
		6.5		2018-12-3 1T00:04:07Z	788576665
		-7.35		2018-12-3 1T00:04:07Z	788576665

次の表は、請求額が支払われた後の月末における Jane のデータフィードの関連情報を示しています。

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
130	77778888999	11112222333		prod-o4grxfafcxxxx	DISBURSED	SELLER_REV_SHARE
131	77778888999	11112222333		prod-o4grxfafcxxxx	DISBURSED	SELLER_TAX_SHARE
132	11112222333	AWS		prod-o4grxfafcxxxx	DISBURSED	AWS_REV_SHARE

billing_event_id	from_account_id	to_account_id	end_user_account_id	product_id	アクション	transaction_type
I33	AWS	11112222333			DISBURSED	DISBURSEMENT

parent_billing_event_id	disbursement_billing_event_id	amount	currency	invoice_date	invoice_id
I5	I33	-73.5	USD		
I6	I33	-6.5	USD		
I7	I33	7.35	USD		
		72.65	USD		

クエリの例

[データフィードの使用](#) で説明しているように、[Athena](#) を使用して、マネージド Amazon S3 バケットにデータフィードとして収集されて保存されたデータに対してクエリを実行できます。このセクションでは、これを行う一般的な方法の例をいくつか示します。すべての例で、単一の通貨を使用するものとします。

例 1: 請求金額 (税込み)

購入者への請求金額 (税込み) を確認するには、次の例に示すようなクエリを実行できます。

```
SELECT sum(amount) FROM billing_event
WHERE
  action = 'INVOICED'
  AND
  (
    (transaction_type in ('SELLER_REV_SHARE', 'SELLER_TAX_SHARE'))
    -- to discard SELLER_REV_SHARE from Manufacturer to Channel Partner, aka cost of
    goods
    AND to_account_id='seller-account-id'
```

```

)
OR transaction_type= 'AWS_TAX_SHARE'
);

```

例 2: 販売者に代わって購入者に請求する金額

販売者に代わって購入者に請求した金額を確認するには、次の例に示すようなクエリを実行できます。

```

SELECT sum(amount) FROM billing_event
WHERE
  action = 'INVOICED'
  AND transaction_type in ('SELLER_REV_SHARE', 'SELLER_TAX_SHARE')
  AND to_account_id='seller-account-id'
;

```

例 3: 販売者に代わって金額が収集 AWS できる

販売者に代わって が AWS 収集する金額から返金、クレジット、および許可アカウントを引いた金額を調べるには、次の例に示すようにクエリを実行できます。

```

SELECT sum(amount) FROM billing_event
WHERE
  -- what is invoiced on behalf of SELLER, incl. refunds/ credits and cost of goods
  transaction_type like 'SELLER_%'
  -- FORGIVEN action records will "negate" related INVOICED
  and action in ('INVOICED','FORGIVEN')
;

```

例 4: 販売者が徴収できる金額

販売者が徴収できる金額を確認するには、次の例に示すようなクエリを実行できます。この例では、 が AWS 収集する出品手数料と税金を削除し、例外的な残高調整を追加します。

```

SELECT sum(amount) FROM billing_event
WHERE
  (transaction_type like 'SELLER_%' -- what is invoiced on behalf of SELLER
  or transaction_type like 'AWS_REV_%' -- what is owed to AWS
  or transaction_type = 'BALANCE_ADJUSTMENT' -- exceptionnal case
  )
  and action in ('INVOICED','FORGIVEN')
;

```

また、次の例に示すように、次のクエリを使用して同じ情報を収集することもできます。

```
SELECT sum(amount) FROM billing_event
WHERE
  balance_impacting = 1
  and action in ('INVOICED','FORGIVEN')
;
```

次の例は同じ情報を示していますが、2018年の取引に限定し、すべての購入者が請求額を支払ったと仮定しています。

```
SELECT sum(amount) FROM billing_event
WHERE
  invoice_date between '2018-01-01' and '2018-12-31'
  and balance_impacting = 1
  and action in ('INVOICED','FORGIVEN')
;
```

例 5: 支払い額

支払われた金額を確認するには、次の例に示すようなクエリを実行できます。

```
select sum(amount) FROM billing_event
WHERE
  action = 'DISBURSED'
  and transaction_type like 'DISBURSEMENT%'
;
```

例 6: 保留中の支払い金額

保留中の支払い金額を確認するには、次の例に示すようなクエリを実行できます。このクエリでは、支払い済みの金額を除外します。

```
SELECT sum(amount) FROM billing_event targeted
WHERE
  (transaction_type like 'SELLER_%' -- what is invoiced on behalf of SELLER
  or transaction_type like 'AWS_REV_%' -- what is owed to AWS
  or transaction_type = 'BALANCE_ADJUSTMENT' -- exceptionnal case
  )
  -- DISBURSEMENT action records will "negate" 'INVOICED'
  -- but do not take into account failed disbursements
AND
```

```
(not exists
  (select 1
    from billing_event disbursement
      join billing_event failed_disbursement
        on disbursement.billing_event_id=failed_disbursement.parent_billing_event_id
    where
      disbursement.transaction_type='DISBURSEMENT'
      and failed_disbursement.transaction_type='DISBURSEMENT_FAILURE'
      and targeted.disbursement_billing_event_id=disbursement.billing_event_id
    )
  )
);
```

同じ情報を取得する別の方法としては、次の例に示すようなクエリを実行して販売者の残高を取得します。

```
SELECT sum(amount) FROM billing_event
WHERE
  balance_impacting = 1
;
```

次のクエリでは、この例を拡張します。クエリ結果を 2018 年の取引に限定し、これらの取引に関する追加の詳細を返します。

```
select sum(residual_amount_per_transaction)
from
  (SELECT
    max(billed_invoices.amount) invoiced_amount,
    sum(nvl(disbursed_invoices.amount,0)) disbursed_amount,
    -- Exercise left to the reader:
    -- use transaction_type to distinguish listing fee vs seller-owed money
    -- still pending collection
    max(transaction_type) transaction_type,
    max(billed_invoices.amount)
    + sum(nvl(disbursed_invoices.amount,0)) residual_amount_per_transaction
  FROM billing_event billed_invoices
    -- find related disbursements
  left join billing_event disbursed_invoices
    on disbursed_invoices.action='DISBURSED'
    and disbursed_invoices.parent_billing_event_id=billed_invoices.billing_event_id
  WHERE
    billed_invoices.invoice_date between '2018-01-01' and '2018-12-31')
```

```
and billed_invoices.transaction_type like 'SELLER_%' -- invoiced on behalf of
SELLER
and billed_invoices.action in ('INVOICED','FORGIVEN')
-- do not take into account failed disbursements
AND not exists
  (select 1 from billing_event failed_disbursement
   where disbursed_invoices.disbursement_billing_event_id =
failed_disbursement.parent_billing_event_id
  )
GROUP BY billed_invoices.billing_event_id
);
```

例 7: 一連の請求書の残高

一連の請求書の合計額を確認するには、次の例に示すようなクエリを実行できます。

```
SELECT invoice_id, sum(amount) FROM billing_event targeted
WHERE
  -- invoice_id is only not null for invoiced records AND disbursed records
  -- linking them to related disbursement -> no need to filter more precisely
  invoice_id in ('XXX','YYY')
  -- filter out failed disbursements
AND not exists
  (select 1
   from billing_event disbursement
   join billing_event failed_disbursement
     on disbursement.billing_event_id=failed_disbursement.parent_billing_event_id
   where
     disbursement.transaction_type='DISBURSEMENT'
     and failed_disbursement.transaction_type='DISBURSEMENT_FAILURE'
     and targeted.disbursement_billing_event_id=disbursement.billing_event_id
  )
group by invoice_id;
```

レガシーマッピングデータフィード

このデータフィードは、製品 ID とオファー ID をレガシーのグローバル一意識別子 (GUID) にマッピングする方法を示します。レガシー GUIDs は古いレポートで使用され、新しい IDs はデータフィードと AWS Marketplace APIs で使用されます。

このデータフィードは、登録販売者として作成したすべての製品と、再販が許可されているすべての製品に関する情報を提供します。

レガシーマッピングデータフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィードの列の名前と説明を示しています。

列名	説明
mapping_type	製品 ID とオファー ID のいずれであることを示します。
legacy_id	この製品またはオファーのレガシー ID。
new_id	この製品またはオファーのわかりやすい ID。 この ID は、現在のすべての API アクションで主キーとして使用されます。

レガシーマッピングデータフィードの例

レガシーマッピングデータフィードの例は次のとおりです。見やすいように、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

mapping_type	legacy_id	new_id
OFFER	8a806c74-dbd6-403e-9362-bb08f417ff37	offer-dacpxznflfwin
PRODUCT	1368541d-890b-4b6c-9bb9-4a55306ab642	prod-o4grxfafcxxxxy
OFFER	558d8382-6b3a-4c75-8345-a627b552f5f1	offer-gszhmlle5npzip

オファーデータフィード

オファーデータフィードは、登録販売者として作成したすべてのオファーに関する情報を提供します。1つのオファーに複数のリビジョンがある場合、すべてのリビジョンがデータフィードに含まれます。

オファーリビジョンを作成し、公開フィールドのデータが変わると、同じプライマリキー (offer_id プラス offer_revision) データフィードに新しいレコードが作成されます。ただし、valid_from フィールドの値は異なります。データフィードの履歴列の詳細については、「[データの履歴化](#)」を参照してください。

オファーデータフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表には、データフィードの列の名前と説明が示されています。

列名	説明
offer_id	オファーのわかりやすい識別子。 Offer_Product データフィードの offer_id フィールドへの結合に使用できません。
offer_revision	オファーリビジョン。このフィールドと offer_id フィールドを組み合わせると主キーを形成します。 offer_id では、Target_Offer データフィードの offer_id と offer_revision フィールドの結合に使用できます。
name	販売者が定義したオファー名。
expiration_date	オファーの有効期限が切れる日時。
opportunity_name	このオファーにリンクされた営業案件データ。オファーが AWS の営業案件にバインドされている場合、このフィールドは入力済みです。

列名	説明
opportunity_description	このオファーにリンクされているすべての説明情報。オファーが AWS の営業案件にバインドされている場合、このフィールドは入力済みです。
seller_account_id	販売者のアカウントのグローバル一意識別子 (GUID)。アカウントデータフィードの account_id フィールドと結合するために使用できます。
opportunity_id	機会の ID は、リセラーが製品を販売している場合にのみ入力されます。製品が同じであれば、異なるチャネルパートナー (または販売者) が作成したオファーにはすべて同じ opportunity_id があります。

オファーデータフィードの例

オファーデータフィードの例は次のとおりです。見やすいように、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

offer_id	offer_rev ision	name	expiratio n_date	opportuni ty_name	opportuni ty_descri ption	seller_ac count_id	opportuni ty_id
offer-dac pxznflfwi n	1	エンター プライズ 契約プロ グラムオ ファー	9999-01-0 1T00:00:0 0Z				
offer-gsz hmle5npzi p	1	販売者 が作成し たプライ	2020-10-3 1T00:00:0 0Z				

offer_id	offer_revision	name	expiration_date	opportunity_name	opportunity_description	seller_account_id	opportunity_id
		ベートオファー					
offer-hmzhyle8nphlp	1	エンタープライズ契約プログラムオファー	9999-01-01T00:00:00Z				

オファー製品データフィード

1つのオファーには複数の製品を含めることができ、1つの製品は複数の異なるオファーに含めることができます。このデータフィードは、オファーと製品との関係に関する情報を示します。

このデータフィードは、登録販売者として作成したすべての製品オファーに関する情報を提供します。

オファーに対して製品を追加または削除した場合は、オファーリビジョンを作成できます。

オファー製品データフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィードの列の名前と説明を示しています。データフィードの履歴列の詳細については、「[データの履歴化](#)」を参照してください。

列名	説明
offer_id	このオファーのわかりやすい識別子。 Offer データフィードの offer_id フィールドへの結合に使用できます。

列名	説明		
offer_revision	offer_id フィールドと組み合わせて、オファーリビジョンへの外部キーを形成します。		
product_id	製品のわかりやすい識別子。これは、このオファーで公開する製品への外部キーです。 Product データフィールドの product_id フィールドへの結合に使用できます。		

オファー製品データフィールドの例

オファー製品データフィールドの例は次のとおりです。

offer_id	offer_revision	product_id
offer-dacpxznflfwin	10	prod-o4grxfafcxxxx
offer-gszhmle5npzip	24	prod-o4grxfafcxxxxy

オファーターゲットデータフィールド

このデータフィールドは、登録販売者として作成したすべてのオファーに関するオファーのリビジョンのターゲットを示します。1つのオファーに複数のリビジョンがある場合、すべてのリビジョンがデータフィールドに含まれます。

オファーリビジョンを作成し、公開フィールドのデータが変わると、別のレコードがデータフィールドに作成されます。このレコードの主キー (offer_id プラス offer_revision) は同じですが、valid_from フィールドの値は異なります。

オファーターゲットデータフィールドは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィールドの列の名前と説明を示しています。

列名	説明
offer_target_id	フィールドの主キー。
offer_id+offer_revision	オファターの識別子とリビジョン。これら 2 つの列は、このターゲットに関連するオファターを参照します。 Targetデータフィールドの offer_id および offer_revision フィールドへの結合に使用できます。
target_type	オファターの受取人が BuyerAccounts (プライベートオファターを示す) であるか、ParticipatingPrograms であるかを示します。
polarity	オファターが target_type を対象にしているかどうかを示します。指定できる値は次のとおりです。 <ul style="list-style-type: none"> PositiveTargeting オファターは、この target_type に適用されます。 NegativeTargeting オファターは、この target_type に適用されません。
value	ターゲットを表す文字列: AWS アカウント ID またはオファターで使用できるプログラム。例: AWS Marketplace の標準契約 (SCMP) 、または

列名	説明
	AWS Marketplace フィールドデモンストレーションプログラム (FDP) 。

オファーターゲットデータフィードの例

オファーターゲットデータフィードの例を次に示します。見やすいように、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

offer_target_id	offer_id	offer_revision	target_type	polarity	value
925ddc73f6a373b7d5544ea3210610803b600	offer-dacpxznflfwin	1	ParticipatingPrograms	PositiveTargeting	EnterpriseContract
471ff22ae3165278f1fb960d3e14517bcd601	offer-gszhmle5npzip	1	ParticipatingPrograms	PositiveTargeting	FieldDemonstration
511ff22adfj65278f1fb960d3e14517bcd6e602	offer-gszhmle5npzip	1	ParticipatingPrograms	PositiveTargeting	EnterpriseContract

製品データフィード

このデータフィードは、登録販売者として作成したすべての製品と、再販が許可されているすべての製品に関する情報を提供します。

製品データは変更可能です。つまり、次のいずれかのフィールドの値を変更すると、valid_from フィールドの値が異なる別のレコードがデータフィードに作成されます。データフィードの履歴列の詳細については、「[データの履歴化](#)」を参照してください。

製品データフィードは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

以下の表では、データフィードの列の名前と説明を示しています。

列名	説明
product_id	製品のわかりやすい識別子。 Account、Billing_Event、Offer_Product データフィードの product_id フィールドと結合するために使用できます。
manufacturer_account_id	製品の所有者の識別子。これは、 アカウント データフィードへの外部キーです。 Account データフィードの account_id フィールドへの結合に使用できます。
product_code	製品の計測に使用される、権利が付与された既存の製品コード。この値は、データをレポートに結合する場合や、AWS Marketplace Metering Service で提供される情報を参照する場合にも使用されます。
title	製品のタイトル。

製品データフィードの例

オファーターゲットデータフィードの例を次に示します。見やすいように、データの履歴列は表示していません。データの履歴フィールドの詳細については、「[データの履歴化](#)」を参照してください。

product_id	manufacturer_account_id	product_code	title
prod-o4grxfafcxxxx	555568000000	product_code_1	Product1
prod-t3grxfafcxyy	444457000000	product_code_2	Product2
prod-x8faxxfafcxyy	666678000000	product_code_3	Product3

税品目データフィールド

このデータフィールドは、顧客請求書の税額計算に関する情報を提供します。

特定の顧客請求書 (invoice_id) の特定の製品 (product_id) に対して複数の明細行品目 (line_item_id) が存在する場合があります。顧客請求書は各税務管轄区域に 1 つ以上存在する場合があります。これは、例えば、異なる AWS エンティティ (米国とアイルランドなど) によって異なる AWS リージョンルールを使用している顧客に対する使用量ベースの請求で発生します。が売上に対する売上税、VAT、または GST を AWS 徴収し、AWS, Inc. の名でそのような税金を現地の税務機関に送金する場所の詳細については、[「Amazon Web Service Tax Help」](#) を参照してください。

税品目データフィールドは 24 時間ごとに更新されるため、新しいデータを毎日利用できます。

税項目データはイミュータブルです。

以下の表では、データフィールドの列の名前と説明を示しています。データの履歴列の詳細については、「[データの履歴化](#)」を参照してください。

列名	説明
tax_item_id	税品目レコードの一意の識別子。
invoice_id	AWS 請求書 ID。この値と product_id の値を組み合わせて、関連する税金請求イベントを検索できます。
line_item_id	顧客請求明細項目の一意の識別子。返金取引の明細項目 ID は先渡し取引税のものと同じです。
customer_bill_id	顧客請求書の一意の識別子。購入者は、この識別子を販売者と共有し、税額計算に関する質問を特定して解決できます。
tax_liable_party	AWS または Seller です。販売者が納税義務者である場合は、税金が徴収されます。AWS が納税義務当事者である場合、売上税はによって徴収および送金されます AWS。詳細については、「 AWS Marketplace Sellers & Tax Collection 」を参照してください。

列名	説明
	徴税しない場合、ここには値が表示されません。販売者は、徴税の義務があるため、請求書ごとに徴税されたかどうかを確認する必要があります。
transaction_type_code	取引のタイプ。可能な値は以下のとおりです。 <ul style="list-style-type: none">• AWS - 源泉徴収取引• REFUND - 全額返金または一部返金• TAXONLYREFUND - 税のみの返金 返金取引の明細項目 ID は元の先渡し取引のものと同じです
product_id	製品の外部キー。 product_id フィールドの Product データフィールドへの結合に使用できます。
product_tax_code	製品の税プロパティを識別するための標準コード。販売者は、製品の作成時または変更時にプロパティを選択します。
invoice_date	請求書が作成された日付。
taxed_customer_account_id	課税対象の取引先企業エンティティの外部キー。 account_id フィールドの Account データフィールドへの結合に使用できます。
taxed_customer_country	税額計算に使用する住所の ISO 3166 alpha 2 の国別コード。
taxed_customer_state_or_region	税額計算に使用する州、区、郡。
taxed_customer_city	税額計算に使用する市。

列名	説明
taxed_customer_postal_code	税額計算に使用する郵便番号。
tax_location_code_taxed_jurisdiction	課税場所に関連付けられている頂点ジオコード。
tax_type_code	取引に適用される税のタイプ。指定できる値は、None、Sales および SellerUse です。
jurisdiction_level	納税地に使用される住所の管轄区域。指定できる値は、State、County、City、District です。
taxed_jurisdiction	税務管轄区域の名前。
display_price_taxability_type	購入者に料金を税込みで表示するか、税抜きで表示するか。すべての AWS Marketplace サービスには税金は含まれていません。
taxable_amount	該当する管轄区域レベルで課税対象となる取引金額。
nontaxable_amount	該当する管轄区域レベルで非課税となる取引金額。
tax_jurisdiction_rate	該当する管轄区域レベルで適用される税率。
tax_amount	該当する管轄区域レベルで請求される税額。
tax_currency	上記の金額に関する ISO 4217 alpha 3 の通貨コード。
tax_calculation_reason_code	取引が課税対象となるか、非課税となるか、免除とされるか、ゼロ評価済みであるかを管轄区域別に整理したもの。
date_used_for_tax_calculation	取引の税金の計算に使用される日付。

列名	説明
customer_exemption_certificate_id	免除証明書の証明書 ID。
customer_exemption_certificate_id_domain	Amazon システム上の証明書の保存先。
customer_exemption_certificate_level	免除を提供した管轄区域。
customer_exemption_code	免税を指定するコード (RESALE など)。
customer_exemption_domain	顧客の免税情報 (ある場合) をキャプチャするために使用する Amazon システム。
transaction_reference_id	以下のレポートのデータを相互参照できる識別子。 <ul style="list-style-type: none"> • 支払いレポート • 月別請求済み収益レポート • 販売補償レポート • 米国の消費税と使用税レポート

Note

2021 年 8 月 5 日以降、AWS Marketplace 売上に対する国際 Marketplace 円滑化機関の税金は、税金項目データフィールドに入力されます。つまり、2021 年 8 月 5 日以降、請求イベントデータフィールドのすべての AWS_TAX_SHARE レコードと SELLER_TAX_SHARE レコードは、税項目データフィールドに対応するレコードを持つことが予想されます。

税品目データフィールドの例

税品目データフィールドの例を次に示します。この情報は、データフィールドの単一のテーブルに表示されます。見やすいように、ここではデータを複数のテーブルに表示し、一部の列を省略しています。

tax_item_id	invoice_id	line_item_id	customer_bill_id
6p2ni6tu041xagvhby anbgxl3xameha16txj oav_0001	781216640	710000000 00000000000	221000000 0000000000
6p2ni6tu041xagvhby anbgxl3xameha16txj oav_0002	781216640	530000000 00000000000	221000000 0000000000
f1r4jobxjzww8czdsr q4noue2uxd56j39wxw 0k7_0001	250816266	764000000 00000000000	572000000 0000000000
gfkjjjobxjzw56jgkrs rqgjdk52uxd56j39wg j567d_0002	280336288	764000000 00000000000	572439000 0000000000
wwk1qpvb8ran3geiw8 e3mp6dgs2 qj7wpkuwhgk1_0001	451431024	993000000 00000000000	123000000 0000000000
wwk1qpvb8ran3geiw8 e3mp6dgs2 qj7wpkuwhgk1_0002	451431024	993000000 00000000000	312000000 0000000000
fnohdid8kwgqq9lvii 2k30spn3ftgwihbe8h 75x_0001	229987654	921000000 00000000000	639000000 0000000000

tax_liable_party	transacti on_type_code	product_id	product_t ax_code	invoice_date
販売者	AWS	prod-o4gr xfafcxxx	AWSMP_SOF TWARE_RA	2018-12-3 1T00:00:00Z

tax_liable_party	transacti on_type_code	product_id	product_t ax_code	invoice_date
販売者	AWS	prod-o4gr xfafcxxx	AWSMP_SOF TWARE_RA	2018-12-3 1T00:00:00Z
販売者	AWS	prod-t3gr xfafcxxx	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-3 1T00:00:00Z
販売者	REFUND	prod-t3gr xfafcxxx	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-3 1T00:00:00Z
販売者	AWS	prod-x8fa xxfafcxy	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-3 1T00:00:00Z
販売者	TAXONLYRE FUND	prod-x8fa xxfafcxy	AWS_REMOT E_ACCESS_ SOFTWARE	2018-05-3 1T00:00:00Z
AWS	AWS	prod-wghj 8xfafrhgj	AWS_REMOT E_ACCESS_ SOFTWARE	2019-07-3 1T00:00:00Z

taxed_cus tomer_acc ount_id	taxed_cus tomer_country	taxed_cus tomer_sta te_or_region	taxed_cus tomer_city	taxed_cus tomer_pos tal_code
VleGa2t9j 3MuxioH9w c8lsndXXC gGCGUreeX riocM5	米国	GA	MILTON	48573-4839

taxed_customer_account_id	taxed_customer_country	taxed_customer_state_or_region	taxed_customer_city	taxed_customer_postal_code
VleGa2t9j 3MuxioH9w c8lsndXXC gGCGUreeX riocM5	米国	GA	MILTON	48573-4839
7nyo5jwTR oPlyX81vx 9ji04eEwT urO1Ff8bi Qi88W8	米国	NC	DURHAM	27517-4834
7nyo5jwTR oPlyX81vx 9ji04eEwT urO1Ff8bi Qi88W8	米国	NC	DURHAM	27517-4834
7nyo5jwTR oPlyX81vx 9ji04eEwT urO1Ff8bi Qi88W8	米国	TX	NOT APPLICABLE	75844-1235
7nyo5jwTR oPlyX81vx 9ji04eEwT urO1Ff8bi Qi88W8	米国	TX	HOUSTON	75844-1235
192a04213 13e41f069 b52962ed7 babf716291b688	米国	CT	NEW HAVEN	06002-2948

tax_location_code_taxed_jurisdiction	tax_type_code	jurisdiction_level	taxed_jurisdiction	display_price_taxability_type	taxable_amount	nontaxable_amount
460473664	セールス	状態	GA	除外	100	0
66301164	セールス	郡	FULTON	税抜き	0	100
692938178	SellerUse	状態	NC	除外	58.1	523.8
692938178	SellerUse	状態	NC	除外	-58.1	523.8
356794387	セールス	状態	TX	除外	1105.14	0
528887443	セールス	市町村	HOUSTON	除外	-36	0
171248162	セールス	状態	CT	税抜き	0	114.55

tax_jurisdiction_rate	tax_amount	tax_currency	tax_calculation_reason_code	date_used_for_tax_calculation
0.206	20.6	USD	課税対象	2018-10-3 1T00:00:00Z
0	0	USD	NonTaxable	2018-10-3 1T00:00:00Z
0.1	5.8	USD	課税対象	2018-07-3 1T00:00:00Z
0.1	-5.8	USD	課税対象	2018-07-3 1T00:00:00Z
0.06	66.3	USD	課税対象	2018-07-3 1T00:00:00Z

tax_jurisdiction_rate	tax_amount	tax_currency	tax_calculation_reason_code	date_used_for_tax_calculation
0.01	-0.36	USD	NonTaxable	2018-07-31T00:00:00Z
0	0	USD	免税	2019-06-30T00:00:00Z

販売者レポート

AWS Marketplace は、製品の使用状況、購入者、請求、支払い情報に関する情報を含むレポートを提供します。レポートは、登録済みのすべての AWS Marketplace 販売者が利用できます。

レポートの生成に関する重要なポイントは、次のとおりです。

- レポートは、その種類に応じて毎日、毎週、または毎月生成されます。
- レポートは 00:00 UTC に生成され、前日の 24:00 UTC までを対象とします。
- レポートは .csv ファイルとして生成されます。
- 暗号化された S3 バケットにデータが配信されたときに通知するように Amazon SNS を設定できます。通知を設定すると、は、で登録した AWS アカウントに関連付けられた E メールアドレスに通知 AWS を送信します AWS Marketplace。

通知を設定する方法の詳細については、「Amazon Simple Notification Service デベロッパーガイド」の「[Amazon SNS の開始方法](#)」を参照してください。

通知 E メールを受信をキャンセルするには、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

- 各レポートの詳細については、[サンプルレポート](#)をダウンロードしてください。

レポートへのアクセス

AWS Marketplace には、レポートを設定する 2 つの方法があります。

- API インターフェイスの使用。 [AWS Marketplace コマース分析サービス](#) を使用する。この方法では API インターフェイスを介してレポート内のデータに自動的にアクセスできます。情報の取り込みを自動化し、レポート全体ではなくレポートの一部をダウンロードできます。このサービスは、従来の API の場合のように直接ではなく、Amazon Simple Storage Service (Amazon S3) のファイルに非同期的にデータを返します。データはコンピュータが読み取り可能な形式で配信されるため、システムにデータをインポートしたり組み込んだりすることができます。
- [AWS Marketplace 管理ポータル](#) のレポートダッシュボードを使用する。このダッシュボードは、以前のレポート期間のレポートを提供します。

AWS Identity and Access Management (IAM) アクセス許可を使用して、レポートへのアクセスを制御できます。

日別ビジネスレポート

日次ビジネスレポートは、AWS 顧客が製品を毎日どのように使用しているか、およびその使用による推定収益を理解するのに役立ちます。該当する情報が使用可能な場合にのみ、このレポートを受け取ります。受け取るべきレポートが届かない場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

このレポートは、[AWS Marketplace 管理ポータル](#) でアクセスできます。[the section called “AWS Marketplace コマース分析サービス”](#) に登録している場合は、AWS SDK を使用してレポートにアクセスすることもできます。

各顧客に一意的識別子を使用して、長期にわたってレポート間で顧客を識別できます。この識別子を使用すると、顧客の使用量を追跡して、顧客の出費を見積もり、無料トライアルの使用量、および年間の使用傾向を把握できます。

掲載スケジュール

このレポートは毎日 00:00 UTC に発行され、前日の 00:00 UTC から 23:59 UTC までを対象としています。このスケジュールの例外は日別ビジネスレポートセクションの冒頭に示しています。

トピック

- [セクション 1: インスタンスタイプ別の使用量](#)
- [セクション 2: 料金](#)
- [セクション 3: 無料トライアルのコンバージョン数](#)

- [セクション 4: 新しいインスタンス](#)
- [セクション 5: 新規の製品サブスクライバー](#)
- [セクション 6: キャンセルした製品サブスクライバー](#)

セクション 1: インスタンスタイプ別の使用量

このセクションでは、顧客が使用するインスタンスタイプごとに 1 行ずつデータが一覧表示されます。たとえば、顧客が 1 つのインスタンスタイプで製品を使用し、別のインスタンスタイプで同じ製品を使用している場合、レポートには 2 つのインスタンスタイプそれぞれの行が含まれます。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
ユーザーの都道府県	製品をサブスクライブしているアカウントに関連付けられている請求先住所の都道府県。
ユーザーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
製品タイトル	製品のタイトル。
製品コード	製品の一意的識別子。
インスタンスタイプ	製品の使用量に関連付けられているインスタンスタイプ (t2.micro など)。
使用量の単位	レポート期間中に顧客が使用した使用量の単位数。
使用量の単位タイプ	顧客の使用量の単位。たとえば、時間や日です。

列名	説明
提供タイプの説明	製品提供タイプの説明。たとえば、製品は時間課金、無料トライアル、または年間サブスクリプションで提供されています。
予想収益	製品の使用量からの予想収益。請求は月末に確定されます。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリーは USD です。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意的識別子。
支払者リファレンス ID	アカウント ID ではない一意的識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意的識別子。

セクション 2: 料金

このセクションには、製品に関連付けられている定額制の取引 (年間、月間、SaaS 契約の製品料金、データ製品サブスクリプション料金など) が含まれます。このセクションのデータは、レポート

が生成される 72 時間前の 24 時間を対象としています。たとえば、レポートが 5 月 24 日に生成された場合、データは 5 月 21 日の 24 時間を対象としています。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
ユーザーの都道府県	製品をサブスクライブしているアカウントに関連付けられている請求先住所の都道府県。
ユーザーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
製品タイトル	製品のタイトル。
製品コード	製品の一意的識別子。
量	使用料。返金がある場合、この値はマイナスです。このエントリが AWS Marketplace SaaS 契約の場合、金額は契約全体ではなくディメンションの料金を表します。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリは USD です。
料金の説明	料金の理由 (月額料金、年間料金、返金など)。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。

列名	説明
顧客の市区町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
開始日	AWS Marketplace SaaS 契約またはデータ製品サブスクリプションの開始日。
終了日	AWS Marketplace SaaS 契約またはデータ製品サブスクリプションの終了日。
数量	契約で指定されているディメンションの単位数。
ディメンション	契約で指定されているディメンション。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意的識別子。
支払者リファレンス ID	アカウント ID ではない一意的識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意的識別子。

セクション 3: 無料トライアルのコンバージョン数

このセクションでは、過去 24 時間を対象に、無料トライアルの開始数、コンバージョン数、キャンセル数のデータが一覧表示されます。

列名	説明
製品タイトル	製品のタイトル。
製品コード	製品を表す一意の識別子。
新しい無料トライアル	レポート期間中に開始された新しい無料トライアルの数。
現在の無料トライアルの合計	アクティブな無料トライアルサブスクリプションの合計数。
コンバージョンされた無料トライアル	レポート期間中に無料トライアルから有料使用に移行されたサブスクリプションの合計数。
コンバージョンされなかった無料トライアル	無料トライアルの終了後に有料使用にコンバージョンされなかったサブスクリプションの合計数。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。

