MuleSoft.MCPA-Level-1.v2024-11-19.q91

試験コード:	MCPA-Level-1
試験名称:	MuleSoft Certified Platform Architect - Level 1
認定資格:	MuleSoft
無料問題数:	91
バージョン:	v2024-11-19
アクセス数:	496
ページビュー数:	910
https://www.jpnpdf.com/MuleSoft.MCPA-Level-1.v2024-11-19.q91-mondaishu.html	

最新問題: **1**
大量の統合ロジックを含み、製品 API の呼び出しを伴う注文 API を設計する必要があります。製品 API は組織全体で頻繁に使用され、CTO のオフィスにある専用の開発チームによって開発されているため、注文 API と製品 API 間の力関係は 顧客/サプライヤー」の関係になります。
Order API 内で Product API の API データ モデルを処理するには、どのような戦略を使用する必要がありますか?
A. Order API に、Product API データ モデルを Order API の内部データ型に変換する破損防止レイヤーを実装します。
B. Order API の統合ロジックを実装するときに、Product API の API データ型を直接操作して、Order API が Product API と同じ (変更されていない) データ型を使用するようにします。
C. 製品 API の開発チームに、注文 API の API データ モデルを採用するよう説得し、注文 API の統合ロジックが 1 つの一貫した内部データ モデルで動作できるようにします。
D. 組織全体のデータ モデリング イニシアチブを開始し、製品 API と注文 API の両方で使用されるエンタープライズ データ モデルを作成します。
Answer: B (メーッセージを残す**)**

最新問題: **2**
一部の HTTP リクエストに対する応答は、リクエストで使用される HTTP 動詞に応じてキャッシュできます。HTTP 仕様によると、どの HTTP 動詞に対してこれを安全に実行できますか?
A. PUT、POST、DELETE
B. GET、HEAD、POST
C. GET、PUT、オプション
D. GET、オプション、HEAD
Answer: (解答を表示する)
正解: GET、OPTIONS、HEAD
http://restcookbook.com/HTTP%20Methods/べき等性/

最新問題: **3**
ある組織が、OrderStatus システム API の新しい実装を CloudHub の複数のワーカーにデプロイしています。この API は組織のオンプレミスの注文管理システムの前面に配置されており、API 実装によって IPsec トンネル経由でアクセスされます。
通常、OrderStatus システム API のサービス停止を引き起こさないエラーの種類は何ですか?
A. CloudHub ワーカーがメモリ不足例外で失敗する
B. API 実装の初期展開中に API Manager が長時間停止しました
C. 組織のオンプレミス データ センターのネットワーク障害により、注文管理システムにアクセスできません。
D. 関連するAWSデータセンターへの大規模なネットワーク障害によりAWSリージョンがオフラインになる
Answer: A (メーッセージを残す**)**

最新問題: **4**
API 実装は CloudHub 上の単一のワーカーにデプロイされ、外部 API クライアント (CloudHub 外) によって呼び出されます。その API 実装が API 呼び出しに応答しなくなったらすぐにトリガーされることが保証されるアラートを設定するにはどうすればよいでしょうか。
A. 指定された期間内に API がリクエストを受信しなかった場合にアラートを作成します。
B. Anypoint Runtime Manager で 「ワーカーが応答しない」アラートを設定する
C. API 内にハートビート/ヘルスチェックを実装し、Anypoint プラットフォームの外部から呼び出して、ハートビートが応答しない場合に警告を発します。
D. 呼び出し元の API クライアント内で API 呼び出し例外を処理し、API が利用できない場合にその API クライアントからアラートを発生させます。
Answer: D (メーッセージを残す**)**

最新問題: **5**
プロセス API に適用される可能性が最も低い API ポリシーは何ですか?
A. カスタム回路ブレーカー
B. クライアントIDの強制
C. レート制限
D. JSON 脅威保護
Answer: (解答を表示する**)**
JSON脅威保護
****
事実: 技術的には、どのレイヤーにどのポリシーを適用できるかについての制限はありません。どのレイヤー API にも、どのポリシーも適用できます。ただし、API にポリシーを盲目的に適用する前に、コンテキストも適切に考慮する必要があります。
そのため、この質問では、プロセス API に適用される可能性が最も低いポリシーを求めました。
与えられたオプションから:

>> 「JSON 脅威保護」を除くすべてのポリシーは、プロセス層の API にためらうことなく適用できます。
>> JSON 脅威保護ポリシーは、外部 API クライアントからの疑わしい JSON ペイロードを防ぐためのエクスペリエンス API に最適です。これにより、エクスペリエンス API を呼び出す外部クライアントからの悪意のある、