セクション 4: 新しいインスタンス

このセクションでは、過去 24 時間を対象に、新しい EC2 インスタンスとインスタンスタイプのデータが一覧表示されます。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。

列名	説明
ユーザーの都道府県	製品をサブスクライブしているアカウントに関連付けられている請求先住所の都道府県。
ユーザーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
製品タイトル	製品のタイトル。
製品コード	製品の一意的識別子。
タイプ	Amazon EC2 インスタンスタイプ
カウント	EC2 インスタンスの数。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
顧客の市区町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
ソリューションタイトル	ソリューションの名前。

列名	説明
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

セクション 5: 新規の製品サブスクライバー

このセクションでは、過去 24 時間を対象に、新しい購入者のデータが一覧表示されます。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
ユーザーの都道府県	製品をサブスクライブしているアカウントに関連付けられている請求先住所の都道府県。
ユーザーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
製品タイトル	製品のタイトル。
製品コード	製品の一意の識別子。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。

列名	説明
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
顧客の市区町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意的識別子。
支払者リファレンス ID	アカウント ID ではない一意的識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意的識別子。

セクション 6: キャンセルした製品サブスクライバー

このセクションでは、過去 24 時間を対象に、購入者のキャンセルに関するデータが一覧表示されません。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
ユーザーの都道府県	製品をサブスクライブしているアカウントに関連付けられている請求先住所の都道府県。
ユーザーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
製品タイトル	製品のタイトル。
製品コード	製品の一意的識別子。
サブスクライブ日	サブスクリプションが開始された日付。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
顧客の市区町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。

列名	説明
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

日次顧客サブスクライバーレポート

このレポートには、お客様の製品を購入した購入者のデータが一覧表示されます。このレポートでは、現在または過去の使用量は示されません。顧客が製品をサブスクライブしていることだけがわかります。該当する情報が使用可能な場合にのみ、このレポートを受け取ります。受け取るべきレポートが届かない場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

このレポートは、[AWS Marketplace 管理ポータル](#) でアクセスできます。[the section called “AWS Marketplace コマース分析サービス”](#) に登録している場合は、AWS SDK を使用してレポートにアクセスすることもできます。

このレポートには 2 つのセクションがあります。1 つは時間課金と月間サブスクリプション用、もう 1 つは年間サブスクリプション用です。レポートには、製品をサブスクライブしているすべての顧客の AWS アカウント IDs のリストが含まれます。

掲載スケジュール

このレポートは毎日 00:00 UTC に発行され、前日の 00:00 UTC から 23:59 UTC までを対象としています。

トピック

- [セクション 1: 時間単位と月単位のサブスクリプション](#)
- [セクション 2: 可変長のサブスクリプション](#)

セクション 1: 時間単位と月単位のサブスクリプション

このセクションでは、前日の 23:59:59 UTC の時点でのすべての従量制サブスクリプションのデータが一覧表示されます。

列名	説明
顧客 AWS アカウント番号	製品をサブスクライブしているアカウント。
製品タイトル	製品のタイトル。
製品 ID	ソフトウェア製品の一意的識別子。
製品コード	ソフトウェア製品の一意的識別子。
サブスクリプション開始日	サブスクリプションの開始日 (YYYY-MM-DD 形式)。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意的識別子。
支払者リファレンス ID	アカウント ID ではない一意的識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
リセラーアカウント ID	チャンネルパートナーリセラー用の一意的識別子。

列名	説明
リセラーアカウント名	チャンネルパートナーリセラーの名前。

セクション 2: 可変長のサブスクリプション

このセクションには、前日の 23:59:59 UTC の時点でのすべての定額制サブスクリプションのデータが一覧表示されます。

列名	説明
顧客 AWS アカウント番号	製品をサブスクライブしているアカウントの ID。
製品タイトル	製品のタイトル。
製品 ID	ソフトウェア製品の一意的識別子。
製品コード	ソフトウェア製品の一意的識別子。この情報は Amazon EC2 インスタンスのメタデータの一部としても使用できます。
サブスクリプション ID	サブスクリプションの ID。
サブスクリプション数	顧客が購入したライセンスの合計数。
サブスクリプションタイプ	サブスクリプションのタイプ。
サブスクリプションの目的	このオファーが以前のオファーのアップグレードや更新であるかどうか。
オファー ID	購入者が署名したオファーの識別子。
サブスクリプション開始日	顧客が製品をサブスクライブした日付 (YYYY-MM-DD 形式)。
以前のオファー ID	アップグレードや更新のオファーに先立つ以前のオファーの ID (存在する場合)。

列名	説明
オファターの可視性	オファターがパブリック、プライベート、エンタープライズ契約オファターのいずれになるか。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
リセラーアカウント ID	チャンネルパートナーリセラー用の一意の識別子。
リセラーアカウント名	チャンネルパートナーリセラーの名前。

支払いレポート

支払いレポートでは、前回の支払い以降に、販売者の銀行口座で回収された資金および支払われた資金に関する情報が提供されます。支払いには、販売者の製品のサブスクリプションに対する顧客の支払いまたは返金、顧客に対して徴収または返金される税金が含まれます。資金が顧客から回収されるまで、資金の支払いを受け取りません。では顧客ごとに支払い条件が異なるため AWS、未回収の各年齢カテゴリの資金の一部は顧客から支払われていない可能性があります。

返金は販売者による承認後に行われるため、マイナスの金額として表示されます。

このレポートは、レポートタブの AWS Marketplace 管理ポータルで使用できます。このような独自のレポートを作成するには、[をベースレポート例 1: 製品別の支払い](#)として使用し [AWS Marketplace 販売者配送データフィードサービス](#)、ニーズに合わせてカスタマイズできます。

掲載スケジュール

このレポートは、銀行振込の支払いを開始してから 3~5 日後に発行されます。一般的に、これは支払いを受け取る販売者の月次レポートです。支払いが開始されていない場合、支払いレポートは生成されません。

トピック

- [セクション 1: 製品別の支払い額](#)
- [セクション 2: 顧客地域別の支払い額](#)
- [セクション 3: インスタンス時間単位の支払い額](#)
- [セクション 4: 未回収資金の期間](#)
- [セクション 5: 回収済み資金の期間](#)
- [セクション 6: 支払期限を過ぎた資金の期間](#)
- [セクション 7: 未回収資金の内訳](#)

セクション 1: 製品別の支払い額

このセクションでは、製品別の支払いデータが一覧表示されます。

列名	説明
製品	製品のタイトル。
製品コード	製品の一意的識別子。
SellerRev	顧客に請求される製品の従量課金額または料金。
AWSRefFee	AWS Marketplace 料金の金額。
SellerRevRefund	データカバレッジ期間中に返金が処理された場合に顧客に払い戻されるサブスクリプション費用の金額。
AWSRefFeeRefund	データカバレッジ期間中に返金が処理された場合に返金される AWS Marketplace 料金の金額。
SellerRevCredit	がお客様のアカウントに AWS Marketplace 付与した AWS クレジット。
AWSRefFeeCredit	がアカウントに AWS Marketplace 付与した AWS クレジット。

列名	説明
正味金額	販売者に支払われた資金の合計。この列は、列から SellerRev AWSRefFee 列を引いた値に等しくなります。顧客に返金が行われると、この列は列から SellerRevRefund 列を引いた数と等しい負の数になります AWSRefFeeRefund。
取引リファレンス ID	AWS Marketplace レポート間でトランザクションを関連付けるのに役立つトランザクションの一意の識別子。
SellerUSSalesTax	この取引で請求された米国の消費税と使用税の合計額。
SellerUSSalesTaxRefund	返金が処理される場合にこの取引に対して払い戻される米国の消費税と使用税の合計金額。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
顧客の市区町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
ソリューションタイトル	ソリューションの名前。

列名	説明
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

セクション 2: 顧客地域別の支払い額

このセクションには、顧客の地域別の支払いデータが一覧表示されます。

列名	説明
決済 ID	支払いの一意の識別子。
決済期間開始日	支払い期間の開始日時。
決済期間終了日	支払い期間の終了日時。
入金日	支払いの発生日時。
支払い額	支払い合計金額
国コード	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
都道府県または地域	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
市町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。

列名	説明
郵便番号	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
納税地別正味金額	納税地別に販売者に支払われる資金の合計 (AWS Marketplace 料金、返金、米国の消費税と使用税を引いたもの)。
納税地別総額	納税地別に販売者に支払われる資金の合計。
販売者米国消費税	販売者の代理でこの取引で請求された米国の消費税と使用税の合計額。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。
販売者米国消費税返金	返金が処理された場合に、このトランザクションに対して返金される米国の売上税および使用税の合計額 (出品者に代わってそのような税金が徴収されたとき)。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。

セクション 3: インスタンス時間単位の支払い額

このセクションでは、Amazon EC2 インスタンス時間別に支払いデータが一覧表示されます。

列名	説明
製品	製品のタイトル。
製品コード	製品の一意的識別子。
使用タイプ説明	使用量の説明 (提供タイプ、リージョン、インスタンスタイプなど)。
Rate	この提供タイプ、リージョン、インスタンスタイプの 1 時間あたりのレート。

列名	説明
ユーザーカウント	この提供タイプ、リージョン、インスタンスタイプを使用する一意の顧客の数。
インスタンス時間	この提供タイプ、リージョン、インスタンスタイプの使用時間。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。

セクション 4: 未回収資金の期間

このセクションでは、未回収資金のデータが期間別に一覧表示されます。未回収資金には、期限を過ぎていない金額が含まれる場合があります。

列名	説明
未回収 (31 日未満保留中)	請求されたが 31 日未満に回収されなかった資金の合計。
未回収 (31 ~ 60 日間保留中)	請求されたが 31 ~ 60 日以内に回収されなかった資金の合計。
未回収 (61 ~ 90 日間保留中)	請求されたが 61 ~ 90 日以内に回収されなかった資金の合計。
未回収 (91 ~ 120 日間保留中)	請求されたが 91 ~ 120 日以内に回収されなかった資金の合計。
未回収 (120 日超保留中)	請求されたが 120 日を超えて回収されなかった資金の合計。
未回収 (全体)	請求されたが回収されなかったすべての資金の合計。

セクション 5: 回収済み資金の期間

このセクションでは、支払い以降に回収された資金のデータが一覧表示されます。

列名	説明
回収 (31 日未満保留中)	0~31 日の範囲内に請求されて回収された資金の合計。
回収 (31~60 日間保留中)	31~60 日の範囲に請求されて回収された資金の合計。
回収 (61~90 日間保留中)	61~90 日の範囲で請求されて回収された資金の合計。
回収 (91~120 日間保留中)	91~120 日の範囲で請求されて回収された資金の合計。
回収 (120 日超保留中)	120 日を超えて請求されて回収された資金の合計。
回収 (全体)	回収の資金の合計。

セクション 6: 支払期限を過ぎた資金の期間

このセクションでは、顧客によって支払い可能であるが、AWSとの契約に従って未払いになっている資金のデータが一覧表示されます。

列名	説明
支払期限経過 (31 日未満)	過去 0~31 日間に発生していて支払期限が到来しているが顧客が支払っていない資金の合計。
支払期限経過 (31~60 日)	過去 31~60 日間に発生していて支払期限が到来しているが顧客が支払っていない資金の合計。

列名	説明
支払期限経過 (61 ~ 90 日)	過去 61 ~ 90 日間に発生していて支払期限が到来しているが顧客が支払っていない資金の合計。
支払期限経過 (91 ~ 120 日)	過去 91 ~ 120 日間に発生していて支払期限が到来しているが顧客が支払っていない資金の合計。
支払期限経過 (120 日超)	過去 121 日以上に発生していて支払期限が到来しているが顧客が支払っていない資金の合計。
支払期限経過 (全体)	発生していて支払期限が到来しているが顧客が支払っていない資金の合計。

セクション 7: 未回収資金の内訳

このセクションには、すべての未回収資金が、支払期日順にソートされて一覧表示されます。

列名	説明
支払者 AWS アカウント番号	ソフトウェア料金の請求先となるアカウント。
製品コード	製品の一意的識別子。
総収益	製品の使用に対して請求された金額、または製品の使用に対する料金。
AWS 収益分配	決済時に請求額から控除される AWS 料金。
総返金	取引の返金の合計金額。
AWS 返金分配	取引で返金される AWS 料金の一部。
正味収益	この取引で請求された正味額から、AWS 料金、返金、および米国の消費税と使用税を差し引いた額。

列名	説明
通貨	取引の通貨。例えば、取引が US ドルの場合、エント리는 USD です。
AR 期間	取引の月と年 (YYYY-MM 形式)。
取引リファレンス ID	トランザクションを表す一意の識別子。これを使用して、AWS Marketplace レポート間でトランザクションを関連付けることができます。
オポチュニティの名前	登録された機会の一意の識別子。
オポチュニティの説明	登録されたオポチュニティの任意のメタデータ。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。
支払期日	YYYY-MM-DD 形式の支払期日。

月別請求済み収益レポート

月別請求済み収益レポートは、会計およびその他の財務レポートの目的で毎月の請求済み収益に関する正式な情報を提供します。このレポートには、が製品の時間単位、年単位、または月単位の使用に対して顧客に AWS 請求する合計金額が表示されます。レポートには 4 つのセクションとして、時間課金額と月額料金、任意期間のサブスクリプション数、フィールドデモ使用量、Flexible Payment 額があります。

⚠ Important

このレポートの金額には、顧客から回収した金額ではなく、顧客に請求した収益のみが反映されています。

このレポートは、レポートタブの AWS Marketplace 管理ポータル で使用できます。AWS Marketplace コマース分析サービスに登録している場合は、API コールを使用してこのレポートのプルダウンセクションを使用できます。詳細については、「[the section called “AWS Marketplace コマース分析サービス”](#)」を参照してください。

掲載スケジュール

このレポートは、毎月 15 日の 00:00 UTC に発行されます。

請求および収益データは翌月の 15 日から 14 日までを対象としています。

長期のサブスクリプションは、前月の初日 00:00 UTC から月末の日 23:59 UTC までを対象とします。

たとえば、5 月 15 日に発行されたレポートは、4 月 1 日の 00:00 UTC から 4 月 30 日の 23:59 UTC までを対象としています。

トピック

- [セクション 1: 請求および収益データ](#)
- [セクション 2: 可変長のサブスクリプション](#)
- [セクション 3: AWS フィールドデモンストレーションの使用](#)
- [セクション 4: Flexible Payment Schedule による契約](#)

セクション 1: 請求および収益データ

このセクションでは、従量課金額、返金、料金、徴収される米国の消費税と使用税のデータが一覧表示されます。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
状態	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
市町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
郵便番号	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
製品タイトル	製品のタイトル。
製品コード	製品の一意的識別子。
お客様の請求額	顧客に請求される製品の従量課金額または月額料金。
AWS 出品料金	請求金額から差し引かれる AWS Marketplace 料金額。
返金額	データカバレッジ期間中に返金が処理された場合に顧客に払い戻されるサブスクリプション費用の合計金額。
AWS 料金の返金	データカバレッジ期間中に返金が処理された場合に返金された AWS Marketplace 料金の一部。
コスト	リセラーに対する製品の費用 (リセラーが製品販売時に支払う金額など)。

列名	説明
パートナー収益額	取引に対して請求される合計金額、AWS Marketplace 手数料、返金、および米国の売上税と使用税を差し引いた金額。
通貨	取引の通貨。たとえば、取引が US ドルの場合、エント리는 USD です。
取引リファレンス ID	AWS Marketplace レポート間でトランザクションを関連付けるのに役立つトランザクションの一意の識別子。
米国消費税お客様の請求額	販売者の代理でこの取引で請求された米国の消費税と使用税の合計額。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。
米国消費税返金額	返金が処理された場合に、このトランザクションに対して返金される米国の売上税および使用税の合計額 (出品者に代わってそのような税金が徴収されたとき)。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが <code>liu-jie@example.com</code> の場合、エント리는 example.com です。
オポチュニティの名前	登録された機会の一意の識別子。

列名	説明
オポチュニティの説明	登録されたオポチュニティのメタデータ。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

セクション 2: 可変長のサブスクリプション

このセクションでは、定額制料金のデータが一覧表示されます。

列名	説明
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
状態	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
市町村	ソフトウェア料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。

列名	説明
製品タイトル	製品のタイトル。
製品コード	製品の一意の識別子。
サブスクリプション数	任意期間のサブスクリプション購入の一部として指定されたライセンスの合計数。
サブスクリプション開始日	任意期間のサブスクリプション購入の開始日。
サブスクリプション終了日	任意期間のサブスクリプション購入の終了日。
サブスクリプションインスタンスタイプ	任意期間のサブスクリプション購入に関連付けられているインスタンスタイプ。
お客様の請求額	従量課金額、月額料金、またはその両方。
AWS 出品料金	請求金額から差し引かれた AWS Marketplace 料金額。
返金額	データカバレッジ期間中に返金が処理された場合に顧客に払い戻される合計金額。
AWS 料金の返金	データカバレッジ期間中に返金が処理された場合に返金された AWS Marketplace 料金の一部。
コスト	リセラーに対する製品の費用 (リセラーが製品販売時に支払う金額など)。
パートナー収益額	この取引に対して請求される合計金額、AWS Marketplace 手数料、返金、および米国の売上税と使用税を差し引いた金額。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリーは USD です。

列名	説明
取引リファレンス ID	AWS Marketplace レポート間でトランザクションを関連付けるのに役立つトランザクションの一意の識別子。
米国消費税お客様の請求額	販売者の代理でこの取引で請求された米国の消費税と使用税の合計額。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。
米国消費税返金額	返金が処理された場合に、このトランザクションに対して返金される米国の売上税および使用税の合計額 (出品者に代わってそのような税金が徴収されたとき)。(つまり、米国の売上高および税レポートの関連レコードには「納税義務者」 == 「SELLER」と表示されます)。
顧客 AWS アカウント番号	料金の請求先となるアカウントの ID。
顧客の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリーは example.com です。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
契約開始日	AWS Marketplace SaaS 契約の開始日。
契約終了日	AWS Marketplace SaaS 契約の終了日。
オポチュニティの名前	登録された機会の一意の識別子。
オポチュニティの説明	登録されたオポチュニティのメタデータ。
ソリューションタイトル	ソリューションの名前。

列名	説明
ソリューション ID	ソリューションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

セクション 3: AWS フィールドデモンストレーションの使用

セクションには、製品の AWS [フィールドデモンストレーションの使用](#)に関するデータが一覧表示されます。潜在的な顧客に対して製品デモを実施できるように製品を設定できます。デモからの使用量はここに一覧表示されます。

列名	説明
製品タイトル	製品のタイトル。
製品コード	製品の一意の識別子。
インスタンスタイプ	フィールドデモに関連付けられている Amazon EC2 インスタンスタイプ。
使用量の単位	製品に関連付けられている使用量の単位数。
使用量の単位タイプ	使用量に関連付けられている単位 (時間など)。

セクション 4: Flexible Payment Schedule による契約

このセクションでは、前のレポート期間に Flexible Payment スケジュールを使用して作成したすべての契約のデータが一覧表示されます。

列名	説明
顧客 AWS アカウント番号	料金の請求先となる支払いアカウントの ID。
顧客の国	料金が請求先となる支払いアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
顧客の都道府県	料金の請求先となる支払いアカウントに関連付けられている請求先住所の都道府県。
顧客の市区町村	料金の請求先となる支払いアカウントに関連付けられている請求先住所の市区町村。
顧客の郵便番号	料金の請求先となる支払いアカウントに関連付けられている請求先住所の郵便番号。
顧客の E メールドメイン	料金の請求先となる支払いアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが <i>liu-jie@example.com</i> の場合、エントリは example.com です。
ユーザー参照 ID	料金の請求先となる支払いアカウントの口座。
AWS ユーザーアカウント番号	製品をサブスクライブしているアカウントの ID。
製品 ID	製品の一意的識別子。
製品タイトル	製品のタイトル。
製品タイプ	製品のタイプ。
AWS Marketplace オファー ID	購入者が署名したオファーの識別子。
契約の作成日付	契約作成日。アカウントがオファーをサブスクライブする日付です。
契約の有効期限	契約が期限切れになる日付。

列名	説明
総契約価値 (USD)	契約の総額 (USD)。
支払い回数	契約でスケジュールされている支払いの回数。
請求書日付	請求書が作成された日付。
請求書金額 (USD)	請求書で請求される金額 (USD)。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。

販売補償レポート

レポートには、月別の請求済み収益と、標準の [the section called “月別請求済み収益レポート”](#) にはない追加の顧客情報が一覧表示されます。レポートには、が製品の時間単位、年間、または月単位の使用に対して顧客に AWS 請求する合計金額が一覧表示されます。

Important

このレポートの金額には、顧客から回収された金額ではなく、顧客に請求される収益のみが反映されています。

販売補償レポートおよびこのプログラムの一部として共有される情報は、お客様との非開示契約に基づく Amazon の機密情報、またはそのような契約が存在しない場合は AWS Marketplace 販売者の利用規約を構成します。この情報を使用できるのは、会社名、地域、AWS アカウント ID で AWS Marketplace 収益を担当者とマッピングすることで、販売担当者を補償するためだけです。この情報は、支払うべき手数料の源泉を理解するために知る必要がある従業員と共有できます。そのような情報の使用と共有は、お客様との機密保持契約における義務、および AWS Marketplace 販売者向けの利用規約の第 3.8 条を含む AWS Marketplace 販売者の利用規約に準拠する必要があります。

このような独自のレポートを作成するには、をベースレポート [例 2: 販売補償レポート](#) として使用し AWS Marketplace [販売者配送データフィードサービス](#)、ニーズに合わせてカスタマイズできます。

掲載スケジュール

このレポートは、毎月 15 日の 00:00 UTC に発行されます。また、前の暦月の初日 00:00 UTC から末日 23:59 UTC までを対象とします。たとえば、5 月 15 日に発行されたレポートは、4 月 1 日の 00:00 UTC から 4 月 30 日の 23:59 UTC までを対象としています。

請求済み収益

このレポートの請求済み収益セクションには、前の暦月からの従量制および定額制料金が含まれます。以下に示しているのは列名と説明です。

Note

このレポートでは、出品料金は、階層型出品料金に従って決定される、取引収益の割合です (認定済みリセラーによる認定済み転売製品の転売の除く)。

列名	説明
顧客 AWS アカウント番号	料金の請求先となるアカウント。
国	料金の請求先となるアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
状態	料金の請求先となるアカウントに関連付けられている請求先住所の都道府県。
市町村	料金の請求先となるアカウントに関連付けられている請求先住所の市区町村。
郵便番号	料金の請求先となるアカウントに関連付けられている請求先住所の郵便番号。
E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。たとえば、E メールアドレスが liu-jie@example.com の場合、エントリーは example.com です。

列名	説明
製品コード	製品の一意的識別子。
製品タイトル	製品のタイトル。
総収益	製品の使用に対して請求された金額、または製品の使用に対する月額料金。
AWS 収益共有	決済時に請求金額から差し引かれた AWS 料金額。 the section called “支払いレポート” に表示されます。
総返金	取引の返金の合計金額。
AWS 返金共有	取引に対して返金される AWS 料金の一部。
正味収益	この取引に対して請求される正味金額から、AWS 手数料、返金、米国の売上税と使用税を引いた金額。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリーは USD です。
AR 期間	取引の月と年 (YYYY-MM 形式)。
取引リファレンス ID	トランザクションを表す一意の識別子。これを使用して、AWS Marketplace レポート間でトランザクションを関連付けることができます。
オポチュニティの名前	登録された機会の一意の識別子。
オポチュニティの説明	登録されたオポチュニティの任意のメタデータ。
ソリューションタイトル	ソリューションの名前。
ソリューション ID	ソリューションの一意的識別子。

列名	説明
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。
支払者住所 ID	お客様の住所を表す一意の識別子。

米国の消費税と使用税レポート

この月次レポートは、で販売および使用取引から AWS 収集する米国の売上に関する情報を販売者に提供します AWS Marketplace。販売者の米国の売上税は収集または計算 AWS されません。

AWS が内部税務上の決定に基づいて徴収し、AWS として税務当局に送金した金額が表示されます。詳細については、[「AWS Marketplace Sellers and Tax Collection on AWS Marketplace Tax Help for Sellers」](#)を参照してください。

支払いレポートとこのレポートとの間で取引をマッピングするには、Transaction Reference ID を使用します。

このレポートは、レポートタブの AWS Marketplace 管理ポータルで使用できます。AWS Marketplace Commerce Analytics サービスに登録している場合は、API コールを使用してこのレポートのプルダウンセクションを使用できます。詳細については、[「the section called “AWS Marketplace コマース分析サービス”](#)」を参照してください。

掲載スケジュール

このレポートは、毎月 15 日の 00:00 UTC に発行されます。また、前の暦月の初日 00:00 UTC から末日 23:59 UTC までを対象とします。たとえば、5 月 15 日に発行されたレポートは、4 月 1 日の 00:00 UTC から 4 月 30 日の 23:59 UTC までを対象としています。

米国の消費税と使用税レコード

このセクションでは、ソフトウェア料金に起因する米国の消費税額のデータが一覧表示されます。

列名	説明
明細項目/ID	明細項目の一意的識別子。返金取引の明細項目 ID は先渡し取引税のものと同じです。
顧客請求書 ID	顧客請求書の一意的識別子。
[Product Name] (製品名)	購入された製品の名前。
製品コード	製品の一意的識別子。
製品税コード	製品の税プロパティを識別するための標準コード。製品を作成または変更するときにプロパティを選択します。
販売者 ID	取引の登録販売者の一意的識別子。
販売者名	販売者の正式名称。
取引日	取引の日付。
合計調整価格	取引の最終価格。
合計税額	取引に課される合計税額。
基本通貨コード	すべての AWS Marketplace トランザクションの基本通貨コード。このエントリは常に USD です。
請求先の市区町村	ソフトウェア料金の請求先となる支払いアカウントに関連付けられている請求先住所の市区町村。
請求先の都道府県	ソフトウェア料金の請求先となる支払いアカウントに関連付けられている請求先住所の郵便番号。
請求先の郵便番号	ソフトウェア料金の請求先となる支払いアカウントに関連付けられている請求先住所の郵便番号。

列名	説明
請求先の国	ソフトウェア料金が請求先となる支払いアカウントに関連付けられている 2 文字の国コード。このレポートでは、ISO 3166-1 alpha-2 標準を使用します。
取引タイプコード	取引のタイプコード。有効値: <ul style="list-style-type: none"> • AWS: 先渡し取引税 • REFUND: 全額または一部返金 • TAXONLYREFUND : 税のみの返金 返金取引の明細項目 ID は元の先渡し取引のものと同じです
表示価格課税タイプ	顧客に表示される価格の課税タイプ。すべての AWS Marketplace サービスは排他的です。
納税地コード課税管轄	課税場所に関連付けられている頂点ジオコード。
税タイプコード	取引に適用される税のタイプ。指定できる値は、None、Sales および SellerUse です。
管轄区域	納税地に使用される住所の管轄区域。指定できる値は、State、County、City、District です。
課税管轄	課税管轄地域の名前。
課税売上金額	管轄区域別に課税対象となる取引金額。
非課税売上金額	管轄区域別に非課税となる取引金額。
課税額	管轄区域で課される税額。

列名	説明
税管轄税率	管轄区域で適用される税率。
税計算理由コード	取引が課税対象となるか、非課税となるか、免除とされるか、ゼロ評価済みであるかを管轄区域別に整理したもの。
税計算に使用された日付	取引の税金の計算に使用される日付。
顧客の免除証明書 ID	免除証明書の証明書 ID。
顧客の免除証明書 ID ドメイン	証明書が Amazon システムに保存されている場所。
顧客の免除証明書レベル	免除を提供した管轄区域。
顧客の免除コード	免除を指定するコード (RESALE など)。
顧客の免除ドメイン	顧客の免除情報 (ある場合) を取得するために使用される Amazon システム。
カスタマーリファレンス ID	アカウント ID ではない一意の識別子。顧客別の使用量、収益、サブスクリプションの追跡に使用されます。
取引リファレンス ID	AWS Marketplace レポート間でトランザクションを関連付けるのに役立つトランザクションの一意の識別子。
支払者リファレンス ID	アカウント ID ではない一意の識別子。料金が請求されるアカウントと関連付けられています。これは、すべての AWS Marketplace 財務レポートの顧客による使用状況、収益、サブスクリプションの追跡に役立ちます。

列名	説明
納税義務者	このフィールドには、Seller または AWS が入力されます。販売者が納税義務者である場合、販売者は税務上の決定に基づいて、自らの徴収義務と送金義務を負います。AWS が納税義務者である場合、売上税は AWS によって徴収され税務当局に送金されます。詳細については、アマゾン ウェブ サービス 税務ヘルプの「 AWS Marketplace Sellers & Tax Collection 」を参照してください。

補足レポート

AWS Marketplace は [販売者配送データフィードサービス](#)、を通じて、販売者の AWS Marketplace 出品に関連する AWS Marketplace 販売者アカウント ID に接続されている販売者所有の Amazon S3 アカウントに補足レポートを提供します。詳細については、「[送信先の Amazon S3 バケットを作成する](#)」を参照してください。

補足レポートは、前日に新しいサブスクライバーがいた場合、毎日 16:00 UTC に発行されます。これらのレポートは、前日の 13:59 UTC から翌日の 16:01 UTC までの期間を対象としています。

契約詳細レポート

契約詳細レポートは、Software as a Service (SaaS) の無償トライアルを利用している顧客をサポートするのに役立ちます。レポートには、サブスクライバー名、サブスクライバー ID、オファー ID、契約開始、契約終了日などの契約の詳細が含まれます。

該当する情報が使用可能な場合にのみ、このレポートを受け取ります。受け取るべきと思われるレポートが届かない場合は、[AWS Marketplace Seller Operations](#) チームまでお問い合わせください。

このレポートには、AWS Marketplace Seller Account ID に関連付けられた Amazon S3 バケットからアクセスできます。

次の表は、契約詳細レポートの列名と説明をまとめたものです。

SaaS 契約無料トライアルレポートデータ

名前	説明
vendor_display_name	製品を販売したベンダーの名前。
vendor_aws_account_id	製品を販売したベンダーの ID。
subscriber_aws_account_id	製品にサブスクライブ AWS アカウント されている に関連付けられた ID。
customer_id	ソフトウェア製品の一意的識別子。
product_title	製品のタイトル。
offer_id	購入者が署名したオファーの識別子。
offer_visibility	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるかの表示。
reseller_name	チャンネルパートナーリセラーの名前。
reseller_aws_account_id	チャンネルパートナーリセラー用の一意的識別子。
agreement_id	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意的契約データフィードリファレンス。
agreement_acceptance_date	契約が受諾された日付。
agreement_start_date	契約の開始日。
agreement_end_date	契約の終了日。従量制/従量課金制/サブスクリプションの場合は、9999 年 1 月 1 日に設定されます。
is_free_trial_offer	オファーまたは契約が無料トライアルオファーかどうかを示すフラグ。

名前	説明
is_upgraded_after_free_trial	契約が有料契約にアップグレードされたかどうかを示すフラグ。
total_contract_value	契約の総額。

販売者ダッシュボード

AWS Marketplace は、財務データと売上データへのアクセスと分析に役立つグラフ、グラフ、インサイトを [Amazon QuickSight](#) が提供するダッシュボードに提供します。販売者ダッシュボードには以下が含まれます。

[the section called “財務業務用ダッシュボード”](#)

- [the section called “請求済み収益ダッシュボード”](#) - 会計やその他の財務報告の目的で請求された収益に関する情報を提供します。
- [the section called “回収と支払いダッシュボード”](#) - 前回の支払い以降に銀行口座に AWS 回収および支払われた資金に関する情報を提供します。
- [the section called “税金ダッシュボード”](#) - 販売者の取引の税金に関する情報を提供します。

[the section called “販売業務用ダッシュボード”](#)

- [the section called “契約および更新ダッシュボード”](#) - AWS Marketplaceで契約に署名してから 24 時間以内に、契約と更新に関する情報が表示されます。
- [the section called “使用状況ダッシュボード”](#) - SaaS およびサーバー使用量ベースの製品を使用しているお客様に、視覚化ときめ細かなデータを提供します。

ダッシュボードは、適切なアクセス許可を持つ AWS Marketplace 販売者が利用できます。

ダッシュボードへのアクセス

デフォルトでは、販売者アカウントの AWS Marketplace システム管理者は、の Insights タブにあるすべてのダッシュボードにアクセスできます AWS Marketplace 管理ポータル。システム管理者は AWS Identity and Access Management (IAM) ポリシーを作成して、販売会社の他のユーザーに特定のダッシュボードへのアクセスを提供できます。

Note

2023年9月には、従来の IAM 権限で有効になっていた出品者ダッシュボードへのアクセスはサポートされなくなります。以下のコード例の新しい Amazon リソースネーム (ARN) 形式を使用して IAM 許可を更新してください。

ポリシーの作成の詳細については、「[IAM ポリシーの作成](#)」を参照してください。

ダッシュボードポリシー

請求収益ダッシュボードおよび回収と支払いのダッシュボードへのアクセスを提供するには、次のいずれかのポリシーを使用します。

次のコード例を使用して、現在および将来のデータフィードに基づいて、現在および将来の AWS Marketplace リソース (ダッシュボードとレポートを含む) へのアクセスを提供できます。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetSellerDashboard"
    ],
    "Resource": [
      "arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/*",
    ]
  }]
}
```

または、次のコード例に示すように、特定の ARN セクションを含めて、1 つ以上のダッシュボードへのアクセスを提供できます。例えば、請求収益ダッシュボード、契約と更新のダッシュボード、および使用状況ダッシュボードのみにアクセスできるようにするには、arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/ReportingData/BillingEvent_V1/Dashboard/CollectionsAndDisbursements_V1 のコード例からこの行を削除します。

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

```
    "aws-marketplace:GetSellerDashboard"
  ],
  "Resource": [
    "arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/ReportingData/BillingEvent_V1/Dashboard/BilledRevenue_V1",
    "arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/ReportingData/BillingEvent_V1/Dashboard/CollectionsAndDisbursements_V1",
    "arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/ReportingData/Agreement_V1/Dashboard/AgreementsAndRenewals_V1",
    "arn:aws:aws-marketplace::<awsAccountID>:AWSMarketplace/ReportingData/Usage_V1/Dashboard/Usage_V1",
    "arn:aws:aws-marketplace::123456789012:AWSMarketplace/ReportingData/TaxItem_V1/Dashboard/Tax_V1"
  ]
}]
}
```

Note

AWS Identity and Access Management (IAM) ポリシーの作成の詳細については、[「ユーザーガイド」の「IAM ポリシーの作成AWS Identity and Access Management」](#)を参照してください。

財務業務用ダッシュボード

AWS Marketplace には、財務データの追跡に役立つ複数のダッシュボードが用意されています。

トピック

- [請求済み収益ダッシュボード](#)
- [回収と支払いダッシュボード](#)
- [税金ダッシュボード](#)

請求済み収益ダッシュボード

請求済み収益ダッシュボードには、AWS Marketplaceのすべての請求済み売上に関する情報が表示されます。請求済み収益ダッシュボードを用いてレポートを作成すると時間の節約となり、[従来の月次請求収益レポート](#)よりも約 45 日早くレポートを入手できます。従来の月次請求収益レポートは毎月 15 日に配信されるため、前月の請求額の確認が遅れます。

Amazon QuickSight ダッシュボードの任意のチャート、グラフ、およびテーブルから (.csv または Microsoft Excel ファイルとして) データをエクスポートおよびダウンロードできます。詳細については、「Amazon QuickSight ユーザーガイド」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

運用プロセスと財務プロセスについては、以下のトピックをご確認ください。

トピック

- [請求済み収益ダッシュボードの更新頻度](#)
- [セクション 1: コントロール](#)
- [セクション 2: 請求書の日付範囲](#)
- [セクション 3: メトリクス](#)
- [セクション 4: トレンド](#)
- [セクション 5: 内訳](#)
- [セクション 6: 詳細なデータ](#)

請求済み収益ダッシュボードの更新頻度

請求済み収益ダッシュボードは毎日午後 4 時 (UTC 時間の午前 0 時) に更新されます。5 月 10 日午後 4 時 (UTC 時間の午前 0 時) 以前に請求書が作成された場合、5 月 11 日の更新で請求書が表示されます。5 月 10 日午後 4 時 (UTC 時間の午前 0 時) 以降 (UTC) に請求書が作成された場合、5 月 12 日の更新で請求書が表示されます。アップストリームシステムから受信した最新の請求データまたは支払いデータが遅延した場合、最新のデータがダッシュボードに反映されるまでに 1~2 日かかる場合があります。

予想される顧客請求書が表示されない場合は、AWS Marketplace ビジネス開発担当者に連絡する前に、以下の手順に従います。

請求済み収益ダッシュボードの顧客請求書が見つからない場合のトラブルシューティング方法

1. まず次の 1 つ以上のツールを使用して、顧客がオファーを受け入れたことを確認します。
 - [日次顧客サブスクライバーレポート](#)
 - [AWS Marketplace イベントの通知](#)
 - [AWS Marketplace 管理ポータル 契約タブ](#)
2. (プライベートオファーの場合) 顧客がオファーを受け入れたことを確認したら、オファーの請求書のスケジュールと金額を確認します。

- プライベートオファーについては、AWS Marketplace 管理ポータル オフータブを確認してください。
 - チャネルパートナーのプライベートオファーについては、AWS Marketplace 管理ポータル パートナータブをチェックして、パートナーオポチュニティ内のカスタム支払いスケジュールを表示します。
3. 取引において、前月の使用分を毎月 2 日と 3 日に顧客に請求する [使用量料金モデル](#)が指定されているかどうかを確認してください。

セクション 1: コントロール

請求済み収益ダッシュボードの本セクション内には、同ダッシュボードのデータを絞り込むためのフィルターがあります。例えば、[AWS Marketplace イベントの通知](#)からフィールドのフィルターを選択して、特定の顧客アカウント ID、サブスクリバの会社名、またはオファー ID の請求を確認できます。また、ビジュアルに含める日付範囲などを設定できるフィルターを分析に追加することも可能です。セクション内で選択済みのフィルターにより、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。

フィルタリングの詳細については、[「Amazon ユーザーガイド」の「Amazon でのデータのフィルタリング QuickSight」](#)を参照してください。 QuickSight

コントロールの記述

コントロール名	説明
サブスクリバ AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクリバの会社名	製品をサブスクライブしているアカウント名。
製品のタイトル	製品のタイトル。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。

コントロール名	説明
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意的な契約データフィードリファレンス。
AWS 登録販売者	取引を促進した事業者の識別子。可能な値は以下のとおりです： <ul style="list-style-type: none"> • AWS_INC: AWS, Inc. (米国に拠点を置く) の識別子。 • AWS_EUROPE: Amazon Web Services EMEA SARL (ルクセンブルク拠点) の識別子。 • AWS_AUSTRALIA: Amazon Web Services オーストラリア株式会社の識別子 • AWS_JAPAN: Amazon Web Services 日本合同会社の識別子
支払者 AWS アカウント ID	料金の請求先となるアカウントの ID。
支払人の会社名	料金の請求先となるアカウントの商号。
リセラー会社	ソフトウェア製造元の製品の販売を許可されているリセラーアカウントの商号。
リセラー AWS アカウント ID	ISV から卸売で製品またはサービスを購入して顧客に再販したアカウントの ID。
再販承認 ID	登録された機会の一意的な識別子。
再販承認名	登録された機会の一意的な名前。
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある州または地域

コントロール名	説明
取引リファレンス ID	AWS Marketplace レガシーレポート間でトランザクションを関連付けるのに役立つトランザクションの一意的識別子。

セクション 2: 請求書の日付範囲

ダッシュボードの本セクションには、請求日が指定した日付の前か後か、または日付範囲内かどうかに基づいて、レコードを絞り込むためのフィルターがあります。請求書の日付範囲内で選択済みのフィルターにより、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。デフォルトの請求日付範囲は、過去 6 か月のデータを取り込むように設定されています。

セクション 3: メトリクス

ダッシュボードの本セクションには、主要業績評価メトリクス (KPI) が表示され、主要な請求済み収益を視覚的に比較できます。指定した請求日付範囲における総収益、総返金額、出品料金、卸売コスト (該当する場合)、販売者純収益の KPI が表示されます。[請求日付範囲] フィールドの日付条件を更新して、日付範囲を更新できます。

セクション 4: トレンド

ダッシュボードの本セクションには、指定した日付範囲の請求済み収益のトレンドが表示されます。日、、、などの指定された日付集計で傾向を表示 year-over-yearして month-over-month quarter-over-quarter、請求済み収益に関するインサイトを得ることができます。請求済み収益のトレンド情報は、集計レベル、またはオファーの表示タイプ別に確認できます。

- 請求トレンド - 日付集計フィルターから選択した日付範囲における総収益、販売者純収益、卸売コスト (該当する場合) 返金額の推移のスナップショットを表示します。
- オファーの可視性別の請求トレンド - [プライベートオファー](#)、パブリックオファー (またはセルフサービス)、エンタープライズプログラムにおけるオファー数と総収益の推移のスナップショットを、オファーの可視性タイプ別に表示します。

セクション 5: 内訳

ダッシュボードの本セクションには、サブスクライバー、レコードの販売者、サブスクライバーの地域、製品のタイトルにわたるビジネスに関する主要なメトリクスが表示されます。総収益、支払人

数、サブスクライバー数、総返金額、出品手数料、販売者純収益、卸売コストでフィルタリングできます。

セクション 6: 詳細なデータ

ダッシュボードのこのセクションには、製品の使用単位、年単位、または月単位の使用に対して顧客に AWS 請求する合計金額を含む、すべての請求済み売上が表示されます。は、次の 3 つの頻度で顧客に AWS 請求します。

- サブスクリプションの承認日 (前払い請求)
- カスタム支払いスケジュール (柔軟な支払いスケジュールによって作成されたプライベートオファー)
- 前月の使用量に対する当月の 2 日目と 3 日目の従量制使用量

Note

2021 年 4 月 1 日より前に作成された請求書には、契約 ID、オファー ID、サブスクライバー AWS アカウント ID、サブスクライバーの会社名が関連付けられていない場合があります。

顧客への請求は、顧客に請求されてから 24 時間後に、本詳細データセクションに表示されます。例えば、11 月 3 日に顧客に請求が行われた場合、ダッシュボードには 11 月 4 日に同請求書が表示されます。QuickSight テーブルからデータをエクスポートおよびダウンロードする方法の詳細については、「Amazon QuickSight [ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

詳細なデータの説明

列	[Description] (説明)
請求書日付	製品サブスクリプションの料金を顧客に請求した日付。
支払期日	YYYY-MM-DD 形式の支払期日。
支払い条件	顧客の AWS 請求書の支払い条件。

列	[Description] (説明)
請求書 ID	顧客に請求される料金に割り当てられた AWS ID。
出品料金の請求書 ID	AWS Marketplace サブスクリプションが AWS EMEA SARL、日本、またはオーストラリアの法人 (登録販売者) を通じて取引される場合、販売者は販売者出品料金に付加価値税 (VAT) を請求し、税準拠の請求書を渡す必要があります。該当する取引において出品料金に課される VAT の請求書 ID は、ソフトウェアまたは製品のサブスクリプション請求書 ID とは異なります。AWS, Inc. からのトランザクションの値は「MISSING_LISTING_FEE_Invoice_ID」となります。これは、出品料金の請求書 ID が適用されないためです。
サブスクライバーの会社名	製品をサブスクライブしているアカウント名。
サブスクライバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクライバーの E メールドメイン	製品をサブスクライブしているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
サブスクライバーの市町区村	製品をサブスクライブしているアカウントに関連付けられた請求先住所の市町区村。
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある都道府県
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある国
サブスクライバーの郵便番号	製品をサブスクライブしているアカウントに関連付けられた請求先住所の郵便番号。

列	[Description] (説明)
製品のタイトル	製品のタイトル。
オファー名	販売者が定義したオファー名。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意の契約データフィードリファレンス。
契約開始日	顧客の製品サブスクリプションの開始日 (MM-DD-YYYY) 形式。将来の日付の契約の場合、この日付は受理日とは異なる場合があります。
契約受理日	顧客が製品をサブスクライブした日付 (MM-DD-YYYY 形式)。
契約終了日	契約の有効期限が終了する日付 (MM-DD-YY YY 形式)。従量制/pay-as-you-go サブスクリプションの場合、この日付は JAN-1-9999 に設定されます。
使用期間終了日	製品の使用期間終了日。
使用期間開始日	製品の使用期間開始日。
支払いステータス	前回の支払い以降、AWS が資金を集めてお客様の銀行口座に支払いを行ったことを確認するための請求書に関連付けられたステータス。関連する請求書の支払い資金が回収され、支払いが完了しました。関連する未払請求書の資金は、回収も支払いも行われていません。

列	[Description] (説明)
支払日	販売者の銀行への支払い AWS が開始された日付。
銀行への支払いのトレース ID	銀行への支払いの場合、トレース ID は銀行によって割り当てられます。支払い銀行トレース ID は、販売者の銀行提供の預金通知とレポートを AWS Marketplace レポートの請求書に関連付けるために使用できます。
総収益	顧客に請求される製品の従量課金額または月額料金。
総返金	データカバレッジ期間中に返金が処理された場合に顧客に払い戻されるサブスクリプション費用の合計金額。
出品料金	請求金額から差し引かれる AWS Marketplace 料金額。
出品料の払い戻し	データカバレッジ期間中に返金が処理された場合に返金された AWS Marketplace 料金の一部。
出品料金の割合	請求金額から差し引かれる AWS Marketplace 料金の割合。
販売者のタックスシェア	この取引で請求された米国の消費税と使用税の合計額
販売者のタックスシェアの払い戻し	返金が処理される場合にこの取引に対して払い戻される米国の消費税と使用税の合計金額。
AWS のタックスシェア	販売者の代理でこの取引で請求された米国の消費税と使用税の合計額。

列	[Description] (説明)
AWS のタックスシェアの払い戻し	返金が処理された場合に、このトランザクションに対して返金される米国の売上税および使用税の合計額 (出品者に代わってそのような税金が徴収されたとき)。
卸売コスト	チャンネルパートナーのプライベートオファーマのみ。リセラーに請求される製品のコスト。例えば、リセラーが製造元の製品を販売したときに製造元に支払う金額などです。卸売コストは、希望小売り価格に割引率を掛けたものです。
卸売コストの払い戻し	チャンネルパートナーのプライベートオファーマのみ。リセラーから返金された製品のコスト。
販売者の純収益	取引に対して請求される合計金額、AWS Marketplace 手数料、返金、および米国の売上税と使用税を差し引いた金額。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリーは USD です。
取引リファレンス ID	トランザクションを表す一意の識別子。これを使用して、AWS Marketplace レポート間でトランザクションを関連付けることができます。
AWS 登録販売者	<p>取引を促進した事業体の識別子。可能な値は以下のとおりです。</p> <ul style="list-style-type: none"> • AWS_INC: AWS, Inc. (米国拠点) の識別子 • AWS_EUROPE: Amazon Web Services EMEA SARL (ルクセンブルク拠点) の識別子 • AWS_AUSTRALIA: Amazon Web Services オーストラリア株式会社の識別子 • AWS_JAPAN: Amazon Web Services 日本合同会社の識別子

列	[Description] (説明)
再販承認 ID	登録された機会の一意の識別子。
再販承認名	登録された機会の一意の名前。
再販承認の説明	ISV の定義による登録済み機会の説明。
再販会社名	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウント名。
支払者 AWS アカウント ID	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウントの ID。
支払人の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。例えば、E メールアドレスが liu-jie@example.com の場合、エントリーは example.com です。
支払人の市町区村	ソフトウェア料金の請求先となるアカウントに関連付けられた請求先住所の市区町村。
支払人の都道府県または地域	料金の請求先となるアカウントに関連付けられた請求先住所の都道府県。
支払人の国	料金の請求先となるアカウントに関連付けられた 2 文字の国コード。
支払人の郵便番号	ソフトウェア料金の請求先となるアカウントに関連付けられた請求先住所の郵便番号。
ISV アカウント ID	製品またはサービスの所有者の識別子。
ISV の会社名	製品またはサービスの所有者の商号。
製品 ID	ソフトウェア製品のわかりやすい一意の識別子。

回収と支払いダッシュボード

回収と支払いダッシュボードには、前回の支払い以降に AWS 回収され、銀行口座に支払われた資金に関するデータが表示されます。また、未払いの請求書のすべてのリストも表示されます。

支払いには、販売者の製品のサブスクリプションに対する顧客の支払いまたは返金、顧客に対して徴収または返金される税金が含まれます。請求書に記載されている全額が顧客から回収されるまで、資金の支払いは受け取れません。返金は販売者による承認後に行われるため、ダッシュボード上の返金はマイナスの金額として表示されます。

回収と支払いダッシュボードを使用すると、顧客の支払いにタイムリーにアクセスできるようになります。支払いが送信されてから 5 日後に作成される従来の[支払いレポート](#)と比較して、約 4 日間の節約が見込まれます。

Note

お客様は異なる支払い条件を持っているため AWS、未回収の年齢カテゴリの一部の資金は、顧客から支払われていない可能性があります。

回収と支払いダッシュボードには、運用プロセスと財務プロセスに関する情報が表示されます。ダッシュボードは毎日更新されます。詳細については、以下のトピックを参照してください。

トピック

- [回収と支払いダッシュボードの更新頻度](#)
- [セクション 1: コントロール](#)
- [セクション 2: 日付カテゴリの選択](#)
- [セクション 3: メトリクス](#)
- [セクション 4: トレンド](#)
- [セクション 5: 内訳](#)
- [セクション 6: 詳細なデータ](#)

回収と支払いダッシュボードの更新頻度

回収と支払いダッシュボードは、北米の営業日にのみ更新されます。支払い済みの請求書は、銀行への入金を受け取ってから 1 日以内に確認できます。

セクション 1: コントロール

ダッシュボードの本セクション内には、ダッシュボードのデータを絞り込むためのフィルターがあります。例えば、[AWS Marketplace イベントの通知](#)からフィールドのフィルターを選択して、特定の顧客アカウント ID、サブスクリイバーの会社名、またはオファー ID の支払いを確認できます。また、支払いステータス別にフィルタリングして、自分に支払われたすべての請求書、または未払いの請求書を確認することもできます。ビジュアルに含める日付範囲などを設定できるフィルターを分析に追加することが可能です。セクション内で選択済みのフィルターにより、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。

フィルタリングの詳細については、[「Amazon ユーザーガイド」の「Amazon でのデータのフィルタリング QuickSight」](#)を参照してください。 QuickSight

コントロールの記述

コントロール名	説明
サブスクリイバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクリイバーの会社名	製品をサブスクライブしているアカウント名。
製品のタイトル	製品のタイトル。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意的な契約データフィードリファレンス。
AWS 登録販売者	取引を促進した事業者の識別子。可能な値は以下のとおりです: <ul style="list-style-type: none"> AWS_INC: AWS, Inc. (米国に拠点を置く) の識別子。

コントロール名	説明
	<ul style="list-style-type: none"> • AWS_EUROPE: Amazon Web Services EMEA SARL (ルクセンブルク拠点) の識別子。 • AWS_AUSTRALIA: Amazon Web Services Australia Pty Ltd (AWS オーストラリア) の識別子 • AWS_JAPAN: Amazon Web Services 日本合同会社の識別子
支払いステータス	<p>が前回の支払い以降に資金を AWS 回収して銀行口座に支払ったことを確認する請求書に関連付けられたステータス。関連する請求書の支払い資金が回収され、支払いが完了しました。関連する未払請求書の資金は、回収も支払いも行われていません。</p>
支払者 AWS アカウント ID	料金の請求先となるアカウントの ID。
支払人の会社名	料金の請求先となるアカウントの商号。
リセラー会社	ソフトウェア製造元の製品の販売を許可されているリセラーアカウントの商号。
リセラー AWS アカウント ID	ISV から卸売で製品またはサービスを購入して顧客に再販したアカウントの ID。
再販承認 ID	登録された機会の一意の識別子。
再販承認名	登録された機会の一意の名前。
サブスクリバターの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。
サブスクリバターの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある州または地域

コントロール名	説明
取引リファレンス ID	AWS Marketplace レガシーレポート間でトランザクションを関連付けるのに役立つトランザクションの一意の識別子。
銀行への支払いのトレース ID	銀行への支払いの場合、トレース ID は銀行によって割り当てられます。銀行のトレース ID では、販売者の銀行が提供する入金通知や AWS Marketplace レポートをレポート内の請求書と関連付けることができます。

セクション 2: 日付カテゴリの選択

ダッシュボードの本セクションには、2 つの異なる日付ディメンション、および日付フィールド値が指定した日付の前か後か、または日付範囲内かどうかに基づいて、レコードを絞り込むためのフィルターがあります。日付ディメンションは支払い期日または最終支払い日です。日付カテゴリフィルターでは、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。デフォルトの日付カテゴリは最終支払い日で、過去 6 か月のデータを取得します。

セクション 3: メトリクス

ダッシュボードの本セクションには、主要業績評価メトリクス (KPI) が表示され、支払い済みおよび未支払いの収益を視覚的に比較できます。指定した日付カテゴリと日付範囲における総収益、純収益、卸売コスト (該当する場合)、支払い額、未支払い額の KPI が表示されます。

セクション 4: トレンド

ダッシュボードのこのセクションには、指定した日付範囲における支払いと延滞のトレンドが表示されます。日、月、四半期、年など、指定した日付集計で傾向を表示して、AWS Marketplace コレクションの健全性を把握できます。トレンドビューには、次のようなものが含まれます。

- 支払いトレンド - 支払いまでの平均日数とそれに関連する純収益のスナップショットが表示されます。このトレンドでは、請求日から支払い日までの日数を測定して、回収効率をレポートします。日付集計フィルターから日付範囲を選択できます。
- 支払い済み支払いの期間 - 標準期間の売掛金バケット (期限前、1~30 日、31~60 日など) ごとに分類された、純収益と支払い済み請求書の数のスナップショットが表示されます。トレンドでは、支払い期日と支払い日の間の日数を測定して、支払いが顧客の支払い条件の範囲内であったかどうかを報告します。

- 未払い支払いの期間 - 期日経過バケット (期限前、1~30 日、31~60 日など) ごとに分類された、純収益、未支払いの請求の数のスナップショットが表示されます。未払い資金には、期限を過ぎていない金額が含まれる場合があります。トレンドでは、今日から支払期日までの日数を測定して、受取売掛金を表示します。

セクション 5: 内訳

ダッシュボードのこのセクションには、オファー ID、製品のタイトル、支払者の会社名、サブスクライバーの会社名、リセラー名 (チャンネルパートナーのプライベートオファーに参加している場合)、支払人の地域、サブスクライバーの地域ごとに、売掛金が表示されます。この内訳を使用して、各カテゴリの支払い済み売掛金と未払いの売掛金を比較します。

セクション 6: 詳細なデータ

ダッシュボードのこのセクションには、製品、顧客、オファーの詳細別に、すべての支払いと未回収額が表示されます。

Note

2021 年 4 月 1 日より前に作成された請求書には、契約 ID、オファー ID、サブスクライバー AWS アカウント ID、サブスクライバーの会社名が関連付けられていない場合があります。

QuickSight テーブルからデータをエクスポートおよびダウンロードする方法については、「Amazon QuickSight [ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

詳細なデータの説明

列	[Description] (説明)
請求書日付	製品サブスクリプションの料金を顧客に請求した日付。
支払期日	YYYY-MM-DD 形式の支払期日。
支払い条件	顧客の AWS 請求書の支払い条件。
請求書 ID	顧客に請求される料金に割り当てられた AWS ID。

列	[Description] (説明)
出品料金の請求書 ID	EMEA AWS SARL、日本、またはオーストラリアの法人 (登録販売者) を通じて AWS Marketplace サブスクリプションが取引される場合、販売者出品料金に VAT を請求するには、販売者のマーケットプレイスオペレーター (AWS EMEA SARL など) が必要です。該当する取引において出品料金に課される VAT の請求書 ID は、ソフトウェアまたは製品のサブスクリプション請求書 ID とは異なります。
サブスクライバーの会社名	製品をサブスクライブしているアカウント名。
サブスクライバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクライバーの E メールドメイン	製品をサブスクライブしているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが liu-jie@example.com の場合、エントリーは example.com です。
サブスクライバーの市町区村	製品をサブスクライブしているアカウントに関連付けられた請求先住所の市町区村。
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある都道府県
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある国
サブスクライバーの郵便番号	製品をサブスクライブしているアカウントに関連付けられた請求先住所の郵便番号。
製品のタイトル	製品のタイトル。
オファー名	販売者が定義したオファー名。
オファー ID	購入者が署名したオファーの識別子。

列	[Description] (説明)
オファターの可視性	オファターがパブリック、プライベート、エンタープライズ契約オファターのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意的な契約データフィードリファレンス。
契約開始日	顧客の製品サブスクリプションの開始日 (MM-DD-YYYY) 形式。将来の日付の契約の場合、この日付は受理日とは異なる場合があります。
契約終了日	契約の有効期限が終了する日付 (MM-DD-YY YY 形式)。従量制/pay-as-you-go サブスクリプションの場合、この日付は JAN-1-9999 に設定されます。
契約受理日	顧客が製品をサブスクライブした日付 (MM-DD-YYYY 形式)。
使用期間終了日	製品の使用期間終了日。
使用期間開始日	製品の使用期間開始日。
支払いステータス	が前回の支払い以降に資金を AWS 回収して銀行口座に支払ったことを確認する請求書に関連付けられたステータス。関連する請求書の支払い資金が回収され、支払いが完了しました。関連する未払請求書の資金は、回収も支払いも行われていません。
支払日	販売者の銀行への支払い AWS が開始された日付。

列	[Description] (説明)
銀行への支払いのトレース ID	銀行への支払いの場合、トレース ID は銀行によって割り当てられます。銀行トレース ID は、販売者の銀行提供の預金通知とレポートを AWS Marketplace レポートの請求書に関連付けるために使用できます。
総収益	顧客に請求される製品の従量課金額または月額料金。
総返金	データカバレッジ期間中に返金が処理された場合に顧客に払い戻されるサブスクリプション費用の合計金額。
出品料金	請求金額から差し引かれる AWS Marketplace 料金額。
出品料の払い戻し	データカバレッジ期間中に返金が処理された場合に返金された AWS Marketplace 料金の一部。
出品料金の割合	請求金額から差し引かれる AWS Marketplace 料金の割合。
販売者のタックスシェア	この取引で請求された米国の消費税と使用税の合計額
販売者のタックスシェアの払い戻し	返金が処理される場合にこの取引に対して払い戻される米国の消費税と使用税の合計金額。
卸売コスト	チャネルパートナーのプライベートオファーマのみ。リセラーに請求される製品のコスト。例えば、リセラーが製造元の製品を販売したときに製造元に支払う金額などです。卸売コストは、希望小売り価格に割引率を掛けたものです。
卸売コストの払い戻し	チャネルパートナーのプライベートオファーマのみ。リセラーから返金された製品のコスト。

列	[Description] (説明)
販売者の純収益	取引に対して請求される合計金額、AWS Marketplace 手数料、返金、および米国の売上税と使用税を差し引いた金額。
通貨	取引の通貨。例えば、取引が US ドルの場合、エントリーは USD です。
取引リファレンス ID	トランザクションを表す一意の識別子。これを使用して、AWS Marketplace レポート間でトランザクションを関連付けることができます。
AWS 登録販売者	取引を促進した事業者の識別子。可能な値は以下のとおりです。 <ul style="list-style-type: none"> • AWS_INC: AWS, Inc. の識別子 (米国を拠点とする) • AWS_EUROPE: Amazon Web Services EMEA SARL (ルクセンブルク拠点) の識別子。 • Amazon Web Services Australia Pty Ltd (AWS オーストラリア) • Amazon Web Services 日本合同会社
再販承認 ID	登録された機会の一意の識別子。
再販承認名	登録された機会の一意の名前。
再販承認の説明	ISV の定義による登録済み機会の説明。
リセラーの会社名	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウント名。
リセラー AWS アカウント ID	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウントの ID。
支払人の会社名	料金の請求先となるアカウントの名前。

列	[Description] (説明)
支払者 AWS アカウント ID	料金の請求先となるアカウントの ID。
支払人の E メールドメイン	料金の請求先となるアカウントに関連付けられている E メールドメイン。例えば、E メールアドレスが liu-jie@example.com の場合、エンタリは example.com です。
支払人の市町区村	ソフトウェア料金の請求先となるアカウントに関連付けられた請求先住所の市区町村。
支払人の都道府県または地域	料金の請求先となるアカウントに関連付けられた請求先住所の都道府県。
支払人の国	料金の請求先となるアカウントに関連付けられた 2 文字の国コード。
支払人の郵便番号	ソフトウェア料金の請求先となるアカウントに関連付けられた請求先住所の郵便番号。
ISV アカウント ID	製品またはサービスの所有者の識別子。
ISV の会社名	製品またはサービスの所有者の商号。
製品 ID	ソフトウェア製品のわかりやすい一意の識別子。
支払済み純収益	販売者に支払われた取引の合計額
未支払いの純収益	販売者に支払われていない取引の合計額
支払い期間	売掛金に関するカテゴリは、資金が回収された期間の範囲です (期限でない、1~30 日、31~60 日など)。

税金ダッシュボード

税ダッシュボードには、での取引に対する米国および国際税に関する可視化データと詳細なデータが表示されます AWS Marketplace。このダッシュボードには、財務オペレーションの AWS Marketplace 管理ポータル Insights タブからアクセスできます。

Amazon QuickSight ダッシュボードの任意のチャート、グラフ、およびテーブルから (.csv または Microsoft Excel ファイルとして) データをエクスポートおよびダウンロードできます。詳細については、「Amazon QuickSight ユーザーガイド」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

運用プロセスと財務プロセスについては、以下のトピックをご確認ください。

トピック

- [税金ダッシュボードの更新頻度](#)
- [セクション 1: コントロール](#)
- [セクション 2: 日付をフィルタリングする](#)
- [セクション 3: メトリクス](#)
- [セクション 4: 税金の傾向](#)
- [セクション 5: 内訳](#)
- [セクション 6: 詳細なデータ](#)
- [米国の免除コード](#)
- [EU 免除コード](#)

税金ダッシュボードの更新頻度

ダッシュボードは毎日更新されます。アップストリームシステムから受信した最新のトランザクションデータが遅延した場合、最新のデータがダッシュボードに反映されるまでに 1~2 日の遅延が発生する可能性があります。

セクション 1: コントロール

ダッシュボードのこのセクションには、税金データを絞り込むためのフィルターが用意されています。また、ビジュアルに含める日付範囲などを設定できるフィルターを分析に追加することも可能です。セクション内で選択済みのフィルターにより、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。

フィルタリングの詳細については、「[Amazon ユーザーガイド](#)」の「[Amazon でのデータのフィルタリング QuickSight](#)」を参照してください。 QuickSight

コントロールの記述

コントロール名	説明
請求書 ID	顧客に請求される料金に割り当てられた AWS ID。
納税対象顧客 AWS アカウント ID	製品サブスクリプションに税金が課されたアカウントの AWS ID。
AWS 登録販売者	取引を促進した事業者の識別子。
顧客の国に税金がかかる	税制上の顧客に関連付けられている 2 文字の国コード
顧客の州または地域に税金がかかる	税制上の顧客に関連付けられた請求先住所の州または地域。
顧客の市区町村	税制上の顧客に関連付けられた請求先住所の市区町村
課税対象	税金、非税金、税金の払い戻しの対象となる出典取引。
納税義務者	<p>AWS または Seller です。販売者が納税義務者である場合は、税金が徴収されます。AWS が納税義務当事者である場合、売上税はによって徴収および送金されます AWS。詳細については、「AWS Marketplace 販売者の税務ヘルプ」を参照してください。</p> <p>税金が徴収されない場合は、値が表示されません。販売者は税金徴収の責任を負うため、請求書ごとに一部の税金が徴収されたかどうかを判断する必要があります。</p>

コントロール名	説明
税金タイプ	取引に適用される税のタイプ。指定できる値はNone、、Sales、および SellerUse、Tax registration type、VAT (付加価値税) CNPJ、、IGST、ですCT。

セクション 2: 日付をフィルタリングする

ダッシュボードのこのセクションには、指定した日付範囲の請求日に基づいてレコードを絞り込むためのフィルターが用意されています。請求書日付内で選択されたフィルターは、メトリクス、傾向、内訳、および詳細なデータセクションに表示されるデータを更新します。例えば、日付フィルターを選択するとPast 30 days、すべてのセクションに過去 30 日以内に税金が課された請求書に関連するデータが反映されます。

セクション 3: メトリクス

ダッシュボードのこのセクションには、税金に関連するメトリクスを視覚化するための主要業績評価指標 (KPI) が表示されます。これには、税額、非税額、税額、税金が課される請求書の数、取引の合計数が含まれます。フィルターセクションの日付フィルターを更新することで、日付範囲を更新できます。

セクション 4: 税金の傾向

ダッシュボードのこのセクションには、指定した日付範囲の税金の傾向が表示されます。日次、、、、など month-over-month、指定した日付集計で傾向を表示 year-over-yearして quarter-over-quarter、税金に関するインサイトを得ることができます。次の税金の傾向情報を利用できます。

- 税金の傾向 - 金額 - 選択した期間の税金と税金の両方のスナップショットを請求日別に表示します。
- 税金の傾向 - 請求書数 - 選択した期間の請求書数のスナップショットを請求書日付別に表示します。

セクション 5: 内訳

ダッシュボードのこのセクションには、請求書 ID、税金タイプ、製品タイトル、顧客地域別に、ビジネスの税金メトリクスが表示されます。

セクション 6: 詳細なデータ

ダッシュボードのこのセクションには、での組織の取引に関する詳細な税金データが表示されます AWS Marketplace。

詳細なデータの説明

列	[Description] (説明)
請求書 ID	顧客に請求される料金に割り当てられた AWS ID。
明細項目 ID	明細項目の一意的識別子。返金取引の明細項目 ID は先渡し取引税のものと同じです。
顧客請求書 ID	顧客請求書 ID
納税義務者	<p>AWS または Seller です。販売者が納税義務者である場合は、税金が徴収されます。AWS が納税義務当事者である場合、売上税はによって徴収および送金されます AWS。詳細については、「AWS Marketplace 販売者の税務ヘルプ」を参照してください。</p> <p>税金が徴収されない場合は、値が表示されません。販売者は税金徴収の責任を負うため、請求書ごとに一部の税金が徴収されたかどうかを判断する必要があります。</p>
トランザクションタイプコード	<p>取引のタイプコード。値には以下のものが含まれます。</p> <ul style="list-style-type: none"> • AWS – 転送税取引。 • REFUND – 全額または一部の返金。 • TAXONLYREFUND – 税金のみの返金。 <p>返金取引の明細項目 ID は元の先渡し取引のものと同じです</p>

列	[Description] (説明)
製品 ID	ソフトウェア製品のわかりやすい一意の識別子。
製品のタイトル	購入された製品の名前。
製品税コード	製品の税プロパティを識別するための標準コード。製品を作成または変更するときにプロパティを選択します。
請求書日付	顧客が製品サブスクリプションに対して請求され、税金が課された日付。
納税対象顧客 AWS アカウント ID	税制上の顧客の AWS アカウント ID。
顧客の国に税金がかかる	税制上の顧客に関連付けられている 2 文字の国コード。
顧客の州または地域に税金がかかる	税制上の顧客に関連付けられた請求先住所の州または地域。
顧客の市区町村	税制上の顧客に関連付けられた請求先住所の市区町村。
顧客の郵便番号	税額計算に使用する郵便番号。
税金タイプ	取引に適用される税のタイプ。指定できる値は None、Sales、および SellerUse、Tax registration type、VAT、CNPJ、IGST です CT。
管轄レベル	納税地に使用される住所の管轄区域。指定できる値は、State、County、City、District です。
税制上の管轄区域	課税管轄地域の名前。
請求書料金	税額と非税額の概要。

列	[Description] (説明)
価格の税金タイプを表示する	顧客に表示される価格の課税タイプ。すべての AWS Marketplace サービスは排他的です。
納税対象金額	該当する管轄区域レベルで課税対象となる取引金額。
非税額	管轄区域別に非課税となる取引金額。
税務管轄率	管轄区域で適用される税率。
税額	管轄区域で課される税額。
税金通貨	請求された税金の通貨。例えば、取引が US ドルの場合、エント리는 USD です。
税計算理由コード	取引が課税対象となるか、非課税となるか、免除とされるか、ゼロ評価済みであるかを管轄区域別に整理したもの。
税計算に使用される日付	取引の税金の計算に使用される日付。
顧客免除証明書 ID	免除証明書の証明書 ID。
顧客免除証明書 ID ドメイン	証明書が Amazon システムに保存されている場所。
顧客免除証明書レベル	免除を提供した管轄区域。
顧客免除コード	免除を指定するコード。例えば RESALE です。
取引リファレンス ID	AWS Marketplace レポート間でトランザクションを関連付けるのに役立つトランザクションの一意的識別子。
AWS 登録販売者	取引を促進した事業者の識別子。

米国の免除コード

免除コード	説明
501C	501C 組織
AP	農業生産
CO	チャリティ組織
直接	直接支払いの許可
DISTSBT	ディストリビューター
DP	国際連合/外交省
DV	無効な退役軍人
EDI	教育機関
FG	連邦政府
HCP	医療提供者
HO	勲章 (非営利または州)
アンケート	保険
IPM	IPM
LB	[Library] (ライブラリ)
MPU	複数の使用場所の免除、デジタル製品/ソフトウェア免除
該当なし	ネイティブアメリカ語
NAI	ネイティブアメリカ個人
NP	非営利組織
NPR	非営利の宗教組織

免除コード	説明
OT	その他
再処理	リセラー
SBE	中小企業の免除
SLG	州/地域政府

EU 免除コード

免除コード	説明
SpecialOrg	VAT の請求が免除される組織。
SpecialZone	免税の対象となる地域。
SplitPayment	購入者は、請求書に記載されている VAT の支払いを税務局に直接行います。

販売業務用ダッシュボード

AWS Marketplace には、売上データの追跡に役立つ複数のダッシュボードが用意されています。

トピック

- [契約および更新ダッシュボード](#)
- [使用状況ダッシュボード](#)

契約および更新ダッシュボード

契約および更新ダッシュボードには、AWS Marketplaceで契約に署名してから 24 時間以内に、契約と更新に関する情報が表示されます。期限切れ間近のサブスクリプションを追跡して更新できるようにするのに役立ちます。このダッシュボードには、販売オペレーションの AWS Marketplace 管理ポータル インサイトタブからアクセスできます。

Amazon QuickSight ダッシュボードの任意のチャート、グラフ、およびテーブルから (.csv または Microsoft Excel ファイルとして) データをエクスポートおよびダウンロードできます。詳細については、「Amazon QuickSight ユーザーガイド」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

トピック

- [セクション 1: コントロール](#)
- [セクション 2: 日付カテゴリの選択](#)
- [セクション 3: メトリクス](#)
- [セクション 4: サブスクリプションのトレンド](#)
- [セクション 5: 内訳](#)
- [セクション 6: 詳細なデータ](#)

セクション 1: コントロール

ダッシュボードの本セクション内には、契約および更新ダッシュボードのデータを絞り込むためのフィルターがあります。以下のフィルターから選択できます。

Note

フィルタリングの詳細については、「[Amazon ユーザーガイド](#)」の「[Amazon でのデータのフィルタリング QuickSight](#)」を参照してください。 QuickSight

コントロールの記述

コントロール名	説明
サブスクリイバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクリイバーの会社名	製品をサブスクライブしているアカウント名。
オファー ID	購入者が署名したオファーの識別子。

コントロール名	説明
オファ어의可視性	オファ어가パブリック、プライベート、またはエンタープライズ契約オファ어의いずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者との間で締結された、契約書の一意的契約データフィードリファレンス。
CPPO フラグ	チャンネルパートナーのプライベートオファ어를使用して契約が締結されたかどうかを示す「はい/いいえ」のフラグ。「はい」の場合、登録販売者がチャンネルパートナーです。「いいえ」の場合、登録販売者はソフトウェアメーカー(独立系ソフトウェアベンダー)です。
ISV AWS アカウント ID	製品またはサービスの所有者のアカウント。
ISV の会社名	製品またはサービスの所有者の商号。
レガシ어의製品 ID	ソフトウェア製品のレガシ어의一意的識別子。
製品のタイトル	製品のタイトル。
再販承認 ID	ISV がチャンネルパートナーに提供した再販承認の ID。
再販承認名	ISV がチャンネルパートナーに提供した再販承認の名前。
リセラー AWS アカウント ID	ISV から卸売で製品またはサービスを購入して顧客に再販したアカウントの ID。
リセラーの会社名	ソフトウェア製造元の製品の販売を許可されているリセラーアカウントの商号。
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。

コントロール名	説明
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある州または地域
サブスクライバーの市町区村	製品をサブスクライブしているアカウントに関連付けられた請求先住所の市町区村。

セクション 2: 日付カテゴリの選択

ダッシュボードのこのセクションには、指定した日付範囲の契約終了日、契約終了期間、または契約開始日に基づいてレコードを絞り込むためのフィルターが用意されています。選択した日付カテゴリ内で選択したフィルターにより、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。

最近署名した契約書に関する情報は、契約開始日で絞り込むことができます。契約の更新について知りたい場合は、契約終了日を使用してフィルタリングできます。

セクション 3: メトリクス

ダッシュボードの本セクションには、主要業績評価メトリクス (KPI) が表示され、主要な契約データを視覚的に比較できます。有効な契約の数と終了した契約の数を示す KPI が表示されます。[日付カテゴリを選択] フィールドの日付条件を更新して、日付範囲を更新できます。

セクション 4: サブスクリプションのトレンド

ダッシュボードの本セクションには、指定した日付範囲のサブスクリプションのトレンドが表示されます。日次、、、など month-over-month、指定された日付集計で傾向を表示 year-over-yearして quarter-over-quarter、契約に関するインサイトを得ることができます。契約と更新のトレンド情報は集計されたレベルで確認できます。

- サブスクリプションのトレンド - 選択した日付集計フィルターと選択した日付カテゴリフィルターに基づいて、有効な契約の数と終了した契約の数のスナップショットが表示されます。契約開始日のサブスクリプションのトレンドを表示するには、[日付カテゴリを選択] フィルターで[契約開始日]を選択します。契約終了日のサブスクリプションのトレンドを表示するには、[日付カテゴリを選択] フィルターで[契約終了日]を選択します。
- 更新内訳 - 特定の日数内に終了した、または終了間近である更新のスナップショットが表示されます。1~30 日前、31~60 日前、または 61 日以上前に終了した契約など、複数の期間の更新内訳

を表示できます。また、0～30日、31～60日というような期間以内に終了する、終了間近の契約を絞り込むこともできます。

セクション 5: 内訳

ダッシュボードのこのセクションには、アクティブなサブスクライバー、ISV、リセラーの会社名にわたるビジネス指標が表示されます。契約 ID、終了契約 ID、オファー ID、更新契約 ID、サブスクライバーの数を表示できます。

セクション 6: 詳細なデータ

ダッシュボードのこのセクションには、契約、オファー、製品、契約の提案者、サブスクライバー、再販承認、リセラー、ISV に関する詳細なデータが表示されます。

Note

QuickSight テーブルからデータをエクスポートおよびダウンロードする方法の詳細については、「Amazon QuickSight [ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

詳細なデータの説明

列	[Description] (説明)
サブスクライバーの会社名	サブスクライバーの会社の名前。
サブスクライバーの AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクライバーの E メールドメイン	製品をサブスクライブしているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが liu-jie@example.com の場合、エントリは example.com です。
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある国

列	[Description] (説明)
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある州または地域
サブスクライバーの市町区村	製品をサブスクライブしているアカウントに関連付けられた請求先住所の市町区村。
サブスクライバーの郵便番号	製品をサブスクライブしているアカウントに関連付けられた請求先住所の郵便番号。
製品のタイトル	製品のタイトル。
製品 ID	ソフトウェア製品のわかりやすい一意の識別子。
レガシーの製品 ID	ソフトウェア製品のレガシーの一意の識別子。
オファー名	販売者が定義したオファー名。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、またはエンタープライズ契約オファーのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意の契約データフィードリファレンス。
契約開始日	顧客の製品サブスクリプションの開始日 (MM-DD-YYYY) 形式。将来の日付の契約の場合、この日付は受理日とは異なる場合があります。
契約受理日	顧客が製品をサブスクライブした日付 (MM-DD-YYYY 形式)。

列	[Description] (説明)
契約終了日	契約の有効期限が終了する日付 (MM-DD-YY YY 形式)。従量制/pay-as-you-go サブスクリプションの場合、この日付は JAN-1-9999 に設定されます。
リセラーの会社名	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウント名。
リセラーの AWS アカウント ID	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウントの ID。
再販承認 ID	登録された機会の一意の識別子。
再販承認名	登録された機会の一意の名前。
再販承認の説明	ISV の定義による登録済み機会の説明。
CPPO フラグ	Amazon の計算フィールド QuickSight。

列	[Description] (説明)
契約の意図	<p>現在の契約が、以前の agreement_id 列に入力された以前の契約の更新またはアップグレードである場合に入力されます。可能な値は以下のとおりです:</p> <ul style="list-style-type: none"> • アップグレード: プライベートオファーの場合、アップグレードとはあらゆる変更 (アップグレード、ダウングレード、修正、拡張、更新) を指す広義の用語です。パブリックオファーの場合、アップグレードとは、条件を変更 (別のオファーを使用) して新しい契約を結ぶという特定の契約であり、その特定の契約はアーカイブされます。 • 更新: パブリックオファーのみが対象です。更新契約の開始日が特定の契約の終了日であり、オファー ID は変更されていないが契約 ID は変更されている場合。例えば、契約 A が契約 B になったとします。 • 自動更新: パブリックオファーのみが対象です。現在の契約が終了する時点で新しい契約が作成される場合。購入者は自動更新を有効にしています。 • 新規: 購入者が (オファーを使用して) 条件に同意し、新しい契約をサブスクライブした場合。
以前の契約 ID	本契約が更新またはアップグレードされた場合の契約リファレンス。
次の契約 ID	現在の契約が更新またはアップグレードされた場合の、次の契約の契約 ID。
以前のオファー ID	現在の契約が更新またはアップグレードされた場合の、以前の契約に関連付けられたオファー ID。

列	[Description] (説明)
次のオファー ID	現在の契約が更新またはアップグレードされた場合の、次の契約に関連付けられたオファー ID。
次の契約 ID の承認日	購入者が次の契約を承諾した日付。
提案者の AWS アカウント ID	製品またはサービスの所有者の識別子。
提案者の会社名	製品またはサービスの所有者の商号。
ISV AWS アカウント ID	販売者の AWS 識別子。
ISV の会社名	販売者のビジネス名。
契約終了 (日単位)	QuickSight 内の計算フィールド。本日から契約が終了するまでの残り日数。
契約終了期間	の計算フィールド QuickSight。契約が終了するまでの期間。

使用状況ダッシュボード

使用状況ダッシュボードは、SaaS およびサーバー使用量ベースの製品を使用しているお客様に可視化ときめ細かなデータを提供します。AWS Marketplace 販売者は、このダッシュボードを使用して、使用状況ベースの製品全体の顧客の消費を追跡し、製品サポート、価格設定、パブリックオファーからプライベートオファーへの変換、製品の廃止を決定できます。このダッシュボードには、販売オペレーションのインサイトタブ AWS Marketplace 管理ポータル からアクセスできます。過去 6 か月間のデータが表示されます。この期間は移動します。

Amazon QuickSight ダッシュボードの任意のチャート、グラフ、およびテーブルから (.csv ファイルまたは Microsoft Excel ファイルとして) データをエクスポートおよびダウンロードできます。詳細については、「[Amazon QuickSight ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

Note

このダッシュボードには、オファー作成時に指定されたすべてのディメンションキーの使用状況が表示されます。例えば、Amazon マシンイメージ (AMI) 製品の場合、ディメンションキーはインスタンスタイプであり、価格が 0 USD であっても、オファーで指定されているすべてのインスタンスタイプがこのダッシュボードに表示されます。オファーの作成時に指定されていなかったディメンションの製品使用状況を確認するには、必要なディメンションを含むように製品を再公開することを検討します。

トピック

- [使用状況ダッシュボードの更新頻度](#)
- [セクション 1: コントロール](#)
- [セクション 2: フィルター](#)
- [セクション 3: メトリクス](#)
- [セクション 4: トレンド](#)
- [セクション 5: 内訳](#)
- [セクション 6: 詳細なデータ](#)

使用状況ダッシュボードの更新頻度

請求済み収益ダッシュボードは毎日午後 4 時 (UTC 時間の午前 0 時) に更新されます。使用状況データは上流のデータソースから受信されるため、遅延が発生する可能性があることに注意してください。使用日と使用状況報告日を参照して、使用がなされた時期を、ダッシュボード上で報告対象期間との対比で明確にすることができます。

Amazon QuickSight ダッシュボードの任意のチャート、グラフ、およびテーブルから (.csv ファイルまたは Microsoft Excel ファイルとして) データをエクスポートおよびダウンロードできます。詳細については、「[Amazon QuickSight ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

運用プロセスと財務プロセスについては、以下のトピックをご確認ください。

セクション 1: コントロール

ダッシュボードの本セクション内には、使用状況データを絞り込むためのフィルターがあります。例えば、以下のフィルターから選択できます。

Note

フィルタリングの詳細については、[「Amazon ユーザーガイド」の「Amazon でのデータのフィルタリング QuickSight」](#)を参照してください。 QuickSight

コントロールの記述

コントロール名	説明
エンドユーザー会社名	製品を使用しているアカウント名。
エンドユーザー AWS アカウント ID	製品を使用しているアカウントの ID。
エンドユーザー国	製品を使用しているアカウントに関連付けられている 2 文字の国コード。
製品のタイトル	製品のタイトル。
製品コード	製品の計測に使用される、権利が付与された既存の製品コード。この値は、データをレポートに結合する場合や、AWS Marketplace メータリングサービスで提供される情報を参照する場合にも使用されます。
オファー ID	購入者が署名したオファーの識別子。
オファーの可視性	オファーがパブリック、プライベート、エンタープライズ契約オファーのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意の契約データフィードリファレンス。
ディメンションキー	製品の使用状況に関連するリソースタイプ。ディメンションキーは SaaS およびサーバー使用量ベースの製品に適用されます。
サブスクリバの会社名	製品をサブスクライブしているアカウント名。

コントロール名	説明
サブスクライバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられている 2 文字の国コード。
リセラーの会社名	製品製造元の製品の販売を認可されているリセラーアカウントの名前。
リセラー AWS アカウント ID	ISV から卸売で製品またはサービスを購入して顧客に再販したアカウントの ID。
再販承認 ID	ISV から卸売で製品またはサービスを購入して顧客に再販したアカウントの ID。
CPPO フラグ	チャンネルパートナーのプライベートオファーを使用して契約が締結されたかどうかを示す「はいいいえ」のフラグ。「はい」の場合、登録販売者がチャンネルパートナーです。「いいえ」の場合、記録上の販売者は製品製造元 (独立系ソフトウェアベンダー) です。

セクション 2: フィルター

ダッシュボードの本セクションには、使用日に基づいてレコードを絞り込むためのフィルターがあります。これらのフィルターで選択された値により、メトリクス、トレンド、内訳、詳細データセクションに表示されるデータが更新されます。デフォルトの選択では、過去 6 か月間の使用量のデータを取得するようになっています。

セクション 3: メトリクス

ダッシュボードの本セクションには、推定使用単位数、使用のある顧客、使用量の多い製品などの、使用に関するメトリクスを視覚化する主要業績評価指標 (KPI) が表示されます。これらのフィルターセクションの使用日条件を更新して、日付範囲を更新できます。主要メトリクスにはすべての単位タイプのデータが表示されることに注意してください。

セクション 4: トレンド

ダッシュボードの本セクションには、指定した日付範囲の使用状況のトレンドが表示されます。日次、、、など month-over-month、指定した日付集計で傾向を表示 year-over-year して quarter-over-quarter、使用状況を把握できます。また、使用単位タイプを選択して、その使用状況のトレンドをグラフィカルに表示することもできます。

セクション 5: 内訳

ダッシュボードの本セクションには、会社名、製品名、ディメンションキー、選択した単位タイプのオファー ID についてのビジネスの推定使用状況メトリクスが表示されます。また、表示するエントリ数も選択できます。

セクション 6: 詳細なデータ

ダッシュボードの本セクションには、使用状況、オファー、製品、サブスクライバー、支払者、エンドユーザー、再販認可、リセラー、独立系ソフトウェアベンダー (ISV) に関する詳細なデータが表示されます。詳細データテーブルには、すべての単位タイプのデータが表示されることに注意してください。

月末に請求が確定するまでは、収益は推定値とみなす必要があります。使用料ベースの請求書は、当月分の使用量について、翌月の 2 日または 3 日に購入者に提示されます (例えば、11 月 1 日から 11 月 30 日までの使用量を計測した顧客には、12 月 2 日または 12 月 3 日に使用量の請求書が提示されます)。計測された使用量は、実際の使用日の数日後にこのダッシュボードに届く場合があるため、使用日と、使用量の報告日は異なる場合があります。つまり、当月 up-to-date の追跡のために、毎月ダッシュボードに毎日アクセスする必要がある場合があります。権威のある顧客請求情報については、[財務業務] タブの「[請求収益ダッシュボード](#)」を参照してください。

Note

フィルタリングの詳細については、「[Amazon QuickSight ユーザーガイド](#)」の「[ビジュアルからのデータのエクスポート](#)」を参照してください。

詳細なデータの説明

列	[Description] (説明)
使用日	顧客が製品を使用した日付。

列	[Description] (説明)
使用状況レポート日	顧客の製品使用状況がインサイトダッシュボードに表示された日付。
エンドユーザー会社名	製品を使用しているアカウント名。
エンドユーザー AWS アカウント ID	製品を使用しているアカウントの ID。
エンドユーザー E メールドメイン	製品を使用しているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが abc@example.com の場合、エントリは example.com です。
エンドユーザー市区町村	製品を使用したアカウントに関連付けられている市区町村。
エンドユーザーの都道府県またはリージョン	製品を使用したアカウントに関連付けられている都道府県または地域。
エンドユーザー国	製品を使用しているアカウントに関連付けられている 2 文字の国コード。
エンドユーザー郵便番号	製品を使用しているアカウントに関連付けられた請求先住所の郵便番号。
製品のタイトル	製品のタイトル。
レガシーの製品 ID	製品のレガシーの一意の識別子。
製品 ID	製品のわかりやすい一意の識別子。
製品コード	製品の計測に使用される、権利が付与された既存の製品コード。この値は、データをレポートに結合する場合や、AWS Marketplace Metering Service で提供される情報を参照する場合にも使用されます。
オファー ID	購入者が署名したオファーの識別子。

列	[Description] (説明)
オファー名	販売者が定義したオファー名。
オファーの可視性	オファーがパブリック、プライベート、またはエンタープライズ契約オファーのいずれになるか。
契約 ID	製品の使用を開始するために提案者と承諾者の間で締結された、契約書の一意的な契約データフィードリファレンス。
契約受理日	顧客が製品をサブスクライブした日時 (UTC) のタイムスタンプ。
契約開始日	顧客の製品サブスクリプションが開始された日付 (UTC) のタイムスタンプ。将来の日付の契約の場合、この日付は受理日とは異なる場合があります。
契約終了日	契約が失効する日付 (UTC)。従量制/pay-as-you-goサブスクリプションの場合、この日付は 9999 年 1 月 1 日午前 12 時 00 分に設定されます。
ディメンションキー	製品の使用状況に関連するリソースタイプ。ディメンションキーは SaaS およびサーバー使用量ベースの製品に適用されます。
リージョン	Amazon EC2 インスタンスを購入者がデプロイしたリージョン。
推定使用量	製品について記録された使用量。
使用単位タイプ	使用量が記録される単位タイプ。

列	[Description] (説明)
予想収益	製品の使用量からの予想収益。月末に請求が確定するまでは、収益は推定値とみなす必要があります。使用量ベースの請求書は、当月分の使用量が、翌月の 2 日または 3 日に購入者に提示されます。
通貨	取引の通貨。例えば、取引が米ドル建ての場合、エント리는 USD です。
サブスクライバーの会社名	製品をサブスクライブしているアカウント名。
サブスクライバー AWS アカウント ID	製品をサブスクライブしているアカウントの ID。
サブスクライバーの E メールドメイン	製品をサブスクライブしているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが abc@example.com の場合、エント리는 example.com です。
サブスクライバーの市町区村	製品をサブスクライブしているアカウントに関連付けられた請求先住所の市町区村。
サブスクライバーの州または地域	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある都道府県
サブスクライバーの国	製品をサブスクライブしているアカウントに関連付けられた請求先住所がある国
サブスクライバーの郵便番号	製品をサブスクライブしているアカウントに関連付けられた請求先住所の郵便番号。
支払人の会社名	製品の支払いを行っているアカウント名。
支払者 AWS アカウント ID	製品の支払いを行っているアカウントの ID。

列	[Description] (説明)
支払人の E メールドメイン	製品の支払いを行っているアカウントに関連付けられた E メールドメイン。例えば、E メールアドレスが abc@example.com の場合、エントリーは example.com です。
支払人の市町区村	製品の支払いを行っているアカウントに関連付けられた請求先住所の市町区村。
支払人の都道府県または地域	製品の支払いを行っているアカウントに関連付けられた請求先住所がある都道府県
支払人の国	製品の支払いを行っているアカウントに関連付けられた請求先住所がある国
支払人の郵便番号	製品の支払いを行っているアカウントに関連付けられた請求先住所の郵便番号。
リセラーの会社名	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウント名。
リセラー AWS アカウント ID	ISV から卸売コストで製品またはサービスを購入して顧客に再販したアカウントの ID。
再販承認 ID	登録された再販オポチュニティの一意的識別子。
再販承認名	登録された再販オポチュニティの一意的名前。
再販承認の説明	登録された再販オポチュニティの説明。
CPPO フラグ	チャンネルパートナーのプライベートオファーを使用して契約が締結されたかどうかを示す「はい/いいえ」のフラグ。「はい」の場合、登録販売者がチャンネルパートナーです。「いいえ」の場合、記録上の販売者は製品メーカー (ISV) です。

列	[Description] (説明)
ISV の会社名	製品またはサービスの所有者の名前。
ISV AWS アカウント ID	製品またはサービスの所有者の識別子。

AWS Marketplace Vendor Insights

AWS Marketplace Vendor Insights は、組織が実行するソフトウェアリスク評価を簡素化し、信頼できる調達ソフトウェアを保護し、基準を満たす機能です。AWS Marketplace Vendor Insights を使用すると、購入者は 1 つのコンソールから製品のセキュリティプロファイルをほぼリアルタイムでモニタリングできます。AWS Marketplace Vendor Insights を使用すると、購入者の調達プロセスが簡単になり、販売者の売上が増加する可能性があります。ソフトウェア製品のセキュリティとコンプライアンス情報のダッシュボードを提供することにより、購入者の評価にかかる労力を低減できます。

AWS Marketplace Vendor Insights ダッシュボードのすべてのセキュリティおよびコンプライアンス情報は、以下のソースから収集された証拠に基づいています。

- AWS Marketplace Vendor Insights のセキュリティ自己評価とコンセンサス評価イニシアチブアンケート (CAIQ) を含む販売者の自己認証
- 業界標準の監査報告書 (例えば、国際標準化機構 ISO 27001)
- 販売者の本番環境からの証拠収集を自動化する、AWS Audit Manager

AWS Marketplace Vendor Insights は、製品に関するコンプライアンスアーティファクトとセキュリティコントロール情報を収集し、ダッシュボードに表示します。このダッシュボードは、販売者の自己評価から得たデータ、監査レポートからの証拠、および AWS アカウントから得たライブエビデンスを取得します。このデータはセキュリティコントロールにフィードされ、次に購入者が確認できるようにダッシュボードにフィードされます。ライブエビデンスとは、複数のソースからのデータを常に更新して最新の情報を表示する方法です。AWS Config は、販売者の環境では有効になっています。設定、有効になっているバックアップ、その他の情報に関するデータは自動的に更新されます。例えば、製品のアクセスコントロールが [準拠] していて、Amazon Simple Storage Service (Amazon S3) バケットが公開されたとします。ダッシュボードには、コントロールのステータスが [準拠] から [未定] に変わったことが表示されます。

AWS Marketplace Vendor Insights を使用する AWS アカウント 前に、 でベースラインリソースとインフラストラクチャを設定する必要があります。セットアップが完了すると、AWS Marketplace Vendor Insights は情報を収集し、 で Software as a Service (SaaS) 製品のセキュリティプロファイルを生成できます AWS Marketplace。

内容

- [AWS Marketplace Vendor Insights について](#)
- [AWS Marketplace Vendor Insights のセットアップ](#)

- [AWS Marketplace Vendor Insights プロファイルの表示](#)
- [AWS Marketplace Vendor Insights でのスナップショットの管理](#)
- [AWS Marketplace Vendor Insights でのアクセスの制御](#)

AWS Marketplace Vendor Insights について

AWS Marketplace Vendor Insights は、製品のコンプライアンスアーティファクトとセキュリティコントロール情報を収集し、ダッシュボードに表示します。このダッシュボードは、製品所有者の自己評価から得たデータ、監査レポートからの証拠、および AWS アカウントから得たライブエビデンスを取得します。このデータはセキュリティコントロールにフィードされ、次に購入者が確認できるようにダッシュボードにフィードされます。

ダッシュボードには、複数のセキュリティコントロールカテゴリから AWS Marketplace Vendor Insights によって収集された証拠ベースの情報が表示されます。これにより、セキュリティプロファイルをほぼリアルタイムで確認できるため、購入者と販売者間の話し合いが減ります。購入者は、評価を完了する販売者の情報を数時間以内に検証できます。AWS Marketplace Vendor Insights は、販売者がセキュリティおよびコンプライアンス体制の情報 up-to-date を自動的に保持するメカニズムを提供します。購入者とオンデマンドで情報を共有できるため、ランダムにアンケートに回答する必要がなくなります。

AWS Marketplace Vendor Insights は、次の 3 つのソースから証拠ベースの情報を収集します。

- ベンダーの自己評価 – サポートされている自己評価には、AWS Marketplace Vendor Insights のセキュリティ自己評価とコンセンサス評価イニシアチブアンケート (CAIQ) が含まれます。
- 本番環境アカウント - 複数のコントロールのうち、25 のコントロールが本番環境アカウントからのライブエビデンスの収集をサポートしています。各コントロールのライブ証拠は、1 つ以上の AWS Config rules を使用して AWS リソースの設定を評価することで生成されます。は証拠 AWS Audit Manager をキャプチャし、AWS Marketplace Vendor Insights が使用できるように準備します。オンボーディング AWS CloudFormation テンプレートは、ライブ証拠収集を有効にするために必要な前提条件のステップを自動化します。AWS Config は販売者の環境で有効になっています。設定、有効になっているバックアップ、その他の情報に関するデータは自動的に更新されます。例えば、製品のアクセスコントロールが [準拠] していて、Amazon S3 バケットが公開されたとします。ダッシュボードには、コントロールのステータスが [準拠] から [未定] に変わったことが表示されます。
 - AWS Config と AWS Audit Manager サービスをオンにします。
 - AWS Config ルールの作成と AWS Audit Manager 自動評価。

- AWS Marketplace Vendor Insights が評価結果を取得できるように AWS Identity and Access Management (IAM) ロールをプロビジョニングします。
- [ISO 27001 および SOC2 Type II レポート] - コントロールカテゴリは、国際標準化機構 (ISO) レポートおよび System and Organization Controls (SOC2) レポート内のコントロールにマッピングされます。これらのレポートを AWS Marketplace Vendor Insights と共有すると、Vendor Insights はこれらのレポートから関連する証拠を抽出してダッシュボードに表示できます。

AWS Marketplace Vendor Insights のセットアップ

次の手順では、Software as a Service (SaaS) リストで AWS Marketplace Vendor Insights を設定するための大まかな手順について説明します。SaaS

SaaS リストに AWS Marketplace Vendor Insights を設定するには

1. [the section called “セキュリティプロファイルを作成する”](#).
2. (オプション) [the section called “証明書のアップロード”](#)。
3. [the section called “自己評価をアップロードする”](#).
4. (オプション) [the section called “AWS Audit Manager 自動評価を有効にする”](#)。

セキュリティプロファイルを作成する

セキュリティプロファイルにより、購入者はソフトウェア製品のセキュリティ状況に関する詳細な情報を得ることができます。セキュリティプロファイルは、自己評価、認定、AWS Audit Manager 自動評価など、関連するデータソースを使用します。

Note

作成できるセキュリティプロファイルは数に限りがあります。より多くのセキュリティプロファイルを作成するには、クォータの引き上げをリクエストします。詳細については、「AWS 全般のリファレンス」の「[AWS の Service Quotas](#)」を参照してください。

セキュリティプロファイルを作成するには

1. AWS Marketplace 販売者アカウントにアクセスできる IAM ユーザーまたはロールを使用してサインインします。

2. [製品] を選択し、[SaaS] を選択して [SaaS 製品] ページに移動します。
3. [製品] を選択します。
4. [ベンダーインサイト] タブを選択し、[セキュリティプロファイルの追加をサポートに問い合わせる] を選択します。
5. フォームに入力し、[送信] を選択します。

AWS Marketplace Seller Operations チームがセキュリティプロファイルを作成します。セキュリティプロファイルの作成が完了すると、フォームに記載されている受信者に通知メールが送信されます。

証明書のアップロード

証明書は、複数のディメンションにわたる製品のセキュリティ体制の証拠を提供するデータソースです。AWS Marketplace Vendor Insights は、以下の証明書をサポートしています。

- FedRAMP 認定 - 米国政府のクラウドセキュリティ標準への準拠を検証します
- GDPR コンプライアンスレポート - 一般データ保護規則 (GDPR) の要件の遵守、個人データおよび個人のプライバシー権の保護を実証します。
- HIPAA コンプライアンスレポート - 保護対象の医療情報を保護する、「医療保険の相互運用性と説明責任に関する法令 (HIPAA)」の規制に準拠していることを証明します。
- ISO/IEC 27001 監査報告書 - 情報セキュリティ基準に重点を置いた国際標準化機構 (ISO)/国際電気標準会議 (IEC) 27001 への準拠を確認しています
- PCI DSS 監査報告書 - PCI セキュリティ標準審議会が設定したセキュリティ基準への準拠を証明します。
- SOC 2 タイプ 2 監査報告書 - サービス組織統制 (SOC) のデータプライバシーとセキュリティ統制への準拠を確認します。

証明書のアップロードするには

1. [ベンダーインサイト] タブで、[データソース] セクションに移動します。
2. [証明書] で [認定資格をアップロード] を選択します。
3. [認定資格の詳細] で、必要な情報を入力し、証明書をアップロードします。
4. (オプション) [タグ] で、新しいタグを追加します。

Note

タグの詳細については、「AWS リソースのタグ付けユーザーガイド」の「[Tagging your AWS resources](#)」を参照してください。

5. [証明書をアップロード] を選択します。

Note

証明書は現在のセキュリティプロファイルに自動的に関連付けられます。アップロード済みの証明書を関連付けることもできます。製品詳細ページで、[認定] の下の [認定を関連付ける] を選択し、リストから認定を選択して、[認定を関連付ける] を選択します。証明書をアップロードしたら、製品詳細ページの [証明書のダウンロード] ボタンを使用してダウンロードできます。[認定書の更新] ボタンを使用して認定書の詳細を更新することもできます。

証明書のステータスは、証明書の詳細が検証ValidationPendingされるまで に変わります。データソースの処理中と処理後に、別のステータスが表示されます。

- 使用可能 - データソースがアップロードされ、システム検証が正常に完了しました。
- AccessDenied – データソースの外部ソースリファレンスは、AWS Marketplace Vendor Insights が読み取ることができなくなります。
- ResourceNotFound – データソースの外部ソースリファレンスは、VendorInsights で読み取ることができなくなります。
- ResourceNotSupported – データソースはアップロードされましたが、提供されたソースはまだサポートされていません。検証エラーの詳細については、ステータスメッセージを参照してください。
- ValidationPending – データソースはアップロードされましたが、システム検証はまだ実行中です。この段階では実行できるアクション項目はありません。ステータスが Available、ResourceNotSupported、または に更新されず ValidationFailed。
- ValidationFailed – データソースがアップロードされましたが、1 つ以上の理由でシステム検証が失敗しました。検証エラーの詳細については、ステータスメッセージを参照してください。

自己評価をアップロードする

自己評価は、製品のセキュリティ体制の証拠を提供するデータソースの一種です。AWS Marketplace Vendor Insights は、以下の自己評価をサポートしています。

- AWS Marketplace Vendor Insights の自己評価
- コンセンサス評価イニシアティブアンケート (CAIQ)

自己評価をアップロードするには

1. [ベンダーインサイト] タブで、[データソース] セクションに移動します。
2. [自己評価] で [自己評価をアップロード] を選択します。
3. [自己評価の詳細] で、次の情報を入力します。
 - a. [名前] - 自己評価の名前を入力します。
 - b. [タイプ] - ドロップダウンリストから評価タイプを選択します。

Note

[ベンダーインサイトのセキュリティ自己評価] を選択した場合は、[テンプレートをダウンロード] を選択して自己評価をダウンロードします。スプレッドシートの回答ごとに [はい]、[いいえ]、または [N/A] を選択します。

4. 完了した評価をアップロードするには、[自己評価をアップロード] を選択します。
5. (オプション) [タグ] で、新しいタグを追加します。

Note

タグの詳細については、[「AWS リソースのタグ付けユーザーガイド」](#)の「AWS リソースのタグ付け」を参照してください。

6. [自己評価をアップロード] を選択します。

Note

自己評価は現在のセキュリティプロファイルに自動的に関連付けられます。アップロード済みの自己評価を関連付けることもできます。製品詳細ページで、[自己評価] の下の

[自己評価を関連付ける] を選択し、リストから自己評価を選択して、[自己評価を関連付ける] を選択します。

自己評価をアップロードしたら、製品詳細ページの [自己評価をダウンロード] ボタンを使用してダウンロードできます。[自己評価の更新] ボタンを使用して自己評価の詳細を更新することもできます。

ステータスは、次のいずれかの値に更新されます。

- 使用可能 - データソースがアップロードされ、システム検証が正常に完了しました。
- AccessDenied - データソースの外部ソースリファレンスは、VendorInsights で読み取ることができなくなります。
- ResourceNotFound - データソースの外部ソースリファレンスは、VendorInsights で読み取ることができなくなります。
- ResourceNotSupported - データソースはアップロードされましたが、提供されたソースはまだサポートされていません。検証エラーの詳細については、ステータスメッセージを参照してください。
- ValidationPending - データソースはアップロードされましたが、システム検証はまだ実行中です。この段階では実行できるアクション項目はありません。ステータスが Available、ResourceNotSupported、または に更新されます ValidationFailed。
- ValidationFailed - データソースがアップロードされましたが、1 つ以上の理由でシステム検証が失敗しました。検証エラーの詳細については、ステータスメッセージを参照してください。

AWS Audit Manager 自動評価を有効にする

AWS Marketplace Vendor Insights は、複数の AWS のサービスを使用して、セキュリティプロファイルの証拠を自動的に収集します。

自動評価には、次の AWS のサービス とリソースが必要です。

- AWS Audit Manager - AWS Marketplace Vendor Insights のセットアップを簡素化するために、必要なリソースのプロビジョニングと設定 StackSetsを行う AWS CloudFormation スタックとを使用します。スタックセットは、AWS Configによって自動的に入力されるコントロールを含む自動評価を作成します。

の詳細については AWS Audit Manager、 「 [AWS Audit Manager ユーザーガイド](#) 」 を参照してください。

- AWS Config – スタックセットは、AWS Config 必要な AWS Config ルールを設定するためのコンフォーマンスパックをデプロイします。これらのルールにより、Audit Manager の自動評価は、そのに AWS のサービス デプロイされた他の のライブ証拠を収集できます AWS アカウント。AWS Config 機能の詳細については、「[AWS Config デベロッパーガイド](#)」を参照してください。

Note

記録の最初の月は、その後の月 AWS Config と比較して、アカウントでのアクティビティが増加することがあります。最初のブートストラッププロセス中に、は、が記録するように AWS Config 選択したアカウント内のすべてのリソース AWS Config を確認します。エフェメラルワークロードを実行している場合、これらの一時リソースの作成と削除に関連する設定変更が記録されるため AWS Config、からのアクティビティが増加することがあります。一時的なワークロードとは、必要なときにロードされて実行されるコンピューティングリソースを一時的に使用することです。エフェメラルワークロードの例には、Amazon Elastic Compute Cloud (Amazon EC2) スポットインスタンスや Amazon EMR ジョブ、AWS Auto Scaling、AWS Lambdaがあります。エフェメラルワークロードの実行によるアクティビティの増加を回避するには、これらのタイプのワークロードを AWS Config をオフにした別のアカウントで実行できます。このアプローチでは、設定の記録やルール評価が増えるのを防ぐことができます。

- Amazon S3 - このスタックセットは、以下の 2 つの Amazon Simple Storage Service (Amazon S3) バケットを作成します。
 - vendor-insights-stack-set-output-bucket-{account number} – このバケットには、スタックセット実行からの出力が含まれています。AWS Marketplace Seller Operations チームは出力を使用して、自動化されたデータソース作成プロセスを完了します。
 - vendor-insights-assessment-reports-bucket-{account number} – 評価レポートをこの Amazon S3 バケットに AWS Audit Manager 発行します。評価レポートの公開に関する詳細については、「AWS Audit Manager ユーザーガイド」の「[評価レポート](#)」を参照してください。

Amazon S3 機能でのファイルの保存に関する詳細については、「[Amazon S3 ユーザーガイド](#)」を参照してください。

- IAM – オンボーディングスタックセットは、アカウントで次の AWS Identity and Access Management (IAM) ロールをプロビジョニングします。
 - VendorInsightsPrerequisiteCFT.yml テンプレートをデプロイすると、管理者ロール AWSVendorInsightsOnboardingStackSetsAdmin と実行ロール AWSVendorInsightsOnboardingStackSetsExecution が作成されます。スタックセット

は管理者ロールを使用して、必要なスタックを複数の AWS リージョン スタックに同時にデプロイします。管理者ロールは、AWS Marketplace Vendor Insights のセットアッププロセスの一環として、必要な親スタックとネストされたスタックをデプロイする実行ロールを引き受けません。セルフマネージド型アクセス許可の詳細については、「[AWS CloudFormation ユーザーガイド](#)」の「セルフマネージド型のアクセス許可を付与する」を参照してください。

- このAWSVendorInsightsRoleロールは、AWS Marketplace Vendor Insights に AWS Audit Manager リソース内の評価を読み取るためのアクセスを提供します。AWS Marketplace Vendor Insights は、Vendor Insights AWS Marketplace プロファイルの評価で見つかった証拠を表示します。
- AWSVendorInsightsOnboardingDelegationRole は、AWS Marketplace Vendor Insights にvendor-insights-stack-set-output-bucketバケット内のオブジェクトを一覧表示して読み取るためのアクセスを提供します。この機能を使用すると、AWS Marketplace カタログオペレーションチームが AWS Marketplace Vendor Insights プロファイルの設定を支援できます。
- このAWSAuditManagerAdministratorAccessロールは、の有効化または無効化 AWS Audit Manager、設定の更新、評価、コントロール、フレームワークの管理を行うための管理アクセスを提供します。ユーザーまたはチームがこの役割を引き受け、AWS Audit Managerで自動評価のためのアクションを実行できます。

AWS Audit Manager 自動評価を有効にするには、オンボーディングスタックをデプロイする必要があります。

オンボーディングスタックをデプロイする

AWS Marketplace Vendor Insights の設定を簡素化するために、AWS CloudFormation スタックとを使用します。これにより StackSets、必要なリソースのプロビジョニングと設定が処理されます。複数のアカウントまたは複数の AWS リージョン SaaS ソリューションがある場合は、中央管理アカウントからオンボーディングスタックをデプロイ StackSets できます。

の詳細については CloudFormation StackSets、「[ユーザーガイド](#)」の [AWS CloudFormation StackSets](#) 「の使用AWS CloudFormation」を参照してください。

AWS Marketplace Vendor Insights の設定では、次の CloudFormation テンプレートを使用する必要があります。

- VendorInsightsPrerequisiteCFT – アカウント CloudFormation StackSets で を実行するために必要な管理者ロールとアクセス許可を設定します。このスタックは販売者アカウントで作成してください。

- VendorInsightsOnboardingCFT - 必要な AWS のサービスを設定し、適切な IAM アクセス許可を設定します。これらのアクセス許可により、AWS Marketplace Vendor Insights は で実行されている SaaS 製品のデータを収集 AWS アカウントし、そのデータを AWS Marketplace Vendor Insights プロファイルに表示できます。このスタックは、 を通じて SaaS ソリューションをホストしている販売者アカウントと本番稼働用アカウントの両方で作成します StackSets。

VendorInsightsPrerequisiteCFT スタックを作成します。

VendorInsightsPrerequisiteCFT CloudFormation スタックを実行すると、スタックセットのオンボーディングを開始する IAM アクセス許可が設定されます。

VendorInsightsPrerequisiteCFT スタックを作成するには

1. ウェブサイトの [AWS Samples Repo for Vendor Insights テンプレートフォルダ](#) から最新の VendorInsightsPrerequisiteCFT.yml ファイルを確認してダウンロードします GitHub。
2. AWS Marketplace 販売者アカウント AWS Management Console を使用して にサインインし、 <https://console.aws.amazon.com/cloudformation> でコンソールを開きます AWS CloudFormation 。
3. CloudFormation コンソールのナビゲーションペインで、スタック を選択し、ドロップダウンからスタックの作成と新しいリソース (標準) を選択します。(ナビゲーションペインが表示されていない場合は、左上隅にあるナビゲーションペインを選択して展開します)。
4. [テンプレートの指定] で、[テンプレートファイルのアップロード] を選択します。ダウンロードした VendorInsightsPrerequisiteCFT.yml ファイルをアップロードするには、[ファイルを選択] を使用します。次いで、[次へ] を選択します。
5. スタックの名前を入力し、[次へ] を選択します。
6. (オプション) 必要に応じてスタックオプションを設定します。

[次へ] をクリックします。

7. [確認] ページで選択内容を確認します。変更するには、変更する領域で [編集] を選択します。スタックを作成する前に、[能力] エリアの確認チェックボックスを選択する必要があります。

[送信] を選択します。

8. スタックが作成されたら、[リソース] タブを選択し、作成された以下のロールを書き留めます。

- AWSVendorInsightsOnboardingStackSetsAdmin
- AWSVendorInsightsOnboardingStackSetsExecution

VendorInsightsOnboardingCFT スタックセットを作成する

VendorInsightsOnboardingCFT CloudFormation スタックセットを実行することで、必要な AWS のサービスを設定し、適切な IAM アクセス許可を設定します。これにより、AWS Marketplace Vendor Insights は で実行されている SaaS 製品のデータを収集 AWS アカウントし、Vendor Insights AWS Marketplace プロファイルに表示できます。

複数のアカウントソリューションがある場合、または個別の販売者と本番稼働用アカウントがある場合は、このスタックを複数のアカウントにデプロイする必要があります。StackSets では、前提条件スタックを作成した管理アカウントからこれを行うことができます。

スタックセットは、セルフマネージド型のアクセス許可を使用してデプロイされます。詳細については、AWS CloudFormation ユーザーガイドの[セルフマネージド型のアクセス許可を持つスタックセットの作成](#)を参照してください。

VendorInsightsOnboardingCFT スタックセットを作成するには

1. ウェブサイトの [AWS Samples Repo for Vendor Insights テンプレートフォルダ](#) から最新の VendorInsightsOnboardingCFT.yml ファイルを確認してダウンロードします GitHub。
2. AWS Marketplace 販売者アカウント AWS Management Console を使用して にサインインし、<https://console.aws.amazon.com/cloudformation> でコンソールを開きます AWS CloudFormation 。
3. CloudFormation コンソールのナビゲーションペインで、 の作成 StackSetを選択します。(ナビゲーションペインが表示されていない場合は、左上隅にあるナビゲーションペインを選択して展開します)。
4. アクセス許可 で、管理者ロールに IAM ロール名 を選択し、ドロップダウンからAWSVendorInsightsOnboardingStackSetsAdminロール名に を選択します。
5. [IAM 実行ロール名] として **AWSVendorInsightsOnboardingStackSetsExecution** を入力します。
6. [テンプレートの指定] で、[テンプレートファイルのアップロード] を選択します。ダウンロードした VendorInsightsOnboardingCFT.yml ファイルをアップロードするには、[ファイルを選択] を使用し、[次へ] を選択します。
7. 次の StackSet パラメータを指定し、次へ を選択します。
 - CreateVendorInsightsAutomatedAssessment – このパラメータは、 で AWS Audit Manager 自動評価を設定します AWS アカウント。管理アカウントと本稼働アカウントが別々にある場合、このオプションは運用アカウントでのみ選択し、管理アカウントには選択しないでください。

- `CreateVendorInsightsIAMRoles` – このパラメータは、AWS Marketplace Vendor Insights が 内の評価データを読み取ることを許可する IAM ロールをプロビジョニングします AWS アカウント。
 - `PrimaryRegion` - このパラメータは、SaaS デプロイメントのプライマリ AWS リージョンを設定します。これは、で S3 バケットが作成されるリージョンです AWS アカウント。SaaS 製品が 1 つのリージョンのみにデプロイされている場合、そのリージョンがプライマリリージョンです。
8. 必要に応じて `StackSet` オプションを設定します。[実行] 設定を [非アクティブ] のままにして、[次へ] を選択します。
 9. デプロイオプションを設定します。マルチアカウントソリューションを使用している場合は、1 回の操作で複数のアカウントとリージョンにスタックセットをデプロイするように設定できます。[次へ] をクリックします。

Note

マルチアカウントソリューションを使用している場合は、1 つのスタックセットとしてすべてのアカウントにデプロイすることはお勧めしません。ステップ 7 で定義したパラメータには細心の注意を払ってください。デプロイ先のアカウントの種類によっては、一部のパラメータを有効または無効にする必要がある場合があります。StackSets は、単一のデプロイで指定されたすべてのアカウントに同じパラメータを適用します。アカウントをスタックセットにグループ化することによりデプロイ時間を短縮できますが、マルチアカウントソリューションではやはり複数回デプロイする必要があります。

Important

複数のリージョンにデプロイする場合は、最初にリストするリージョンが `PrimaryRegion` である必要があります。[リージョンの同時実行] オプションはデフォルト設定の [シーケンシャル] のままにしておきます。

10. [確認] ページで選択内容を確認します。変更するには、変更する領域で [編集] を選択します。スタックセットを作成する前に、[能力] エリアの確認チェックボックスを選択する必要があります。

[送信] を選択します。

スタックセットはリージョンごとに約 5 分で完了します。

AWS Marketplace Vendor Insights プロファイルの表示

AWS Marketplace Vendor Insights のプロフィールは、購入者が製品を評価する際に使用する重要な情報を提供します。データ保護の目的で、AWS アカウント 認証情報を保護し、AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。このアプローチでは、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。IAM でのユーザーの作成方法の詳細については、「[the section called “グループの作成または使用”](#)」を参照してください。

購入者は Vendor AWS Marketplace Insights ダッシュボードを使用して製品を評価できます。そこでは、購入者はプロフィールに追加したデータソースによって定義された製品概要を確認できます。セキュリティプロフィールは、10 のカテゴリの複数のセキュリティコントロールによって定義されます。

データの定義に使用される 10 種類のセキュリティカテゴリは次のとおりです。

- アクセス管理
- アプリケーションセキュリティ
- 監査、コンプライアンス、セキュリティポリシー
- ビジネスの回復力
- データセキュリティとプライバシー
- エンドユーザー、デバイス、モバイルセキュリティ
- 人事
- インフラストラクチャセキュリティ
- リスク管理とインシデント対応
- セキュリティと設定に関するポリシー

詳細については、「AWS Marketplace購入者ガイド」の「[コントロールカテゴリを理解する](#)」を参照してください。

AWS Marketplace Vendor Insights をセットアップして使用することで、ユーザー情報を非公開かつ安全に保つための AWS のサービス 条件とデータプライバシールールを遵守することに同意したものとみなされます。AWS データプライバシー条件の詳細については、「[データプライバシーに関するよくある質問](#)」を参照してください。サービス条件の詳細については、「[AWS のサービス 条件](#)」を参照してください。

販売者としてのセキュリティプロファイルを確認する

自己評価を完了し、その他の実在する証拠を追加したら、販売者としてのプロフィールを確認することが重要です。プロフィールに追加された情報の確認が必要になります。

Note

Vendor AWS Marketplace Insights サポートチームに可視性の更新をリクエストするまで、このプロフィールは購入者には表示されません。サポートチームが更新を完了すると、秘密保持契約 (NDA) に署名した購入者はセキュリティプロフィールにアクセスできるようになります。

AWS Marketplace Vendor Insights プロファイルからサブスクライバーの個人を特定できる情報 (PII) データを削除する場合は、[AWS Support](#) に連絡してサポートケースを開始します。

購入者が表示するようにセキュリティプロフィールを表示するには

1. AWS Management Consoleにサインインします。
2. ポータルの [SaaS 製品](#) ページに移動します。
3. セキュリティプロフィールが関連付けられている製品を選択します。
4. [ベンダーインサイト] タブを選択し、[最新リリースのスナップショットを表示] を選択します。
5. [概要] タブには、アップロードしたすべての証明書バッジが表示されます。
6. [セキュリティとコンプライアンス] タブを選択すると、複数のコントロールから収集されたデータを表示できます。詳細を表示するには、各コントロールセットを選択します。

AWS Marketplace Vendor Insights でのスナップショットの管理

スナップショットは、セキュリティプロフィールの point-in-time 体制です。AWS Marketplace Vendor Insights では、スナップショットを使用して販売者の製品をいつでも評価できます。販売者は、さまざまな時点でのプロフィールのセキュリティ対策を比較したり、さまざまなセキュリティプロフィールの最新のスナップショットを比較したりして、意思決定の参考にすることができます。スナップショットは、データの鮮度やソースに関する透明性を提供するだけでなく、必要なセキュリティ情報を提供します。

AWS Marketplace コンソールの AWS Marketplace Vendor Insights スナップショットの概要セクションで、作成およびリリーススケジュールに関する次のスナップショットの詳細を表示できます。

- [最終作成スナップショット] - このプロファイル用に最後に作成されたスナップショット。
- [次回作成予定] - 次に作成される予定のスナップショット。
- [作成頻度] - スナップショットを作成する間隔またはスナップショットを作成する頻度。
- [次回の予定リリース] - 次にリリースされる予定のスナップショット。
- [ステージング時間] - スナップショットは少なくともこの期間ステージングされ、スナップショットリリースイベント中にリリース可能になります。
- [リリース頻度] - リリースイベント間の時間。

[スナップショットリスト] セクションでは、スナップショットのステータスは次のとおりです。

- [リリース済み] - スナップショットは公開されており、この製品に対する権限を持つユーザーが閲覧できます。
- [リリース待ち] - スナップショットは完成しているか、必須の最小ステージング期間に入っており、次のリリースが予定されています。
- [プライベート] - セキュリティプロファイルの有効化前に作成されたスナップショット、または検証エラーがあり、一般には公開されていないスナップショット。非公開のスナップショットは販売者にのみ表示されます。

トピック

- [スナップショットを作成する](#)
- [スナップショットを表示する](#)
- [スナップショットをエクスポートする](#)
- [最新リリースのスナップショットを表示する](#)
- [スナップショットリリースを延期する](#)
- [スナップショットリストの設定を変更する](#)

スナップショットを作成する

プロフィールのスナップショットを作成するには、以下の手順に従ってください。1日に最大20件のスナップショットを作成できます。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#) を開きます。
2. [ベンダーインサイト] を選択します。

3. [ベンダーインサイト] から製品を選択します。
4. 製品プロファイルページで、[スナップショット] リストに移動し、[新規スナップショットの作成] を選択します。
5. スナップショットスケジュールが変更されることを通知するメッセージが表示されます。[作成] を選択します。

Note

スナップショットスケジュールは、新しいスナップショットが作成されると変更されます。新しいスナップショットは、手動で作成したスナップショットと同じ時間にスケジュールされます。このメッセージには新しいスケジュールが含まれます。

新しいスナップショットは 30 分以内に作成され、スナップショットリストに追加されます。新しいスナップショットは [リリース保留中] ステータスで作成されます。ステータスが [リリース済み] に変わるまで、新しいスナップショットは表示できません。

スナップショットを表示する

プロフィールのスナップショットを表示するには、以下の手順に従ってください。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#) を開きます。
2. [ベンダーインサイト] を選択します。
3. [ベンダーインサイト] から製品を選択します。
4. 製品プロファイルページで、[スナップショットリスト] に移動し、表示するスナップショットの [スナップショット ID] を選択します。
5. 完了したら、[戻る] を選択してスナップショットビューを終了します。

スナップショットをエクスポートする

JSON 形式または CSV 形式にエクスポートできます。スナップショットをエクスポートするには、次の手順を実行します。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#) を開きます。
2. [ベンダーインサイト] を選択します。
3. [ベンダーインサイト] から製品を選択します。

4. 製品プロファイルページで、[スナップショットリスト] に移動し、エクスポートするスナップショットの [スナップショット ID] を選択します。
5. [エクスポート] をクリックします。
6. ドロップダウンリストから [ダウンロード (JSON)] または [ダウンロード (CSV)] を選択します。

最新リリースのスナップショットを表示する

最新リリースのスナップショットは、ユーザーが製品の状態を確認したり評価したりするために使用するものです。製品を正確な情報で表現するには、最新リリースのスナップショットの内容を知ることが重要です。プロフィールの最新のスナップショットを表示するには、次の手順を実行します。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#) を開きます。
2. [ベンダーインサイト] を選択します。
3. [ベンダーインサイト] から製品を選択します。
4. 製品プロファイルページで、[スナップショットリスト] に移動し、表示するスナップショットの [スナップショット ID] を選択します。
5. [最新リリースのスナップショットを表示] を選択します。
6. 完了したら、[戻る] を選択してスナップショットビューを終了します。

スナップショットリリースを延期する

プロフィールへのスナップショットのリリースを遅らせるには、特定の [スナップショット ID] のスナップショットリリースを延期できます。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#) を開きます。
2. [ベンダーインサイト] を選択します。
3. [ベンダーインサイト] から製品を選択します。
4. 製品プロファイルページで、[スナップショットリスト] に移動し、リリースを延期するスナップショットの [スナップショット ID] を選択します。
5. [スナップショットの概要] から [スナップショットリリースを延期] を選択します。
6. スナップショットスケジュールが変更されることを通知するメッセージが表示されます。[延期] を選択します。

この製品のスナップショットリリースを正常に延期したことを示す成功メッセージが表示されます。

スナップショットリストの設定を変更する

スナップショットを作成したら、[スナップショットリスト]でのスナップショットの表示方法の設定を変更できます。

1. にサインイン AWS Management Console し、[AWS Marketplace コンソール](#)を開きます。
2. [ベンダーインサイト]を選択します。
3. [ベンダーインサイト]から製品を選択します。
4. 製品プロファイルページで、[スナップショットリスト]に移動し、変更するスナップショットの [スナップショット ID]を選択します。
5. 設定アイコンを選択します。スナップショットに関する以下の設定をカスタマイズできます。
 - [ページサイズ] - 各ページに表示するスナップショットの数を、1 ページあたり [10 リソース]、[20 リソース]、または [50 リソース]の中から選択します。
 - [行を折り返す] - 行を折り返してレコード全体を表示するオプションを選択します。
 - [時間フォーマット] - [絶対]、[相対]、[ISO]のいずれかを選択します。
 - [表示列] - スナップショットの詳細に表示するオプション ([スナップショット ID]、[ステータス]、[作成日])を選択します。

AWS Marketplace Vendor Insights でのアクセスの制御

AWS Identity and Access Management (IAM) は AWS のサービス、AWS リソースへのアクセスを制御するのに役立ちます。IAM は、追加料金なしで AWS のサービスを使用できます。管理者は、誰を認証 (サインイン) し、誰に リソースの使用 AWS Marketplace を承認する (アクセス許可を付与する)かを制御します。AWS Marketplace Vendor Insights は IAM を使用して、販売者のデータ、評価、販売者の自己認証、および業界標準の監査レポートへのアクセスを制御します。

で誰が何をできるかを制御するための推奨方法は AWS Marketplace 管理ポータル、IAM を使用してユーザーとグループを作成することです。次に、そのユーザーをグループに追加してグループを管理します。読み取り専用のアクセス許可を付与するポリシーまたはアクセス許可をグループに割り当てます。読み取り専用アクセスが必要なユーザーが他にもいる場合は、ユーザーのアクセス許可を追加するのではなく、作成したグループにそれらのユーザーを追加します。

ポリシーは、ユーザー、グループ、またはロールに適用されるアクセス許可を定義したドキュメントです。AWSでユーザーができることを決定するアクセス許可。ポリシーでは通常、特定のアクションへのアクセスを許可し、オプションで、それらのアクションを Amazon EC2 インスタンスや

Amazon S3 バケットなどの特定のリソースで実行することを許可することができます。また、ポリシーにより、アクセスを明示的に拒否することもできます。アクセス許可は、特定のリソースへのアクセスを許可または拒否するポリシー内のステートメントです。

Important

作成したすべてのユーザーは、自分の認証情報を使用して認証します。ただし、同じ AWS アカウントを使用します。ユーザーが行った変更はアカウント全体に影響を及ぼす可能性があります。

AWS Marketplace には、それらのアクセス許可を持つユーザーがで実行できるアクションを制御するためのアクセス許可が定義されています AWS Marketplace 管理ポータル。また、複数のアクセス許可を組み合わせた `AWSMarketplaceSellerProductsFullAccess` ポリシーでもあります。AWSMarketplaceSellerProductsFullAccess ポリシーでは、ユーザーに AWS Marketplace 管理ポータルの製品へのフルアクセスを許可します。

使用可能なアクション、リソース、および条件キーの詳細については、「[サービス認証リファレンス](#)」の [AWS Marketplace 「Vendor Insights のアクション、リソース、および条件キー」](#) を参照してください。

AWS Marketplace Vendor Insights 販売者のアクセス許可

AWS Marketplace Vendor Insights の IAM ポリシーでは、次のアクセス許可を使用できます。アクセス許可を 1 つの IAM ポリシーにまとめて、必要なアクセス許可を付与できます。

CreateDataSource

CreateDataSource では、ユーザーは新しいデータソースリソースを作成できます。サポートされているデータソースは以下の通りです。

- SOC2Type2AuditReport
- ISO27001AuditReport
- AWSAuditManagerSecurityAutomatedAssessment
- FedRAMPCertification
- GDPRComplianceReport
- HIPAAComplianceReport
- PCIDSSAuditReport

- SecuritySelfAssessment

アクショングループ: 読み取り/書き込み

必要なリソース: なし

リソースの作成: DataSource

DeleteDataSource

DeleteDataSource では、ユーザーは所有するデータソースを削除できます。削除するには、データソースとプロファイルの関連付けを解除する必要があります。詳細については、「[the section called “AssociateDataSource”](#)」を参照してください。

アクショングループ: 読み取り/書き込み

必要なリソース: DataSource

GetDataSource

GetDataSource では、ユーザーがデータソースの詳細を取得できます。データソースの詳細には、関連するタイムスタンプ、元の作成パラメータ、処理情報 (ある場合) などのメタデータ情報が含まれます。

アクショングループ: 読み取り専用、読み取り/書き込み

必要なリソース: DataSource

UpdateDataSource

UpdateDataSource では、ユーザーがデータソースの詳細を更新できます。詳細には、名前やソース情報 (ロール、ソース Amazon リソースネーム (ARN)、ソースコンテンツなど) などのメタデータ情報が含まれます。

アクショングループ: 読み取り専用、読み取り/書き込み

必要なリソース: DataSource

ListDataSources

ListDataSources では、ユーザーが所有するデータソースを一覧表示できます。

アクショングループ: 読み取り専用、読み取り/書き込み、リスト専用

必要なリソース: なし

CreateSecurityProfile

CreateSecurityProfile では、ユーザーが新しいセキュリティプロファイルを作成できます。セキュリティプロファイルは、スナップショットをいつどのように生成するかを管理するためのリソースです。また、ユーザーはプロファイルのステータスや適用条件を管理して、購入者がスナップショットにアクセスする方法を制御することもできます。

アクショングループ: 読み取り専用、読み取り/書き込み、リスト専用

必要なリソース: なし

リソースの作成: SecurityProfile

ListSecurityProfiles

ListSecurityProfiles では、ユーザーが所有するセキュリティプロファイルを一覧表示できます。

アクショングループ: 読み取り専用、読み取り/書き込み、リスト専用

必要なリソース: なし

GetSecurityProfile

CreateSecurityProfile では、ユーザーはセキュリティプロファイルの詳細を取得できます。

アクショングループ: 読み取り専用、読み取り/書き込み

必要なリソース: SecurityProfile

AssociateDataSource

AssociateDataSource では、ユーザーは既存のを AWS Marketplace Vendor Insights DataSource プロファイルに関連付けることができます。ユーザーは、データソースをプロファイルに関連付けたり関連付けを解除したりすることにより、スナップショットの内容を制御できます。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile および DataSource

DisassociateDataSource

DisassociateDataSource では、ユーザーは既存の と AWS Marketplace Vendor Insights プロファイルDataSourceの関連付けを解除できます。ユーザーは、データソースをプロファイルに関連付けたり関連付けを解除したりすることにより、スナップショットの内容を制御できます。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile および DataSource

UpdateSecurityProfile

UpdateSecurityProfile では、ユーザーは名前や説明などのセキュリティプロファイル属性を変更できます。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile

ActivateSecurityProfile

ActivateSecurityProfile では、ユーザーがセキュリティプロファイルの Active ステータスを設定できます。セキュリティプロファイルを有効にすると、他の条件が満たされた場合に公開できる Staged 状態で新しいスナップショットを作成できます。詳細については、「[the section called "UpdateSecurityProfileSnapshotReleaseConfiguration"](#)」を参照してください。

少なくとも 1 つの Released スナップショットを持つ Active セキュリティプロファイルは、エンドユーザーの AWS Marketplace Vendor Insights 検出の対象となります。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile

DeactivateSecurityProfile

DeactivateSecurityProfile では、ユーザーがセキュリティプロファイルの Inactive ステータスを設定できます。セキュリティプロファイルのこの終了状態は、プロファイルを共有状態から削除することと同じです。ユーザーがセキュリティプロファイルを非アクティブ化できるのは、プロファイルにアクティブなサブスクライバーがない場合のみです。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile

UpdateSecurityProfileSnapshotCreationConfiguration

UpdateSecurityProfileSnapshotCreationConfiguration では、ユーザーはスナップショット作成設定のカスタムスケジュールを定義できます。週次作成のデフォルト作成設定は、このアクションで上書きできます。

ユーザーはこのアクションを使用して、スケジュールをキャンセルしたり、スケジュールを将来の日付に延期したり、新しいスナップショット作成を以前に開始したりするなど、スケジュールを変更できます。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile

UpdateSecurityProfileSnapshotReleaseConfiguration

UpdateSecurityProfileSnapshotReleaseConfiguration では、ユーザーはスナップショットリリース設定のカスタムスケジュールを定義できます。毎週リリースされ、ステージング期間が 2 日でレビュー対象のデフォルト作成設定は、このアクションで上書きできます。

ユーザーはこのアクションを使用して、スケジュールをキャンセルしたり、スケジュールを将来の日付に延期したりするなど、スケジュールを変更できます。

アクショングループ: 読み取り/書き込み

必要なリソース: SecurityProfile

ListSecurityProfileSnapshots

ListSecurityProfileSnapshots では、ユーザーが所有するセキュリティプロファイルのスナップショットを一覧表示できます。

アクショングループ: 読み取り専用、リスト専用、読み取り/書き込み。

必要なリソース: SecurityProfile

GetSecurityProfileSnapshot

GetSecurityProfileSnapshot では、ユーザーが所有するセキュリティプロファイルのスナップショットを取得できます。

アクショングループ: 読み取り専用、読み取り/書き込み

必要なリソース: SecurityProfile

TagResource

TagResource では、ユーザーがリソースに新しいタグを追加できます。サポートされているリソースは SecurityProfile と DataSource です。

アクショングループ: タグ付け

オプションリソース: SecurityProfile と DataSource

UntagResource

UntagResource では、ユーザーはリソースからタグを削除できます。サポートされているリソースは SecurityProfile と DataSource です。

アクショングループ: タグ付け

オプションリソース: SecurityProfile と DataSource

ListTagsForResource

ListTagsForResource では、ユーザーはリソースのリソースタグを一覧表示できます。サポートされているリソースは SecurityProfile と DataSource です。

アクショングループ: 読み取り専用

オプションリソース: SecurityProfile と DataSource

追加リソース

「IAM ユーザーガイド」の以下のリソースには、IAM の始め方と使用に関する詳細情報が記載されています。

- [IAM でのセキュリティのベストプラクティス](#)
- [IAM ポリシーを管理する](#)
- [ユーザーグループへのポリシーのアタッチ](#)
- [IAM ID \(ユーザー、ユーザーグループ、ロール\)](#)

- [最初のユーザーおよびユーザーグループの作成](#)
- [IAM ポリシーを管理する](#)
- [ポリシーを使用した AWS リソースへのアクセスの制御](#)

AWS Marketplace セキュリティ

のクラウドセキュリティが最優先事項 AWS です。AWS のお客様は、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャからメリットを得られます。

セキュリティは、AWS とユーザーの間で共有される責任です。[責任共有モデル](#)では、これをクラウドのセキュリティおよびクラウド内のセキュリティとして説明しています。

- クラウドのセキュリティ — AWS クラウドで AWS サービスを実行するインフラストラクチャを保護する責任 AWS を担います。AWS また、は、お客様が安全に使用できるサービスも提供します。セキュリティの有効性は、[AWS コンプライアンスプログラム](#)の一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。に適用されるコンプライアンスプログラムの詳細については AWS Marketplace、「[コンプライアンスAWS プログラムによる対象範囲内のサービス](#)」を参照してください。
- クラウドのセキュリティ — お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、お客様のデータの機密性、組織の要件、および適用可能な法律および規制などの他の要因についても責任を担います。

このドキュメントは、を使用する際の責任共有モデルの適用方法を理解するのに役立ちます AWS Marketplace。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するためにへのアクセス AWS Identity and Access Management を管理する AWS Marketplace ようにを設定する方法を示します。また、リソースのモニタリングや保護 AWS Marketplace に役立つ他の AWS のサービスの使用方法についても説明します。

で提供する製品に関するセキュリティおよびその他のポリシーの詳細については AWS Marketplace、以下のトピックを参照してください。

- [AMI ベースの製品要件](#)
- [コンテナベースの製品要件](#)
- [SaaS 製品ガイドライン](#)
- [プロフェッショナルサービス製品の要件](#)

Note

データ製品の AWS Data Exchange のセキュリティについては、「AWS Data Exchange ユーザーガイド」の「[セキュリティ](#)」を参照してください。

で購入者のセキュリティの詳細については AWS Marketplace、「[購入者ガイド](#)」の「[のセキュリティ AWS Marketplace](#)」を参照してください。AWS Marketplace

トピック

- [AWS Marketplace 管理ポータルへのアクセスのコントロール](#)
- [AWS Marketplace 販売者のポリシーとアクセス許可](#)
- [AWS 販売者向けの AWS Marketplace マネージドポリシー](#)
- [AWS Marketplace Commerce Analytics Service アカウントのアクセス許可](#)
- [Amazon SQS のアクセス許可](#)
- [AWS Marketplace 計測と使用権限 API のアクセス許可](#)
- [のサービスにリンクされたロールの使用 AWS Marketplace](#)
- [を使用した AWS Marketplace Metering API コールのログ記録 AWS CloudTrail](#)

AWS Marketplace 管理ポータルへのアクセスのコントロール

AWS Identity and Access Management (IAM) は AWS のサービス、AWS リソースへのアクセスを制御するのに役立ちます。管理者は、誰を認証 (サインイン) し、誰に AWS Marketplace リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで AWS のサービスを使用できます。

で誰が何をできるかを制御するための推奨方法は AWS Marketplace 管理ポータル、IAM を使用してユーザーとグループを作成することです。次に、そのユーザーをグループに追加してグループを管理します。例えば、製品の表示を John に許可する場合は、専用のユーザーを作成し、そのユーザーを読み取り専用アクセス用に作成したグループに追加します。読み取り専用のアクセス許可を付与するポリシーまたはアクセス許可をグループに割り当てます。読み取り専用アクセスが必要なユーザーが他にもいる場合は、アクセス許可をユーザーに許可するのではなく、作成したグループにそれらのユーザーを追加します。例えば、John のロールが変更になり、読み取り専用アクセスが不要になった場合は、John をそのグループから削除します。

ポリシーは、ユーザー、グループ、またはロールに適用されるアクセス許可を定義したドキュメントです。また、AWSでユーザーに許可するアクションをアクセス許可で指定します。ポリシーでは通

常、特定のアクションへのアクセスを許可し、オプションで、それらのアクションを Amazon EC2 インスタンスや Amazon S3 バケットなどの特定のリソースで実行することを許可することができます。また、ポリシーにより、アクセスを明示的に拒否することもできます。アクセス許可は、特定のリソースへのアクセスを許可または拒否するポリシー内のステートメントです。許可は、「A は C に対して B を実行する許可を持つ」という形で提示することができます。例えば、Jane (A) は John の Amazon Simple Queue Service キュー (C) からメッセージ (B) を読む許可がある、などです。Jane が John のキューを使用するリクエストを Amazon SQS に送信するたびに、サービスは Jane にアクセス許可があるかどうか確認します。さらに、アクセス許可に指定された条件をリクエストが満たしているかどうか確認します。

Important

作成したすべてのユーザーは、自分の認証情報を使用して認証します。ただし、同じ AWS アカウントを使用します。ユーザーが行った変更はアカウント全体に影響を及ぼす可能性があります。

AWS Marketplace には、それらのアクセス許可を持つユーザーがで実行できるアクションを制御するためのアクセス許可が定義されています AWS Marketplace 管理ポータル。また、複数のアクセス許可を組み合わせたポリシー AWS Marketplace を作成して管理することもできます。

IAM の使用開始と使用の詳細については、以下のリソースを参照してください。

- [管理ユーザーを作成する](#)
- [IAM でのセキュリティのベストプラクティス](#)
- [IAM ポリシーを管理する](#)
- [IAM ユーザーグループへのポリシーのアタッチ](#)
- [IAM ID \(ユーザー、グループ、ロール\)](#)
- [ポリシーを使用した AWS リソースへのアクセスの制御](#)

ユーザーとグループを作成し、ユーザーとしてサインインするためのいくつかのガイダンスを以下のトピックに示します。

トピック

- [ユーザーの作成](#)
- [グループの作成または使用](#)

• ユーザーとしてのサインイン

ユーザーの作成

社内のユーザーが にサインインできるようにするには AWS Marketplace 管理ポータル、アクセスが必要なユーザーごとにユーザーを作成します。

ユーザーを作成するには

1. にサインイン AWS Management Console し、 <https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. ナビゲーションペインで [ユーザー] を選択してから、[新規ユーザーの作成] を選択します。
3. 番号付きのテキストボックスに、作成する各ユーザーの名前を入力します。
4. [ユーザーごとにアクセスキーを作成] のチェックボックスをオフにし、[作成] を選択します。

作成したばかりの各ユーザーにパスワードを割り当てるには

1. ユーザーのリストで、新しいユーザーの名前を選択します。
2. [セキュリティ認証情報] タブを選択し、[パスワードの管理] を選択します。
3. 自動生成パスワードまたはカスタムパスワードのいずれかのオプションを選択します。次回のサインインで新しいパスワードの選択をユーザーに要求する場合は、[次回のサインインで新しいパスワードの作成をユーザーに要求する] チェックボックスをオンにします。[適用] を選択します。
4. [認証情報をダウンロード] を選択して、サインイン認証情報とアカウント固有のサインイン URL をコンピュータ上のカンマ区切り値 (CSV) ファイルに保存します。次に、[Close] (閉じる) を選択します。

Note

作成したばかりのサインイン認証情報を使用してサインインするには、ユーザーはアカウント固有のサインイン URL に移動する必要があります。この URL はダウンロードしたばかりの認証情報ファイルにあり、IAM コンソールでも確認できます。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーが AWS アカウントにサインインする方法](#)」を参照してください。

i Tip

自分が AWS アカウント 所有者であっても、自分用のサインイン認証情報も作成してください。これは、アカウント所有者であっても、ユーザーとして AWS Marketplace で操作を行うために全員に推奨されるベストプラクティスです。管理者権限を持つユーザーを自分で作成する方法については、「IAM ユーザーガイド」の「[管理ユーザーを作成する](#)」を参照してください。

グループの作成または使用

ユーザーを作成したら、グループを作成し、AWS Marketplace 管理ポータルでページにアクセスするアクセス許可を作成します。次に、そのアクセス許可をグループに追加し、ユーザーをグループに追加します。

グループにアクセス許可を割り当てると、そのグループのすべてのメンバーに、特定のアクションの実行が許可されます。新しいユーザーをグループに追加すると、そのユーザーはグループに割り当てられたアクセス許可を自動的に取得します。グループは複数のアクションのアクセス許可を持つことができます。独自のポリシーを作成せずに [AWS Marketplace 管理ポリシー](#) を使用することをお勧めします。

の 管理ポリシー AWS Marketplace をグループに割り当てるには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで [グループ] を選択後、ポリシーをアタッチするグループを選択します。
3. グループの概要ページの [アクセス許可] タブで、[Attach Policy (ポリシーのアタッチ)] を選択します。
4. [ポリシーのアタッチ] ページで、[Filter:] の横に、「awsmarketplace」と入力します。
5. アタッチするポリシーを選択し、[ポリシーのアタッチ] を選択します。

アクセス AWS Marketplace 管理ポータル 許可を持つポリシーを作成するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで、[Policies (ポリシー)] を選択し、次に [Create Policy (ポリシーの作成)] を選択します。

3. [Policy Generator] の横の [Select] をクリックします。
4. [Edit Permissions] ページで、以下を実行します。
 - a. [Effect] で、[Allow] を選択します。
 - b. [AWS Service (サービス)] で、[AWS Marketplace 管理ポータル] を選択します。
 - c. [Actions (アクション)] で、付与するアクセス許可を選択します。
 - d. [Add Statement (ステートメントの追加)] を選択します。
 - e. [Next Step (次のステップ)] をクリックします。
5. [Review Policy (ポリシーのレビュー)] ページで、以下の作業を行います。
 - a. [Policy Name (ポリシー名)] にこのポリシーの名前を入力します。ポリシー名をメモします。後のステップで必要になります。
 - b. (オプション) [Description (説明)] に、ポリシーの説明を入力します。
 - c. [ポリシーの作成] を選択します。

適切なアクセス許可を持つ IAM グループを作成し、そのグループにユーザーを追加するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで、[Groups]、[Create New Group] の順に選択します。
3. [Group Name: (グループ名:)] に、グループの名前を入力します。その後、[Next Step] (次のステップ) を選択します。
4. [Attach Policy (ポリシーのアタッチ)] ページで、次の操作を実行します。
 - a. [Filter: (フィルター:)] で、[Customer Managed Policies (顧客が管理するポリシー)] を選択します。
 - b. このグループにアタッチするポリシーの名前の横にあるチェックボックスをオンにします。通常、これは先ほど作成したポリシーになります。
 - c. [Next Step (次のステップ)] をクリックします。
5. グループを作成 を選択します。
6. [Groups (グループ)] リストで新しいグループを見つけ、その横にあるチェックボックスをオンにします。[Group Actions (グループアクション)] で、[Add Users to Group (ユーザーをグループに追加)] を選択します。
7. グループに追加する各ユーザーの横にあるチェックボックスをオンにして、[Add Users (ユーザーの追加)] を選択します。

ユーザーとしてのサインイン

IAM でユーザーを作成すると、ユーザーは自分のサインイン認証情報を使用してサインインできます。これを行うには、AWS アカウントに関連付けられている一意の URL を使用する必要があります。サインイン URL を取得し、ユーザーに配信できます。

アカウント固有のサインイン URL を取得するには

1. <https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. ナビゲーションペインで、[ダッシュボード] を選択します。
3. コンテンツペインの上部にある [IAM ユーザーのサインインリンク:]を探し、次のような形式のサインインリンクをメモします。

```
https://AWS_account_ID.signin.aws.amazon.com/console/
```

Note

サインインページの URL に AWS アカウント ID ではなく会社名 (またはその他のわかりやすい識別子) を含む場合は、[をカスタマイズ](#) を選択してアカウントのエイリアスを作成できます。詳細については、「IAM [ユーザーガイド](#)」の AWS 「[アカウント ID とそのエイリアス](#)」を参照してください。

4. この URL を、それぞれに作成したサインイン認証情報とともに AWS Marketplace、を使用できる社内のユーザーに配布します。AWS Marketplaceにアクセスする前に、アカウント固有のサインイン URL を使用してサインインするように指示してください。

AWS Marketplace 販売者のポリシーとアクセス許可

AWS Marketplace には、で使用できる 3 つの管理ポリシーがあります AWS Marketplace 管理ポータル。さらに、個々のアクセス許可を使用して独自の AWS Identity and Access Management (IAM) ポリシーを作成できます。

AWS Marketplace には、で使用できる管理ポリシーがいくつかあります AWS Marketplace 管理ポータル。さらに、個々のアクセス許可を使用して独自の AWS Identity and Access Management (IAM) ポリシーを作成できます。

設定、お問い合わせ、ファイルのアップロード、インサイト タブ AWS Marketplace 管理ポータルへのきめ細かなアクセスを提供することもできます。きめ細かなアクセスを利用すると、次のことが可能になります。

- パスワードやアクセスキーを共有しなくても、お客様の AWS アカウント アカウントのリソースを管理および使用するためのアクセス許可を他の人に付与します。
- さまざまなリソースに対する細やかな権限を複数のユーザーに付与します。例えば、一部のユーザーに AWS Marketplace 管理ポータルの [設定] タブを表示することを許可し、他のユーザーには [設定] タブと [お問い合わせ] タブで編集することを許可できます。

Note

データ製品のための AWS Data Exchange のポリシーと権限の詳細については、「AWS Data Exchange ユーザーガイド」の「[Identity and Access Management in AWS Data Exchange](#)」を参照してください。

AWS Marketplace 購入者のポリシーとアクセス許可の詳細については、「AWS Marketplace 購入者ガイド」の[AWS Marketplace 「サブスクリプションへのアクセスの制御」](#)を参照してください。

AWS Marketplace 販売者向けポリシー

次の管理ポリシーを使用して、AWS Marketplace 管理ポータルへの制御されたアクセスをユーザーに付与することができます。

AWSMarketplaceSellerFullAccess

Amazon マシンイメージ (AMI) 管理など、AWS Marketplace 管理ポータル および他の AWS サービス内のすべてのページへのフルアクセスを許可します。

AWSMarketplaceSellerProductsFullAccess

AWS Marketplace 管理ポータルの[製品](#)ページへのフルアクセスを許可します。

AWSMarketplaceSellerProductsReadOnly

AWS Marketplace 管理ポータルの[製品](#)ページへの読み取り専用アクセスを許可します。

⚠ Important

AWS Marketplace 購入者は、管理ポリシーを使用して、購入するサブスクリプションを管理できます。で使用する マネージドポリシーの名前は、で AWS Marketplace 管理ポータル 始まりますAWSMarketplaceSeller。IAM でポリシーを検索するには、必ず先頭がAWSMarketplaceSeller となっているポリシー名を検索してください。

AWS Marketplace は、特定のシナリオに特化した マネージドポリシーも提供します。AWS Marketplace 販売者向けの AWS 管理ポリシーの完全なリストと、それらが提供するアクセス許可の説明については、「」を参照してください[AWS 販売者向けの AWS Marketplace マネージドポリシー](#)。

AWS Marketplace 販売者のアクセス許可

以下のアクセス許可は、AWS Marketplace 管理ポータルの IAM ポリシーで使用できます。

aws-marketplace-management:PutSellerVerificationDetails

本人確認 (KYC) プロセスを開始するためのアクセスを許可します。

aws-marketplace-management:GetSellerVerificationDetails

AWS Marketplace 管理ポータルの KYC ステータスを表示するためのアクセスを許可します。

aws-marketplace-management:PutBankAccountVerificationDetails

[銀行口座検証](#)プロセスを開始するためのアクセスを許可します。

aws-marketplace-management:GetBankAccountVerificationDetails

AWS Marketplace 管理ポータルに銀行口座検証ステータスを表示するためのアクセスを許可します。

aws-marketplace-management:PutSecondaryUserVerificationDetails

にセカンダリユーザーを追加するアクセスを許可します AWS Marketplace 管理ポータル。

aws-marketplace-management:GetSecondaryUserVerificationDetails

AWS Marketplace 管理ポータルにセカンダリユーザーステータスを表示するアクセスを許可します。

aws-marketplace-management:GetAdditionalSellerNotificationRecipients

AWS Marketplace 通知の E メール連絡先を表示するアクセスを許可します。

aws-marketplace-management:PutAdditionalSellerNotificationRecipients

AWS Marketplace 通知の E メール連絡先を更新するためのアクセスを許可します。

tax:PutTaxInterview

AWS Marketplace 管理ポータルで [Tax Interview](#) を受けるアクセスを許可します。

tax:GetTaxInterview

AWS Marketplace 管理ポータルで Tax Interview ステータスを表示するアクセスを許可します。

tax:GetTaxInfoReportingDocument

AWS Marketplace 販売者が Tax ダッシュボードから税務書類 (1099-K フォームなど) を表示およびダウンロードすることを許可する

payments:CreatePaymentInstrument

に銀行口座を追加するアクセスを許可します AWS Marketplace 管理ポータル。

payments:GetPaymentInstrument

で既存の銀行口座を表示するアクセスを許可します AWS Marketplace 管理ポータル。

aws-marketplace:ListTasks

販売者のアクションが保留になっているタスクのリストを表示するアクセスを許可します。

aws-marketplace:DescribeTask

販売者のアクションが保留になっているタスクの詳細を表示するアクセスを許可します。

aws-marketplace:UpdateTask

販売者のアクションが保留になっているタスクを編集するアクセスを許可します。

aws-marketplace:CompleteTask

販売者のアクションが保留になっているタスクに対する編集を送信するアクセスを許可します。

support:CreateCase

内で AWS Marketplace ケースを作成するためのアクセスを許可します AWS Marketplace 管理ポータル。

aws-marketplace-management:viewSupport

AWS Marketplace 管理ポータルの [\[カスタマーサポート利用資格\]](#) ページへのアクセスを許可します。

aws-marketplace-management:viewReports

AWS Marketplace 管理ポータルの [\[レポート\]](#) ページへのアクセスを許可します。

aws-marketplace:ListEntities

でオブジェクトを一覧表示するためのアクセスを許可します AWS Marketplace 管理ポータル。AWS Marketplace 管理ポータル[\[ファイルのアップロード\]](#)、[\[オファー\]](#)、[\[パートナー\]](#) ページにアクセスするために必要です。

Note

[設定] タブを表示するためのアクセスを許可するには、この権限、ListEntity 権限、および Amazon リソースネーム (ARN) `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}` を使用できます。

aws-marketplace:DescribeEntity

でオブジェクトの詳細を表示するアクセスを許可します AWS Marketplace 管理ポータル。AWS Marketplace 管理ポータル[\[ファイルのアップロード\]](#)、[\[オファー\]](#)、[\[パートナー\]](#)、[\[契約\]](#) ページにアクセスするために必要です。

Note

[設定] タブを表示するためのアクセスを許可するには、この権限、DescribeEntity 権限、および ARN `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/*` を使用できます。

aws-marketplace:StartChangeSet

で製品変更を作成するためのアクセスを許可します AWS Marketplace 管理ポータル。AWS Marketplace 管理ポータル[\[ファイルのアップロード\]](#)、[\[オファー\]](#)、[\[パートナー\]](#)、[\[契約\]](#) ページに変更を加えるために必要です。

Note

で販売者として登録するためのアクセスを許可するには AWS Marketplace、このアクセス許可、`catalog:ChangeType: "CreateSeller"`条件キー、および次の ARN

を使用できます: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}`。

で販売者プロフィールを更新するためのアクセスを許可するには AWS Marketplace、このアクセス許可、`catalog:ChangeType: "UpdateInformation"` 条件キー、および次の ARN を使用できます: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}`。

Amazon Web Services の支払い設定を更新するためのアクセスを許可するには、この権限、`catalog:ChangeType: "UpdateDisbursementPreferences"` 条件キー、および ARN `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}` を使用できます。

aws-marketplace:SearchAgreements

[[契約](#)] ページでの契約の概要リストの表示、 [[パートナー](#)] ページでの ISV とチャネルパートナー間の機会の表示を許可します。

aws-marketplace:DescribeAgreement

[[契約](#)] ページでの契約詳細の概要の表示、 [[パートナー](#)] ページでの ISV とチャネルパートナー間の機会の表示を許可します。

aws-marketplace:GetAgreementTerms

[[契約](#)] ページでのすべての契約条件の詳細の表示、 [[パートナー](#)] ページでの ISV とチャネルパートナー間の機会の表示を許可します。

aws-marketplace:GetSellerDashboard

AWS Marketplace 管理ポータル の [[インサイト](#)] ページのダッシュボードへのアクセスを許可します。

Note

[[製品の管理](#)] ページへのアクセスをユーザーに許可するには、マネージドアクセス許可 `AWSMarketplaceSellerProductsFullAccess` または `AWSMarketplaceSellerProductsReadOnly` のいずれかを使用する必要があります。

前のアクセス許可を1つのIAMポリシーにまとめて、必要なアクセス許可を付与することができます。次の例を参照してください。

例 1: KYC ステータスを表示するアクセス許可

で KYC ステータスを表示するアクセス許可を付与するには AWS Marketplace 管理ポータル、次の例のようなポリシーを使用します。

で KYC ステータスを表示するアクセス許可を付与するには AWS Marketplace 管理ポータル、次の例のようなポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace-management:GetSellerVerificationDetails"
      ],
      "Resource": ["*"]
    }
  ]
}
```

例 2: プライベートオファーのアップグレードと更新を作成するためのアクセス許可

[契約] ページを表示および使用してプライベートオファーのアップグレードと更新を作成するためのアクセス許可を付与するには、次の例に示すようなポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:GetAgreementTerms",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "StringEquals": {

```

```

        "aws-marketplace:PartyType": "Proposer"
      },
      "ForAllValues:StringEquals": {
        "aws-marketplace:AgreementType": [
          "PurchaseAgreement"
        ]
      }
    }
  ]
}

```

例 3: オファーページにアクセスして新しいプライベートオファーを作成する権限

[オファー] ページを表示および使用して既存のプライベートオファーを表示しプライベートオファーを作成するためのアクセス許可を付与するには、次の例に示すようなポリシーを使用します。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}

```

例 4: 設定ページにアクセスする権限

[設定] ページを表示して使用するアクセス許可を付与するには、次の例のようなポリシーを使用します。

```

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

```

```
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
    ],
    "Effect": "Allow",
    "Resource": "arn:{partition}:{aws-marketplace}:{region}:{account-
id}:AWSMarketplace/Seller/*",
    }
]
}
```

例 5: ファイルのアップロードページにアクセスする権限

[ファイルをアップロード] ページを表示して使用するアクセス許可を付与するには、次の例のようなポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

IAM グループの使用

または、AWS Marketplace 管理ポータル内の各個別のページへのアクセス権限を付与する、個別の IAM グループを作成できます。ユーザーは複数のグループに属することができます。そのため、ユーザーが複数のページにアクセスする必要がある場合、適切なすべてのグループにユーザーを追加できます。例えば、1つの IAM グループを作成し、そのグループに [インサイト] ページにアクセスする権限を付与します。次に、別のグループを作成し、そのグループに [ファイルをアップロード] ページにアクセスする権限を付与するなどの操作ができます。ユーザーが [インサイト] ページと [ファイルをアップロード] ページの両方にアクセスするための権限が必要な場合は、ユーザーを両方のグループに追加します。

ユーザーとグループの詳細については、「IAM ユーザーガイド」の「[IAM アイデンティティ \(ユーザー、グループ、ロール\)](#)」を参照してください。

AWS 販売者向けの AWS Marketplace マネージドポリシー

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンポリシーです AWS。AWS 管理ポリシーは、多くの一般的なユースケースにアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケース別に[カスタマー マネージドポリシー](#)を定義して、マネージドポリシーを絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、その更新はポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。は、新しい AWS のサービスが起動されるか、既存のサービスで新しい API AWS オペレーションが使用可能になると、AWS 管理ポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

このセクションには、販売者の AWS Marketplace へのアクセスを管理するために使用される各ポリシーが記載されています。購入者ポリシーの詳細については、「[AWS Marketplace 購入者ガイド](#)」の「[購入者向けの管理ポリシー](#)」を参照してください。AWS Marketplace

トピック

- [AWS マネージドポリシー: AWSMarketplaceAmiIngestion](#)
- [AWS マネージドポリシー: AWSMarketplaceFullAccess](#)
- [AWS マネージドポリシー: AWSMarketplaceGetEntitlements](#)
- [AWS マネージドポリシー: AWSMarketplaceMeteringFullAccess](#)
- [AWS マネージドポリシー: AWSMarketplaceMeteringRegisterUsage](#)
- [AWS マネージドポリシー: AWSMarketplaceSellerFullAccess](#)
- [AWS マネージドポリシー: AWSMarketplaceSellerProductsFullAccess](#)
- [AWS マネージドポリシー: AWSMarketplaceSellerProductsReadOnly](#)
- [AWS マネージドポリシー: AWSVendorInsightsVendorFullAccess](#)
- [AWS マネージドポリシー: AWSVendorInsightsVendorReadOnly](#)
- [AWS Marketplace AWS 管理ポリシーの更新](#)

AWS マネージドポリシー: **AWSMarketplaceAmiIngestion**

このポリシーを使用してサービスロールを作成し、がユーザーに代わってアクションを実行 AWS Marketplace するために使用できます。AWSMarketplaceAmiIngestion の使用の詳細については、「[AMI AWS Marketplace へのアクセスを許可する](#)」を参照してください。

このポリシーは、が Amazon マシンイメージ (AMIsをコピー AWS Marketplace して に一覧表示できるようにする権限を寄稿者に付与するために使用されます AWS Marketplace。

許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:ModifySnapshotAttribute"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:us-east-1::snapshot/snap-*"
    },
    {
      "Action": [
        "ec2:DescribeImageAttribute",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshotAttribute",
        "ec2:ModifyImageAttribute"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

AWS マネージドポリシー: **AWSMarketplaceFullAccess**

AWSMarketplaceFullAccess ポリシーは IAM ID にアタッチできます。

このポリシーは、販売者と購入者の両方として、AWS Marketplace および関連サービスへのフルアクセスを許可する管理権限を付与します。これらの権限には以下の権限が含まれます。

- AWS Marketplace ソフトウェアをサブスクライブおよびサブスクライブ解除します。

- から AWS Marketplace ソフトウェアインスタンスを管理します AWS Marketplace。
- アカウントでプライベートマーケットプレイスを作成して管理する。
- Amazon EC2、AWS CloudFormation、および Amazon EC2 Systems Manager へのアクセスを提供します。

許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:*",
        "cloudformation:CreateStack",
        "cloudformation:DescribeStackResource",
        "cloudformation:DescribeStackResources",
        "cloudformation:DescribeStacks",
        "cloudformation:List*",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CreateSecurityGroup",
        "ec2:CreateTags",
        "ec2:DescribeAccountAttributes",
        "ec2:DescribeAddresses",
        "ec2>DeleteSecurityGroup",
        "ec2:DescribeImages",
        "ec2:DescribeInstances",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets",
        "ec2:DescribeTags",
        "ec2:DescribeVpcs",
        "ec2:RunInstances",
        "ec2:StartInstances",
        "ec2:StopInstances",
        "ec2:TerminateInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
```

```

    "Action": [
      "ec2:CopyImage",
      "ec2:DeregisterImage",
      "ec2:DescribeSnapshots",
      "ec2>DeleteSnapshot",
      "ec2:CreateImage",
      "ec2:DescribeInstanceStatus",
      "ssm:GetAutomationExecution",
      "ssm:UpdateDocumentDefaultVersion",
      "ssm:CreateDocument",
      "ssm:StartAutomationExecution",
      "ssm:ListDocuments",
      "ssm:UpdateDocument",
      "ssm:DescribeDocument",
      "sns:ListTopics",
      "sns:GetTopicAttributes",
      "sns:CreateTopic",
      "iam:GetRole",
      "iam:GetInstanceProfile",
      "iam:ListRoles",
      "iam:ListInstanceProfiles"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:ListBucket",
      "s3:GetObject"
    ],
    "Resource": [
      "arn:aws:s3::*image-build*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "sns:Publish",
      "sns:setTopicAttributes"
    ],
    "Resource": "arn:aws:sns:*:*:*image-build*"
  },
  {
    "Effect": "Allow",

```

```
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "*"
    ],
    "Condition": {
      "StringLike": {
        "iam:PassedToService": [
          "ec2.amazonaws.com",
          "ssm.amazonaws.com"
        ]
      }
    }
  }
}
```

AWS マネージドポリシー: **AWSMarketplaceGetEntitlements**

AWSMarketplaceGetEntitlements ポリシーは IAM ID にアタッチできます。

このポリシーは、Software as a Service (SaaS) 製品販売者が顧客が AWS Marketplace SaaS 製品をサブスクライブしているかどうかをチェックできるようにする読み取り専用アクセス許可を付与します。

許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid" : "AWSMarketplaceGetEntitlements",
      "Effect" : "Allow",
      "Action": [
        "aws-marketplace:GetEntitlements"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS マネージドポリシー: **AWSMarketplaceMeteringFullAccess**

AWSMarketplaceMeteringFullAccess ポリシーは IAM ID にアタッチできます。

このポリシーは、AWS Marketplaceで柔軟な使用料金が設定されている AMI およびコンテナ製品に対応する従量制使用量をレポートできるようにする権限を寄稿者に付与します。

アクセス許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:MeterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

AWS マネージドポリシー: **AWSMarketplaceMeteringRegisterUsage**

AWSMarketplaceMeteringRegisterUsage ポリシーは IAM アイデンティティにアタッチできます。

このポリシーは、で時間単位の料金が設定されたコンテナ製品に対応する従量制使用量をレポートできるようにする権限を寄稿者に付与します AWS Marketplace。

許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:RegisterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

```
}
```

AWS マネージドポリシー: **AWSMarketplaceSellerFullAccess**

AWSMarketplaceSellerFullAccess ポリシーは IAM ID にアタッチできます。

このポリシーは、を含む のすべての販売者オペレーションへのフルアクセス AWS Marketplace と、AMI ベースの製品で使用される Amazon EC2 AMI AWS Marketplace 管理ポータル¹の管理を許可する管理アクセス許可を付与します。

アクセス許可の詳細

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MarketplaceManagement",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace-management:viewReports",
        "aws-marketplace-management:viewSupport",
        "aws-marketplace:ListChangeSets",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:CancelChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:ListTasks",
        "aws-marketplace:DescribeTask",
        "aws-marketplace:UpdateTask",
        "aws-marketplace:CompleteTask",
        "aws-marketplace:GetSellerDashboard",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots",
        "ec2:ModifyImageAttribute",
        "ec2:ModifySnapshotAttribute"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AgreementAccess",
      "Action": [
        "aws-marketplace:SearchAgreements",

```

```

        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:GetAgreementTerms"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws-marketplace:PartyType": "Proposer"
        },
        "ForAllValues:StringEquals": {
            "aws-marketplace:AgreementType": [
                "PurchaseAgreement"
            ]
        }
    }
},
{
    "Sid": "IAMGetRole",
    "Effect": "Allow",
    "Action": [
        "iam:GetRole"
    ],
    "Resource": "arn:aws:iam::*:role/*"
},
{
    "Sid": "AssetScanning",
    "Effect": "Allow",
    "Action": [
        "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::*:role/*",
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "assets.marketplace.amazonaws.com"
        }
    }
},
{
    "Sid": "VendorInsights",
    "Effect": "Allow",
    "Action": [
        "vendor-insights:GetDataSource",
        "vendor-insights:ListDataSources",
        "vendor-insights:ListSecurityProfiles",

```

```

        "vendor-insights:GetSecurityProfile",
        "vendor-insights:GetSecurityProfileSnapshot",
        "vendor-insights:ListSecurityProfileSnapshots"
    ],
    "Resource": "*"
},
{
    "Sid": "TagManagement",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:TagResource",
        "aws-marketplace:UntagResource",
        "aws-marketplace:ListTagsForResource"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
},
{
    "Sid": "SellerSettings",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace-management:GetSellerVerificationDetails",
        "aws-marketplace-management:PutSellerVerificationDetails",
        "aws-marketplace-management:GetBankAccountVerificationDetails",
        "aws-marketplace-management:PutBankAccountVerificationDetails",
        "aws-marketplace-management:GetSecondaryUserVerificationDetails",
        "aws-marketplace-management:PutSecondaryUserVerificationDetails",
        "aws-marketplace-management:GetAdditionalSellerNotificationRecipients",
        "aws-marketplace-management:PutAdditionalSellerNotificationRecipients",
        "payments:GetPaymentInstrument",
        "payments:CreatePaymentInstrument",
        "tax:GetTaxInterview",
        "tax:PutTaxInterview",
        "tax:GetTaxInfoReportingDocument"
    ],
    "Resource": "*"
},
{
    "Sid": "Support",
    "Effect": "Allow",
    "Action": [
        "support:CreateCase"
    ],
    "Resource": "*"
},

```

```

    {
      "Sid": "ResourcePolicyManagement",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:GetResourcePolicy",
        "aws-marketplace:PutResourcePolicy",
        "aws-marketplace>DeleteResourcePolicy"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
    },
    {
      "Sid": "CreateServiceLinkedRole",
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:AWSServiceName": "resale-
authorization.marketplace.amazonaws.com"
        }
      }
    }
  ]
}

```

AWS マネージドポリシー:

AWSMarketplaceSellerProductsFullAccess

AWSMarketplaceSellerProductsFullAccess ポリシーは IAM アイデンティティにアタッチできます。

このポリシーは、製品とへのフルアクセス AWS Marketplace 管理ポータル、および AMI ベースの製品で使用される Amazon EC2 AMI の管理を許可する権限を寄稿者に付与します。

アクセス許可の詳細

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:ListChangeSets",

```

```

        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:CancelChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:ListTasks",
        "aws-marketplace:DescribeTask",
        "aws-marketplace:UpdateTask",
        "aws-marketplace:CompleteTask",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots",
        "ec2:ModifyImageAttribute",
        "ec2:ModifySnapshotAttribute"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "iam:GetRole"
    ],
    "Resource": "arn:aws:iam::role/"
},
{
    "Effect": "Allow",
    "Action": [
        "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::role/",
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "assets.marketplace.amazonaws.com"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "vendor-insights:GetDataSource",
        "vendor-insights:ListDataSources",
        "vendor-insights:ListSecurityProfiles",
        "vendor-insights:GetSecurityProfile",
        "vendor-insights:GetSecurityProfileSnapshot",
        "vendor-insights:ListSecurityProfileSnapshots"
    ]
}

```

```

    ],
    "Resource": "*"
  }
  {
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:TagResource",
      "aws-marketplace:UntagResource",
      "aws-marketplace:ListTagsForResource"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
  }
]
}

```

AWS マネージドポリシー: **AWSMarketplaceSellerProductsReadOnly**

AWSMarketplaceSellerProductsReadOnly ポリシーは IAM アイデンティティにアタッチできません。

このポリシーは、AWS Marketplace 管理ポータルに製品を表示したり、AMI ベースの製品で使用されている Amazon EC2 AMI を表示したりするためのアクセスを許可する読み取り専用権限を付与します。

許可の詳細

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:ListChangeSets",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:ListTasks",
        "aws-marketplace:DescribeTask",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
    }
  ]
}

```

```

    }
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:ListTagsForResource"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
    }
  ]
}

```

AWS マネージドポリシー: **AWSVendorInsightsVendorFullAccess**

AWSVendorInsightsVendorFullAccess ポリシーは IAM ID にアタッチできます。

このポリシーは、AWS Marketplace Vendor Insights ですべてのリソースを作成および管理するためのフルアクセスを付与します。AWS Marketplace Vendor Insights は、このガイドの目的上、評価者を購入者として識別し、ベンダーは販売者と同等であると判断します。AWS Marketplace を更新AWSVendorInsightsVendorFullAccessして、契約検索の追加、プロファイルスナップショットの更新、ベンダーのタグ付けを行い、AWS Artifact サードパーティーのレポートへの読み取り専用アクセスを許可します。

許可の詳細

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "aws-marketplace:DescribeEntity",
      "Resource": "arn:aws:aws-marketplace:*:*:*/SaaSProduct/*"
    },
    {
      "Effect": "Allow",
      "Action": "aws-marketplace:ListEntities",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "vendor-insights:CreateDataSource",
        "vendor-insights:UpdateDataSource",

```

```

    "vendor-insights:DeleteDataSource",
    "vendor-insights:GetDataSource",
    "vendor-insights:ListDataSources",
    "vendor-insights:CreateSecurityProfile",
    "vendor-insights:ListSecurityProfiles",
    "vendor-insights:GetSecurityProfile",
    "vendor-insights:AssociateDataSource",
    "vendor-insights:DisassociateDataSource",
    "vendor-insights:UpdateSecurityProfile",
    "vendor-insights:ActivateSecurityProfile",
    "vendor-insights:DeactivateSecurityProfile",
    "vendor-insights:UpdateSecurityProfileSnapshotCreationConfiguration",
    "vendor-insights:UpdateSecurityProfileSnapshotReleaseConfiguration",
    "vendor-insights:GetSecurityProfileSnapshot",
    "vendor-insights:ListSecurityProfileSnapshots"
    "vendor-insights:TagResource",
    "vendor-insights:UntagResource",
    "vendor-insights:ListTagsForResource",
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "aws-marketplace:AcceptAgreementApprovalRequest",
    "aws-marketplace:RejectAgreementApprovalRequest",
    "aws-marketplace:GetAgreementApprovalRequest",
    "aws-marketplace:ListAgreementApprovalRequests"
    "aws-marketplace:CancelAgreement",
    "aws-marketplace:SearchAgreements"
  ],
  "Resource": "*",
  "Condition": {
    "ForAllValues:StringEquals": {
      "aws-marketplace:AgreementType": "VendorInsightsAgreement"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "artifact:GetReport",
    "artifact:GetReportMetadata",
    "artifact:GetTermForReport",

```

```

        "artifact:ListReports",
    ],
    "Resource": "arn:aws:artifact:*::report/*"
}
]
}

```

AWS マネージドポリシー: **AWSVendorInsightsVendorReadOnly**

AWSVendorInsightsVendorReadOnly ポリシーは IAM ID にアタッチできます。

このポリシーは AWS Marketplace、Vendor Insights プロファイルと関連リソースを表示するための読み取り専用アクセスを許可します。AWS Marketplace Vendor Insights は、このガイドの目的上、評価者を購入者として識別し、ベンダーは販売者と等しくなります。AWS Marketplace を更新AWSVendorInsightsVendorReadOnlyしてタグを一覧表示するアクセス許可を追加し、AWS Artifact サードパーティーのレポートへの読み取り専用アクセスを許可します。

許可の詳細

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "aws-marketplace:DescribeEntity",
      "Resource": "arn:aws:aws-marketplace:*:*:*/SaaSProduct/*"
    },
    {
      "Effect": "Allow",
      "Action": "aws-marketplace:ListEntities",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "vendor-insights:GetDataSource",
        "vendor-insights:ListDataSources",
        "vendor-insights:ListSecurityProfiles",
        "vendor-insights:GetSecurityProfile",
        "vendor-insights:GetSecurityProfileSnapshot",
        "vendor-insights:ListSecurityProfileSnapshots",
        "vendor-insights:ListTagsForResource"
      ]
    }
  ]
}

```

```

    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "artifact:GetReport",
      "artifact:GetReportMetadata",
      "artifact:GetTermForReport",
      "artifact:ListReports"
    ],
    "Resource": "arn:aws:artifact:*::report/*"
  }
]
}

```

AWS MarketplaceAWS 管理ポリシーの更新

このサービスがこれらの変更の追跡を開始した AWS Marketplace 以降の の AWS マネージドポリシーの更新に関する詳細を表示します。このページへの変更に関する自動アラートについては、AWS Marketplace [ドキュメント履歴](#) ページの RSS フィードを購読してください。

変更	説明	日付
AWSMarketplaceSellerFullAccess - ポリシーを更新	、 、 aws-marketplace-management:viewMarketing aws-marketplace-management:viewSettings および のアクションの削除を反映するようにAWSMarketplaceSellerFullAccess ドキュメントを更新しましたaws-marketplace-management:uploadFiles 。この更新には、「きめ細かなアクセス許可の使用」セクションの削除も含まれています。	2024 年 6 月 4 日

変更	説明	日付
AWSMarketplaceGetEntitlements - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceGetEntitlements し、ポリシーステートメントにを追加しました。	2024 年 3 月 22 日
AWSMarketplaceSellerFullAccess - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess、サービスにリンクされたロールを作成するためのアクセス許可が追加されました。	2024 年 3 月 15 日
AWSMarketplaceSellerFullAccess - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess し、税金情報にアクセスするためのアクセス許可を追加しました。	2024 年 2 月 8 日
AWSVendorInsightsVendorFullAccess – Updated policy	AWS Marketplace が更新されAWSVendorInsightsVendorFullAccess、データソースを更新するアクセス許可が追加されました。	2023 年 10 月 18 日
AWSMarketplaceSellerFullAccess - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess、エンティティを共有するためのアクセス許可が追加されました。	2023 年 6 月 1 日

変更	説明	日付
AWSMarketplaceSellerFullAccess - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess、アカウント検証、銀行口座検証、ケース管理、販売者通知の詳細に関連するアクセス許可が追加されました。	2023年6月1日
AWSMarketplaceSellerFullAccess - ポリシーを更新	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess、販売者ダッシュボードにアクセスするためのアクセス許可が追加されました。	2022年12月23日
AWSMarketplaceSellerFullAccess 、 AWSMarketplaceSellerProductsFullAccess 、 AWSMarketplaceSellerProductsReadOnly - 既存のポリシーの更新	AWS Marketplace は、新しいタグベースの認証機能のポリシーを更新しました。	2022年12月9日
AWS Marketplace 更新済み AWSVendorInsightsVendorFullAccess	AWS Marketplace を更新しAWSMarketplaceSellerProductsFullAccessして、契約検索、プロフィールスナップショットの更新、ベンダーのタグ付けを追加し、サードパーティーのレポート (プレビュー) AWS Artifact への読み取り専用アクセスを許可しました。	2022年11月30日

変更	説明	日付
<p>AWS Marketplace 更新済み AWSVendorInsightsVendorReadOnly</p>	<p>AWS Marketplace タグ AWSVendorInsightsVendorReadOnly を一覧表示するアクセス許可を追加し、サードパーティーレポート (プレビュー) への AWS Artifact 読み取り専用アクセスを許可するように更新しました。</p>	<p>2022 年 11 月 30 日</p>
<p>AWSVendorInsightsVendorFullAccess と AWSVendorInsightsVendorReadOnly - ポリシーを追加</p>	<p>AWS Marketplace AWS Marketplace Vendor Insights の新機能である AWSMarketplaceSellerProductsFullAccess およびのポリシーが追加されました AWSVendorInsightsVendorReadOnly 。</p>	<p>2022 年 7 月 26 日</p>
<p>AWSMarketplaceSellerProductsFullAccess および AWSMarketplaceSellerFullAccess - ポリシーを更新</p>	<p>AWS Marketplace AWS Marketplace Vendor Insights の新機能である AWSMarketplaceSellerProductsFullAccess およびのポリシーが更新されました AWSMarketplaceSellerFullAccess 。</p>	<p>2022 年 7 月 26 日</p>
<p>AWSMarketplaceSellerFullAccess と AWSMarketplaceSellerProductsFullAccess - 既存のポリシーに対する更新</p>	<p>AWS Marketplace は、<code>iam:PassedToService</code> 条件がにのみ適用されるようにポリシーを更新しました <code>iam:PassRole</code> 。</p>	<p>2021 年 11 月 22 日</p>

変更	説明	日付
AWSMarketplaceFullAccess - 既存ポリシーへの更新。	AWS Marketplace がAWSMarketplaceFull Access ポリシーから重複するec2:DescribeAccountAttributes アクセス許可を削除しました。	2021 年 7 月 20 日
AWS Marketplace が変更の追跡を開始しました	AWS Marketplace が AWS マネージドポリシーの変更の追跡を開始しました。	2021 年 4 月 20 日

AWS Marketplace Commerce Analytics Service アカウントのアクセス許可

AWS Marketplace Commerce Analytics Service に登録するには、次の IAM アクセス許可ポリシーを使用します。

登録方法については、[オンボーディングガイド](#)を参照してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles",
        "iam:CreateRole",
        "iam:CreatePolicy",
        "iam:AttachRolePolicy",
        "aws-marketplace-management:viewReports"
      ],
      "Resource": "*"
    }
  ]
}
```

以下の IAM アクセス許可ポリシーを使用して、ユーザーが AWS Marketplace Commerce Analytics Service にリクエストを送信できるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

この機能の詳細については、「[AWS Marketplace コマース分析サービス](#)」を参照してください。

Amazon SQS のアクセス許可

SaaS 製品の公開プロセスの一環として、は、顧客のサブスクリプションまたは使用権限のステータスに変更された場合に通知を受け取るために使用できる Amazon SNS トピック AWS Marketplace を提供します。キューが通知に応じてアクションを実行できるよう、1 つまたは複数の Amazon SQS キューをトピックに設定できます。例えば、顧客が SaaS 製品のサブスクリプションにストレージを追加した場合、Amazon SNS トピックは Amazon SQS キューにメッセージを送信し、その顧客が使用可能なストレージ容量を自動的に増加するプロセスを開始できます。

提供された Amazon SNS トピックへの Amazon Simple Queue Service (Amazon SQS) キューをサブスクライブすると、そのトピックがそのキューにメッセージを発行するためのアクセス許可が自動的に追加されます。ただし、AWS Marketplace Metering and Entitlement Service API ユーザーにキューへのアクセスを許可するための IAM ポリシーは必要です。そのポリシーは、サービスが同じ認証情報で実行されている場合は、同じユーザーに適用できます。次の内容のポリシーを作成して、販売者のユーザーまたはロールにアタッチします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "sqs:ReceiveMessage", "sqs:DeleteMessage", "sqs:GetQueueAttributes",
        "sqs:GetQueueUrl"
      ],
    }
  ],
}
```

```
        "Effect": "Allow",
        "Resource": "arn:aws:sqs:REGION_HERE:XXXXXXXXXXXX:NAME_HERE"
    }
]
}
```

Note

Resource フィールドは Amazon SQS キューの Amazon リソースネーム (ARN) です。

SaaS 製品のメッセージ通知およびキューに関する詳細情報については、[the section called “SNS トピックへの SQS キューのサブスクリプション”](#) および [the section called “AWS Marketplace Metering Service API および使用権限管理サービス API へのアクセス”](#) を参照してください。

AWS Marketplace 計測と使用権限 API のアクセス許可

Software as a Service (SaaS) 製品、Amazon Machine Image (AMI) 製品、およびコンテナ製品は、AWS Marketplace Metering Service および AWS Marketplace Entitlement Service APIs を使用できません。タイプ別に異なる AWS Identity and Access Management (IAM) アクセス許可が必要です。製品については、すべての使用量を計測し、指定した計測レコード AWS に基づいてから顧客に請求されます。AWS Marketplace 計測レコードの提供に必要な統合を有効にするには、統合が使用しているサービスアカウントでアクセスを有効にするために制約のある IAM ポリシーが必要です。計測情報を送信する対象の製品タイプのポリシーを、統合で使用しているユーザーまたはロールにアタッチします。

トピック

- [SaaS 製品の IAM ポリシー](#)
- [AMI 製品の IAM ポリシー](#)
- [コンテナ製品の IAM ポリシー](#)

SaaS 製品の IAM ポリシー

以下のポリシーでは、すべての SaaS 統合に最初のアクセス許可、aws-marketplace:ResolveCustomer が必要です。2 つ目のアクセス許可、aws-marketplace:BatchMeterUsage は AWS Marketplace Metering Service API に必要です。3 つ

目のアクセス許可、`aws-marketplace:GetEntitlements` は AWS Marketplace Entitlement Service API に必要です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ResolveCustomer",
        "aws-marketplace:BatchMeterUsage",
        "aws-marketplace:GetEntitlements"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

SaaS 製品の詳細については、「[SaaS ベースの製品](#)」を参照してください。

AMI 製品の IAM ポリシー

AMI 製品には以下の IAM ポリシーを使用してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:MeterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

AMI 製品の詳細については、「[AMI ベースの製品](#)」を参照してください。

コンテナ製品の IAM ポリシー

コンテナ製品には以下の IAM ポリシーを使用してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:RegisterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

コンテナ製品の詳細については、「[コンテナベースの製品](#)」を参照してください。

ユーザーの作成の詳細については、「IAM ユーザーガイド」の「[AWS アカウントでの IAM ユーザーの作成](#)」を参照してください。ポリシーの作成および割り当ての詳細については、「[IAM ユーザーのアクセス許可の変更](#)」を参照してください。

このポリシーでは、ポリシーがアタッチされる IAM ロールまたはユーザーに API へのアクセス許可を付与しています。これらの API コールで別のアカウントによるロールの引き受けを有効にする方法の詳細については、AWS Partner Network (APN) ブログの「[How to Best Architect Your AWS Marketplace SaaS Subscription Across Multiple AWS アカウント](#)」を参照してください。

のサービスにリンクされたロールの使用 AWS Marketplace

AWS Marketplace は AWS Identity and Access Management、(IAM) [サービスにリンクされたロール](#)を使用します。サービスにリンクされたロールは、に直接リンクされた一意のタイプの IAM ロールです AWS Marketplace。サービスにリンクされたロールは、によって事前定義 AWS Marketplace されており、ユーザーに代わってサービスから他の AWS のサービスを呼び出すために必要なすべてのアクセス許可が含まれています。

の再販承認のためのロールの使用 AWS Marketplace

AWS Marketplace は AWS Identity and Access Management、(IAM) [サービスにリンクされたロール](#)を使用します。サービスにリンクされたロールは、に直接リンクされた一意のタイプの IAM ロールです AWS Marketplace。サービスにリンクされたロールはによって事前定義 AWS Marketplace されており、ユーザーに代わってサービスから他の AWS のサービスを呼び出すために必要なすべてのアクセス許可が含まれています。

サービスにリンクされたロールを使用すると、必要なアクセス許可を手動で追加する必要がなくなるため、の設定 AWS Marketplace が簡単になります。は、サービスにリンクされたロールのアクセス許可 AWS Marketplace を定義し、特に定義されている場合を除き、のみがそのロールを引き受け AWS Marketplace ることができます。定義したアクセス許可には、信頼ポリシーと許可ポリシーが含まれます。この許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスリンクロールは、まずその関連リソースを削除しなければ削除できません。これにより、AWS Marketplace リソースへのアクセス許可を誤って削除することがなくなるため、リソースが保護されます。

サービスリンクロールをサポートする他のサービスについては、「[IAM と連動するAWS のサービス](#)」を参照し、[Service-linked role (サービスリンクロール)] の列内で [Yes (はい)] と表記されたサービスを確認してください。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

のサービスにリンクされたロールのアクセス許可 AWS Marketplace

AWS Marketplace は、 という名前のサービスにリンクされたロールを使用します。これによりAWSServiceRoleForMarketplaceResaleAuthorization、 が再販認可 AWS Marketplace に使用または管理する AWS のサービスやリソースにアクセスできます。

AWSServiceRoleForMarketplaceResaleAuthorization サービスにリンクされたロールは、次のサービスを信頼してロールを引き受けます。

- resale-authorization.marketplace.amazonaws.com

という名前のロールアクセス許可ポリシーAWSServiceRoleForMarketplaceResaleAuthorizationServiceRolePolicyは AWS Marketplace 、 が指定されたリソースに対して次のアクションを実行できるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowResaleAuthorizationShareActionsRAMCreate",
    "Effect": "Allow",
    "Action": [
      "ram:CreateResourceShare"
    ],
    "Resource": [
      "arn:aws:ram:*:*:*"
    ],
    "Condition": {
```

```

        "StringEquals": {
            "ram:RequestedResourceType": "aws-marketplace:Entity"
        },
        "ArnLike": {
            "ram:ResourceArn": "arn:aws:aws-marketplace:*:*:AWSMarketplace/
ResaleAuthorization/*"
        },
        "Null": {
            "ram:Principal": "true"
        }
    }
},
{
    "Sid": "AllowResaleAuthorizationShareActionsRAMAssociate",
    "Effect": "Allow",
    "Action": [
        "ram:AssociateResourceShare"
    ],
    "Resource": [
        "arn:aws:ram:*:*:*"
    ],
    "Condition": {
        "Null": {
            "ram:Principal": "false"
        },
        "StringEquals": {
            "ram:ResourceShareName": "AWSMarketplaceResaleAuthorization"
        }
    }
},
{
    "Sid": "AllowResaleAuthorizationShareActionsRAMAccept",
    "Effect": "Allow",
    "Action": [
        "ram:AcceptResourceShareInvitation"
    ],
    "Resource": [
        "arn:aws:ram:*:*:*"
    ],
    "Condition": {
        "StringEquals": {
            "ram:ResourceShareName": "AWSMarketplaceResaleAuthorization"
        }
    }
}

```

```

    },
    {
      "Sid": "AllowResaleAuthorizationShareActionsRAMGet",
      "Effect": "Allow",
      "Action": [
        "ram:GetResourceShareInvitations",
        "ram:GetResourceShareAssociations"
      ],
      "Resource": [
        "arn:aws:ram:*:*:*"
      ]
    },
    {
      "Sid": "AllowResaleAuthorizationShareActionsMarketplace",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:PutResourcePolicy",
        "aws-marketplace:GetResourcePolicy"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": ["ram.amazonaws.com"]
        }
      }
    },
    {
      "Sid": "AllowResaleAuthorizationShareActionsMarketplaceDescribe",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:DescribeEntity"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/*"
    }
  ]
}

```

ユーザー、グループ、ロールなどがサービスにリンクされたロールを作成、編集、削除できるようにするには、アクセス権を設定する必要があります。詳細については、IAM ユーザーガイドの「[サービスリンクロールのアクセス許可](#)」を参照してください。

のサービスにリンクされたロールの作成 AWS Marketplace

サービスリンクロールを手動で作成する必要はありません。でサービスにリンクされたロールを作成すると AWS Marketplace 管理ポータル、によってサービスにリンクされたロール AWS Marketplace が自動的に作成されます。

サービスにリンクされたロールを作成するには

1. で[AWS Marketplace 管理ポータル](#)、管理アカウントにサインインし、設定 を選択します。
2. 「設定」セクションで、「サービスにリンクされたロール」タブを選択します。
3. 「サービスにリンクされたロール」ページで、「再販認可」または「再販認可」統合の「サービスにリンクされたロール」を選択し、「サービスにリンクされたロールの作成」または「統合の設定」を選択します。
4. 「再販売承認」または「再販売承認統合の作成」ページの「サービスにリンクされたロール」で、「サービスにリンクされたロールの作成」または「統合の作成」を選択して情報を確認し、確認します。

サービスにリンクされたロールページに、再販売承認サービスにリンクされたロールが正常に作成されたことを示すメッセージが表示されます。

このサービスリンクロールを削除した後で再度作成する必要がある場合は、同じ方法でアカウントにロールを再作成できます。でサービスにリンクされたロールを作成すると AWS Marketplace 管理ポータル、によってサービスにリンクされたロールが再度 AWS Marketplace 作成されます。

のサービスにリンクされたロールの編集 AWS Marketplace

AWS Marketplace では、`AWSServiceRoleForMarketplaceResaleAuthorization` サービスにリンクされたロールを編集することはできません。サービスリンクロールを作成した後は、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロールの説明の編集はできます。詳細については、「IAM ユーザーガイド」の[「サービスにリンクされたロールの編集」](#)を参照してください。

のサービスにリンクされたロールの削除 AWS Marketplace

サービスリンクロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、モニタリングや保守が積極的に行われていない未使用のエンティティを排除できます。

Note

独立系ソフトウェアベンダー (ISVsに ロールがない場合、ターゲットチャネルパートナーと新しい再販認可を自動的に共有 AWS Resource Access Manager しません。チャネルパートナーが ロールを持っていない場合、チャネルパートナーを対象とする再販承認を自動的に承認 AWS Resource Access Manager しません。

サービスにリンクされたロールを IAM で手動削除するには

IAM コンソール、または AWS API を使用して AWS CLI、サービスにリンクされたロールを削除します `AWSServiceRoleForMarketplaceResaleAuthorization`。詳細については、「IAM ユーザーガイド」の「[サービスにリンクされたロールの削除](#)」を参照してください。

AWS Marketplace サービスにリンクされたロールでサポートされているリージョン

AWS Marketplace は、サービスが利用可能なすべてのリージョンでサービスにリンクされたロールの使用をサポートします。詳細については、「[AWS リージョンとエンドポイント](#)」を参照してください。

を使用した AWS Marketplace Metering API コールのログ記録 AWS CloudTrail

AWS Marketplace は と統合されています AWS Marketplace。これは AWS CloudTrail、. CloudTrail captures API コール AWS のサービス でユーザー、ロール、または によって実行されたアクションをイベント AWS Marketplace として記録するサービスです。キャプチャされた呼び出しには、AWS Marketplace コンソールからの呼び出しと AWS Marketplace API オペレーションへのコード呼び出しが含まれます。

CloudTrail アカウントを作成する AWS アカウントと、 で が有効になります。 でサポートされているイベントアクティビティが発生すると AWS Marketplace、そのアクティビティは CloudTrail イベント履歴 の他の AWS のサービス イベントとともにイベントに記録されます。アカウントで最近のイベントを表示、検索、ダウンロードできます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます:

- リクエストがルート認証情報と AWS Identity and Access Management ユーザー認証情報のどちらを使用して行われたか。

- リクエストが、ロールとフェデレーティッドユーザーのどちらかの一時的なセキュリティ認証情報を使用して送信されたかどうか。
- リクエストが、別の AWS のサービスによって送信されたかどうか。

AWS Marketplace は、BatchMeterUsage オペレーションをイベントとして CloudTrail ログファイルに記録することをサポートします。

AWS Marketplace 計測 API ログファイルエントリの例

例: BatchMeterUsage

次の例は、からの BatchMeterUsage アクションを示す CloudTrail ログエントリを示しています AWS Marketplace Metering Service。販売者が [計測レコードを送信して、にリストされている Software as a Service \(SaaS\) 製品の顧客使用状況を報告する](#) と AWS Marketplace、この CloudTrail ログエントリは販売者の に記録されます AWS アカウント。SaaS

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:user/Alice",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "userName": "Alice"
  },
  "eventTime": "2018-04-19T16:32:51Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "BatchMeterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.0.2/24",
  "userAgent": "Coral/Netty14",
  "requestParameters": {
    "usageRecords": [
      {
        "dimension": "Dimension1",
        "timestamp": "Apr 19, 2018 4:32:50 PM",
        "customerIdentifier": "customer1",
        "quantity": 1
      }
    ]
  },
}
```

```

        "productCode": "EXAMPLE_proCode"
    },
    "responseElements": {
        "results": [
            {
                "usageRecord": {
                    "dimension": "Dimension1",
                    "timestamp": "Apr 19, 2018 4:32:50 PM",
                    "customerIdentifier": "customer1",
                    "quantity": 1
                },
                "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e",
                "status": "Success"
            }
        ],
        "unprocessedRecords": [ ]
    },
    "requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
    "eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
]
}

```

例: コンテナ用 RegisterUsage

次の例は、からの RegisterUsage アクションを示す CloudTrail ログエントリを示しています。AWS Marketplace Metering Service。からの時間単位のコンテナ製品が購入者のにデプロイされると AWS アカウント、コンテナ内のソフトウェア AWS Marketplace は購入者の RegisterUsage AWS アカウント 内で を呼び出して、その Amazon Elastic Container Service (Amazon ECS) タスク または Amazon Elastic Kubernetes Service (Amazon EKS) ポッドの時間単位の計測を開始します。この CloudTrail ログエントリは、購入者のに記録されます AWS アカウント。

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:botocore-session-1111111111",
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/botocore-session-1111111111",

```

```
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {
        "federatedProvider": "arn:aws:iam::123456789012:oidc-provider/
oidc.eks.us-east-1.amazonaws.com/id/EXAMPLEFA1C58F08CDB049167EXAMPLE",
        "attributes": {}
      },
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-23T02:19:34Z"
      }
    }
  },
  "eventTime": "2020-07-23T02:19:46Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "RegisterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "aws-cli/1.18.103 Python/3.8.2 Linux/4.14.181-142.260.amzn2.x86_64
botocore/1.17.26",
  "requestParameters": {
    "productCode": "EXAMPLE_proCode",
    "publicKeyVersion": 1
  },
  "responseElements": {
    "signature": "eyJhbGciOiJIUzI1Ni..."
  },
  "requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
  "eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

例: Amazon EKS 上のコンテナの MeterUsage

次の例は、Amazon EKS 上のコンテナ AWS Marketplace Metering Service のからの MeterUsage アクションを示す CloudTrail ログエントリを示しています。からの [カスタム計測](#) を使用するコンテナ製品が購入者の にデプロイされると AWS アカウント、コンテナ内のソフトウェア AWS Marketplace は購入者の MeterUsage 内で を呼び出し AWS アカウント で 1 時間ごとにレポートします。この CloudTrail ログエントリは、購入者の に記録されます AWS アカウント。

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:botocore-session-1111111111",
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/botocore-session-1111111111",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {
        "federatedProvider": "arn:aws:iam::123456789012:oidc-provider/oidc.eks.us-east-1.amazonaws.com/id/EXAMPLEFA1C58F08CDB049167EXAMPLE",
        "attributes": {}
      },
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-23T01:03:26Z"
      }
    }
  },
  "eventTime": "2020-07-23T01:38:13Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "MeterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "aws-cli/1.18.103 Python/3.8.2 Linux/4.14.181-142.260.amzn2.x86_64 botocore/1.17.26",
```

```

"requestParameters": {
  "timestamp": "Jul 23, 2020 1:35:44 AM",
  "usageQuantity": 1,
  "usageDimension": "Dimension1",
  "productCode": "EXAMPLE_proCode"
},
"responseElements": {
  "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e"
},
"requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
"eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}

```

例: AMI の MeterUsage

次の例は、Amazon マシンイメージ (AMIs) AWS Marketplace Metering Service のからの MeterUsage アクションを示す CloudTrail ログエントリを示しています。からのカスタム計測を使用する AMI 製品が購入者のにデプロイされると AWS アカウント、AMI のソフトウェア AWS Marketplace は購入者の MeterUsage 内で を呼び出し AWS アカウント、1 時間ごとに使用状況を報告します。この CloudTrail ログエントリは、購入者のに記録されます AWS アカウント。

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:i-exampled859aa775c",
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/i-exampled859aa775c",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-10T23:05:20Z"
      }
    }
  }
}

```

```
    },
    "ec2RoleDelivery": "1.0"
  }
},
"eventTime": "2020-07-10T23:06:42Z",
"eventSource": "metering-marketplace.amazonaws.com",
"eventName": "MeterUsage",
"awsRegion": "us-east-1",
"sourceIPAddress": "1.2.3.4",
"userAgent": "aws-cli/1.16.102 Python/2.7.16 Linux/4.14.133-113.112.amzn2.x86_64
botocore/1.12.92",
"requestParameters": {
  "productCode": "EXAMPLE_proCode",
  "timestamp": "Jul 10, 2020 11:06:41 PM",
  "usageDimension": "Dimension1",
  "usageQuantity": 1,
  "dryRun": false
},
"responseElements": {
  "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e"
},
"requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
"eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

ドキュメント履歴

以下の表は、本リリースの AWS Marketplace 販売者ガイドのドキュメントを説明するものです。

このドキュメントの更新に関する通知については、RSS フィードにサブスクライブできます。

変更	説明	日付
非推奨のアクセス許可を削除しました	AWSMarketplaceSellerFullAccess ドキュメントを更新して、3つのアクションを削除しました。	2024年6月4日
Amazon マシンイメージ (AMI) の年間契約修正オプションを更新	販売者は、既存のサブスクリプションのインスタンスタイプを追加または切り替えることができるようになりました。	2024年5月8日
Know Your Customer (KYC) プロセスを更新しました。	販売者の KYC プロセスにステップを追加しました。	2024年5月21日
AWS Marketplace 販売者のプライベートオファーエクスペリエンスを更新	プライベートオファーを作成および管理するためのエクスペリエンスを強化するためのコンテンツを追加しました。	2024年5月20日
Amazon EKS アドオン製品の要件を更新	「AWS Marketplace アドオンとしてコンテナ製品を準備する」セクションを更新し、「アドオン設定要件とアドオンプロバイダーのベストプラクティス」を追加しました。	2024年5月8日
AWS Marketplace 販売者のアクセス許可を更新しました	構文エラーを修正するためにアクセス許可の例を更新しました。	2024年4月2日

SaaS 契約料金の更新	料金モデルを使用した SaaS 契約の pay-as-go 超過の報告に関する内容を更新しました。	2024 年 4 月 2 日
の新しいデモとプライベート オファターのオプション AWS Marketplace	AWS Marketplace では、一部の販売者の製品詳細ページで デモ および プライベート オファター のリクエストオプションがサポートされるようになりました。	2024 年 4 月 1 日
税金の新しいダッシュボード	AWS Marketplace は、での取引に対する米国および国際税に関する可視化およびきめ細かなデータを提供する税務ダッシュボードをサポートするようになりました AWS Marketplace。	2024 年 3 月 29 日
AWSMarketplaceGetEntitlements の更新	が AWSMarketplaceGetEntitlements マネージドポリシーsidに追加されました。	2024 年 3 月 22 日
プロフェッショナルサービスの手順を更新	製品の可視性の編集とプロフェッショナルサービス製品の手順の削除を更新しました。	2024 年 3 月 19 日
更新された AMI アクセスポリシー	Linux 固有の AMI アクセスポリシーと Unix のような AMI アクセスポリシーを明確にするために、セクションを更新しました。	2024 年 3 月 19 日
の検索エンジン最適化情報を追加しました AWS Marketplace	の検索エンジンの最適化に関連するコンテンツを追加しました AWS Marketplace。	2024 年 3 月 19 日

[の販売者の 管理ポリシーを更新しました AWS Marketplace](#)

サービスにリンクされたロールの作成に関連するアクセス許可を追加AWSMarketplaceSellerFullAccess するために更新されました。

2024 年 3 月 15 日

[での製品の新しいサービスにリンクされたロール AWS Marketplace](#)

AWS Marketplace は、 が再販認可 AWS Marketplace に使用または管理する AWS のサービスおよびリソースへのアクセスを可能にするサービスにリンクされたロールを提供するようになりました。

2024 年 3 月 15 日

[の販売者の マネージドポリシーを更新しました AWS Marketplace](#)

税金情報へのアクセスに関連するアクセス許可を追加AWSMarketplaceSellerFullAccess するために更新されました。

2024 年 2 月 8 日

[セキュリティレポートの Amazon EventBridge イベント](#)

AWS Marketplace は、販売者の製品でセキュリティ脆弱性レポートが利用可能になったときに、以前は Amazon CloudWatch Events と呼ばれていた Amazon EventBridge イベントをサポートするようになりました。

2024 年 1 月 31 日

[Amazon EKS アドオンのサポート](#)

コンテナベースの製品から AWS Marketplace Amazon EKS アドオンへの発行に関連するコンテンツと手順を追加しました。

2024 年 1 月 29 日

プロフェッショナルサービス販売者が製品を再販するためのサポートを追加	販売者は、独立系ソフトウェアベンダー (ISVs) としてチャネルパートナーに再販の機会を作成できるようになりました。	2024 年 1 月 18 日
での将来の日付の契約の一般提供 AWS Marketplace	すべての ISVs と AWS Marketplace チャネルパートナーは、前払い料金でプライベートオファーを公開する一環として、将来の開始日を指定できます。	2024 年 1 月 16 日
新しい使用状況ダッシュボード	AWS Marketplace は、SaaS およびサーバー使用量ベースの製品を使用しているお客様に視覚化されたきめ細かなデータを提供する使用状況ダッシュボードをサポートするようになりました。	2024 年 1 月 10 日
販売者向けの新しいクイック起動デプロイオプション	Software as a Service (SaaS) 製品の新しいクイック起動デプロイオプションに関連する内容と手順を追加しました。	2023 年 11 月 29 日
プライベートオファー用の Flexible Payment Schedule が利用可能に	プライベートオファー用の Flexible Payment Schedule (FPS) が AWS Marketplace のすべてのお客様に利用可能になりました。	2023 年 11 月 17 日
コンテナ製品のセルフサービスコンテンツを追加しました	コンテナ製品のセルフサービスアクションに関連するコンテンツと手順を追加しました。	2023 年 11 月 3 日

契約および更新用の新しいダッシュボード	AWS Marketplace は、販売者向けの契約と更新ダッシュボードを提供するようになりました。	2023 年 10 月 31 日
変更セットの Amazon EventBridge イベント	AWS Marketplace は、変更セットが成功、失敗、またはキャンセルのステータスで完了したときに、以前は Amazon CloudWatch Events と呼ばれていた Amazon EventBridge イベントをサポートするようになりました。	2023 年 10 月 31 日
AWS Marketplace Vendor Insights で販売者の 管理ポリシーを更新	AWS Marketplace が更新されAWSVendorInsightsVendorFullAccess 、データソースを更新するアクセス許可が追加されました。	2023 年 10 月 18 日
AWS Marketplace for Desktop Applications (AMDA) のサポート終了	AWS Marketplace は 2023 年 10 月 2 日に AMDA のサポートを終了しました。AMDA に関するすべての内容と手順がガイドから削除されました。	2023 年 10 月 2 日
SaaS 製品のセルフサービスコンテンツを追加しました	Software as a Service (SaaS) 製品のセルフサービスアクションに関連するコンテンツと手順を追加しました。	2023 年 9 月 12 日
AWS Marketplace 販売者が日次支払いを利用できるようになりました	の販売者は AWS Marketplace 管理ポータル、毎日または毎月の支払いを受け取ることができるようになりました。	2023 年 9 月 7 日

Amazon のサポート EventBridge	AWS Marketplace は、販売者が新しいオファーを受け取ったときに EventBridge イベントをサポートするようになりました。	2023 年 9 月 6 日
AWS Marketplace Vendor Insights の新しいセルフサービスオンボーディングエクスペリエンス	AWS Marketplace Vendor Insights でセルフサービスのオンボーディングエクスペリエンスがサポートされるようになりました。	2023 年 8 月 17 日
の販売者の 管理ポリシーを更新しました AWS Marketplace	エンティティの共有に関連するアクセス許可を追加するように AWSMarketplaceSellerFullAccess を更新しました。	2023 年 6 月 1 日
の販売者の 管理ポリシーを更新しました AWS Marketplace	AWSMarketplaceSellerFullAccess が更新され、アカウント検証、銀行口座検証、ケース管理、販売者への通知の詳細に関するアクセス許可が追加されました。	2023 年 6 月 1 日
のきめ細かなアクセスに関するコンテンツを追加 AWS Marketplace	にきめ細かなアクセスの概要情報、アクセス許可、手順を追加しました AWS Marketplace 管理ポータル。	2023 年 6 月 1 日
AMI 製品の手順を追加しました	AMI 製品のセルフサービス手順を追加しました。	2023 年 5 月 12 日
での Amazon Tax Exemption Program と米国税計算サービスのサポートの終了 AWS Marketplace	Amazon 免税プログラムと米国税計算サービスは、2023 年 3 月 6 日にサポートが終了したため、このガイドから削除されました。	2023 年 3 月 6 日

コンテナ製品の手順を追加しました	コンテナ製品の設定を変更する新しい手順を追加しました。	2023年2月13日
AWS Marketplace 製品サポート接続のサポート終了	AWS Marketplace 2022年11月30日をもって、Commerce Analytics Service を使用した製品サポート接続と共有の顧客連絡先の詳細はサポートされなくなり AWS Marketplace ました。ガイドから AWS Marketplace 製品サポート接続のコンテンツを削除しました。	2023年1月27日
コンサルティングパートナー名の変更	AWS Marketplace は、コンサルティングパートナーをチャネルパートナーと呼ぶようになりました。ガイドが更新され、名前の変更のみが反映されました。	2023年1月26日
カスタマーサービスチームの名前が変更されました	Managed Catalog Operations (MCO) チームは、名前を AWS Marketplace Seller Operations チームに変更しました。ガイドが更新され、名前の変更のみが反映されました。	2023年1月24日
プライベートオファーページ	認証された購入者は、AWS Marketplace プライベートオファーページでに拡張 AWS アカウントされたプライベートオファーを表示できるようになりました。	2023年1月19日

SaaS 製品設定に関するトピックを追加しました	SaaS 製品設定を変更する手順を含む新しいトピックを追加しました。	2023 年 1 月 6 日
の販売者の 管理ポリシーを更新しました AWS Marketplace	AWS Marketplace が更新されAWSMarketplaceSellerFullAccess 、販売者ダッシュボードにアクセスするためのアクセス許可が追加されました。	2022 年 12 月 23 日
販売者へのメール通知が更新されました	プライベートオファーが公開されると販売者に通知されるようになりました。	2022 年 12 月 22 日
AWS Marketplaceの販売者は、サブスクリプションの SaaS 無料トライアルを利用できるようになりました	販売者はサブスクリプション商品の無料トライアルを作成できるようになりました。	2022 年 12 月 16 日
AWS Marketplaceでの AMI セルフサービス出品 (バージョン 2) の販売者体験の更新	の販売者は AWS Marketplace 、単一 AMI 製品のセルフサービスリストを作成できるようになりました。販売者は、処理に余計な時間を費やすことなく、自分で更新を行うことができます。	2022 年 12 月 14 日

[タグベースの承認の 3 つのポリシーが更新されました](#)

AWS Marketplace タグベースの認証機能用に AWSMarketplaceSellerFullAccess 3 つのポリシー (AWSMarketplaceSellerProductsFullAccess 、 および AWSMarketplaceSellerProductsReadOnly) を更新しました。

2022 年 12 月 9 日

[AWS Marketplace Vendor Insights で販売者のポリシーを更新](#)

AWS Marketplace Vendor Insights 販売AWSVendorInsightsVendorReadOnly 者の マネージドポリシー AWSVendorInsightsAssessorFullAccess と を更新しました。

2022 年 11 月 30 日

[AWS Marketplace Vendor Insights で販売者のアクセスを制御する](#)

AWS Marketplace Vendor Insights に、販売者が利用できるアクションとアクセス許可を説明する新しいトピックを追加しました。

2022 年 11 月 30 日

[AWS Marketplace Vendor Insights の 4 つの管理ポリシーを更新しました](#)

AWS Marketplace Vendor Insights の AWSVendorInsightsAssessorFullAccess 、 AWSVendorInsightsAssessorReadOnly 、 AWSVendorInsightsVendorFullAccess 、 AWSVendorInsightsVendorReadOnly の マネージドポリシー を更新しました。

2022 年 11 月 28 日

[販売者は Amazon EKS でアドオン製品を公開できます](#)

AWS Marketplace と Amazon EKS の統合により、販売者は購入者向けに Amazon EKS コンソールで製品を提示できます。

2022 年 11 月 28 日

[AWS Marketplace Vendor Insights の設定を更新しました](#)

AWS Marketplace Vendor Insights のセットアップ手順を更新しました。

2022 年 11 月 18 日

[AWS Marketplace Vendor Insights の 2 つのポリシーを更新](#)

AWS Marketplace Vendor Insights AWSMarketplaceSellerFullAccess の 2 つのポリシー AWSMarketplaceSellerProductsFullAccess とを更新しました。

2022 年 7 月 26 日

[Vendor AWS Marketplace Insights にソフトウェアリスク評価を提供する機能に関する 2 つのポリシーを追加しました。](#)

Vendor Insights AWSVendorInsightsVendorReadOnly にソフトウェアリスク評価を提供する機能である 2 AWS Marketplace つのポリシーAWSVendorInsightsVendorFullAccess とを追加しました。

2022 年 7 月 26 日

[AWS Marketplace Vendor Insights は、に追加された新機能です。AWS Marketplace](#)

AWS Marketplace Vendor Insights はソフトウェアリスク評価を提供する機能です。AWS Marketplace Vendor Insights Vendor Insights はソフトウェアリスク評価を提供する機能です。

2022 年 7 月 26 日

AWS Marketplace Commerce Analytics Service のアクセス許可の更新	AWS Marketplace Commerce Analytics Service には追加の IAM アクセス許可がありません。	2022 年 7 月 21 日
「販売者配送データフィードサービス」セクションが追加されました	ドキュメントのみを更新し、「販売者配送データフィードサービス」セクションを追加し、データフィードに関連するセクションを再編成しました。	2022 年 6 月 15 日
補足レポートのセクションの追加	最近の機能起動用に AWS Marketplace を提供する補足レポート用の新しいセクションを追加しました。	2022 年 6 月 14 日
で SaaS 無料トライアルが販売者に利用可能に AWS Marketplace	販売者は、無料トライアルの期間、トライアル期間中に利用できるディメンション、顧客が受け取る無料使用容量を定義することにより、追加の開発作業なしで無料トライアルを作成できるようになりました。	2022 年 5 月 31 日
購入者と販売者の取引にメール通知が追加されました	で行われたオファーと契約を検証する購入者と販売者への E メール通知を可能にする新機能 AWS Marketplace。	2022 年 5 月 23 日
機械学習製品リストの作成に例が追加されました	機械学習セクションのドキュメントのみを更新し、機械学習製品を作成する際の販売者と購入者の見解を比較する例を追加しました。	2022 年 4 月 22 日

機械学習セクションが更新されました	機械学習セクションの更新は、手順を明確にするため、ドキュメントのみの更新が行われました。	2022年4月15日
現在、イスラエルは適格な管轄区域となっています	イスラエルの居住者が、AWS Marketplaceの販売者としての登録対象になりました。	2022年4月13日
オファーデータフィードの新しい項目について更新されました	メーカーは、チャンネルパートナーが作成したオファーのオファー情報を受け取るようになりました。これには、該当するアカウントが登録販売者であり、オファーのメーカーでもある場合のオファーも含まれます。	2022年3月29日
再販オポチュニティの通知	販売者は再販商談の通知を受け取ることができるようになりました。	2022年3月28日
プロフェッショナルサービス製品に動画を追加しました	プロフェッショナルサービス製品ページを更新し、サービス製品の管理方法の詳細を動画にまとめました。	2022年2月24日
サーバーレス SaaS 統合ソリューションのデプロイに関する新しいトピック	サーバーレス SaaS デプロイの統合に関する新しい情報が追加されました。これには、デプロイ手順に関する参照用の AWS クイックスタート へのリンクが含まれます。	2022年2月15日
コンテナベースの要件と AMI セクションへの最小限の更新	コンテナベースの要件のポリシーの更新を最小限に抑え、AMI 価格契約に関する誤った情報を削除しました	2022年2月14日

コンテナバージョンニングの更新	コンテナイメージやその他のアーティファクトをリポジトリにプッシュする方法を明確にするための、ドキュメントのみの更新です。	2022 年 2 月 10 日
SaaS 製品の ResolveCustomer コード例の更新	SaaS 製品の ResolveCustomer コード例が更新され、CustomerAWSAccount ID が含まれるようになりました。	2022 年 2 月 3 日
Containers Anywhere 製品 AWS Marketplace 用の AWS License Manager との統合に関するドキュメントを追加	License Manager と統合することで、Containers Anywhere 製品の に契約料金を追加するための詳細なガイダンスを追加 AWS Marketplace するためのドキュメントのみの更新。	2022 年 2 月 1 日
SaaS SNS 通知の更新	SaaS 通知メッセージを明確にするためのドキュメントのみの更新です。	2022 年 1 月 25 日
販売者が Amazon Web Services EMEA SARL を通じて EMEA に拠点を置く購入者と取り引きする機能	対象となる AWS Marketplace 販売者は、Amazon Web Services EMEA SARL を通じて、欧州、中東、アフリカ (EMEA) の国と地域に拠点を置く AWS アカウント 顧客と取引できるようになりました。	2022 年 1 月 7 日

[Helm チャートの配送方法を使用してコンテナベースの商品の配送オプションを作成する方法に関するドキュメントを追加しました](#)

販売者は Helm チャートの配送方法で配送オプションを提供できるようになりました。購入者は、販売者が提供する Helm チャートを起動環境にインストールすることにより、これらの配送オプションを使用してコンテナベースのアプリケーションを起動できます。Helm チャート配信方法を提供する場合、販売者は購入者 QuickLaunch に対して有効にできます。QuickLaunch は、購入者が新しい Amazon EKS クラスターをすばやく作成し、そのクラスターでコンテナベースのアプリケーションを起動 AWS CloudFormation するために使用できる機能です。

2021 年 11 月 29 日

[既存のポリシーに対する更新](#)

AWS Marketplace 販売者のセキュリティポリシーが更新されました。

2021 年 11 月 22 日

[AMI およびコンテナベースの製品の契約料金](#)

独立系ソフトウェアベンダー (ISV) は、AMI ベースの新製品またはコンテナベースの製品を出品し、購入者に前払いの契約料金を提示できるようになりました。

2021 年 11 月 17 日

[ベンダー計測タグ付け](#)

コード例を含む、ベンダー計測タグ付けに関するドキュメントのみの更新です。

2021 年 11 月 11 日

[AMI またはコンテナ製品用の Amazon Simple Notification Service](#)

独立系ソフトウェアベンダー (ISV) は、顧客が Amazon 簡易通知サービスを通じて AMI またはコンテナ製品をサブスクライブまたはサブスクライブ解除したときに通知を受け取ることができます。

2021 年 11 月 10 日

[新しい販売者アクセス許可](#)

AWS Marketplace は、のオファーとパートナータブにアクセスするための新しいアクセス許可を追加しました AWS Marketplace 管理ポータル。

2021 年 11 月 9 日

[機械学習製品の非同期推論用に設定されたエンドポイントをデプロイできるようになりました](#)

ペイロード推論が最大数よりも高いことを期待する機械学習ソフトウェア、または呼び出しあたりの最大処理時間を超える処理時間を必要とする機械学習ソフトウェアの場合、購入者は Amazon SageMaker 非同期推論用に設定されたエンドポイントをデプロイできます。

2021 年 11 月 8 日

[返金ポリシーと承認](#)

返金ポリシーを明確にし、すべての返金情報を AWS Marketplace 販売者ガイドの 1 か所にまとめるためのドキュメントのみの更新です。

2021 年 8 月 20 日

コンサルティングパートナーのプライベートオファアのEULAを選択またはアップロードする	独立系ソフトウェアベンダーは、コンサルティングパートナーに再販の機会を創出する際に、エンドユーザー使用許諾契約 (EULA) を選択またはアップロードできるようになりました。	2021年8月17日
SaaS 契約製品のカスタム製品ディメンション	独立系ソフトウェアベンダー (ISV) は、コンサルティングパートナーに再販機会を創出する際に、SaaS 契約製品の規模をカスタマイズできるようになりました。	2021年8月17日
AWS Marketplace フィールドデモンストレーションプログラム	AWS Marketplace フィールドデモンストレーションプログラムの AWS Data Exchange データセット製品の要件を明確にするためのドキュメントのみの更新。	2021年8月3日
SaaS 製品ガイドラインの更新	SaaS 製品の製品ガイドラインが更新されました。	2021年7月29日
コンテナベースの製品要件の更新	コンテナベースの製品要件が更新されました。	2021年7月29日
AMI セキュリティポリシーの更新	AMI 製品のセキュリティポリシーが更新されました。	2021年7月29日
対象管轄区域の追加	AWS Marketplace 香港特別行政区とカタールでの販売者になる資格が付与されました。	2021年6月23日

データフィードの概要	販売者が利用できるデータフィードの構造の概要を説明するドキュメントのみの更新。	2021年6月23日
「機械学習」の章を更新しました	機械学習製品の作成と保守に関する情報のドキュメントのみの更新。	2021年5月27日
コンテナ製品のセルフサービスアップデート	販売者は、AWS Marketplace 管理ポータルを使用して、コンテナベースの商品をより簡単かつ迅速に更新できるようになりました。	2020年12月17日
プロフェッショナルサービス	販売者は AWS Marketplace、購入者にプロフェッショナルサービスを提供できるようになりました。ドキュメントにプロフェッショナルサービスのセクションを追加しました。	2020年12月3日
AMI 製品のセルフサービスアップデート	販売者は、AWS Marketplace 管理ポータルを使用して Amazon マシンイメージ (AMI) ベースの商品をより簡単かつ迅速に更新できるようになりました。	2020年11月23日
対象管轄区域の追加	AWS Marketplace バーレーン、ノルウェー、スイス、およびアラブ首長国連邦 (UAE) で販売される資格が付与されました。	2020年6月17日

<u>承諾済みのプライベートオファ어의アップグレードや更新を提供できます</u>	SaaS 契約製品および従量課金制の SaaS 契約製品については、承諾済みのプライベートオファ어に対するアップグレードや更新をプライベートオファ어として提供できません。	2020 年 5 月 28 日
<u>詳細はデータフィードで利用できます。</u>	データの検索と分析を簡素化するために、レポートの詳細情報を小さいデータフィードに分割します。	2020 年 5 月 21 日
<u>標準化されたライセンス条項が利用可能になりました</u>	カスタム EULA の代わりに標準化されたライセンス条項を提供して契約プロセスを簡素化できます。	2020 年 4 月 28 日
<u>オーストラリアとニュージーランドは対象の管轄区域です</u>	現在、(i) オーストラリア AWS Marketplace(AU) ニュージーランド (NZ) の永住者および市民、または (ii) それらの地域のいずれかで組織または法人化された事業体が で販売される資格があります。	2020 年 4 月 2 日
<u>コンテナ製品でカスタム計測と料金設定の拡張がサポートされるようになりました</u>	独自の料金単位を定義し、その使用量を請求用に計測する場合は、Metering Service AWS Marketplace の meterUsage アクションと統合します。	2019 年 11 月 14 日
<u>AWS Marketplace が AWS Data Exchange を通じてデータ製品をサポート</u>	AWS Marketplaceでデータ製品を提供できるようになりました。	2019 年 11 月 13 日

AWS Marketplace Catalog API サービスの紹介	AWS Marketplace Catalog API サービスは、承認された販売者がプログラムで製品を管理するための API インターフェイスを提供します。	2019 年 11 月 12 日
AWS Marketplace が時間単位の有料コンテナをサポート	AWS Marketplace は、Amazon Elastic Kubernetes Service (Amazon EKS) で実行されている有料の時間単位コンテナをサポートするようになりました。	2019 年 9 月 25 日
更新された AMI 製品の機能。	を使用して AMIs と Lambda 関数を一緒にデプロイできるようになりました AWS CloudFormation。	2019 年 9 月 11 日
セキュリティセクションが追加されました	新しいセキュリティセクションの下にセキュリティコンテンツを統合しました。	2019 年 5 月 7 日
AMI Security Policies を更新しました	AMI 製品のセキュリティポリシーを更新しました。	2019 年 4 月 11 日
バージョニング情報を機械学習製品セクションに追加しました	機械学習製品の製品バージョンの機能について説明するコンテンツを追加しました。	2019 年 3 月 21 日
機械学習製品セクションを追加しました	機械学習製品の公開に関するコンテンツを追加しました。	2018 年 11 月 28 日
コンテナベースの製品セクションを追加しました	コンテナベース製品の公開に関するコンテンツを追加しました。	2018 年 11 月 27 日

販売者のヘルプリクエスト送信のリンクを更新しました	E メールアドレスをウェブフォームアドレスに変更しました。	2018 年 10 月 22 日
消費コンテンツに関する SaaS 契約を追加しました	SaaS コンテンツを再構築し、消費機能付きの SaaS 契約のリリースをサポートするためのコンテンツを追加しました。	2018 年 10 月 18 日
プライベートオファー用の Flexible Payment Schedule に関するコンテンツを追加しました	プライベートオファーコンテンツ用の Flexible Payment Scheduler のリリースをサポートするためのコンテンツを追加しました。	2018 年 10 月 15 日
IAM アクセス許可のコンテンツを更新しました	AMMP 読み取り専用アクセス権のための新しい IAM のアクセス許可をサポートするためのコンテンツを追加しました。	2018 年 10 月 9 日
コンサルティングパートナーのプライベートオファーに関するコンテンツを追加しました	コンサルティングパートナーのプライベートオファー機能をサポートするためのコンテンツを追加しました。	2018 年 10 月 9 日
プライベートイメージビルドに関するコンテンツを追加しました	AMI 機能用の Private Image Build のリリースをサポートするためのコンテンツを追加しました。	2018 年 8 月 13 日
販売者の検索エンジン最適化のガイダンスを追加しました。	検索のために製品の最適化を望む販売者向けのガイダンスを追加しました。	2018 年 7 月 3 日

[AWS Marketplace のロゴへのリンクを更新しました](#)

AWS Marketplace の新しいロゴを指すようにリンクを更新しました。

2018 年 12 月 6 日

[販売者向けガイドを追加しました](#)

PDF 版の販売者向けガイドをすべてオンラインコンテンツに変換しました。

2018 年 5 月 9 日

AWS 用語集

最新の AWS 用語については、「AWS の用語集 リファレンス」の[AWS 「用語集」](#)を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。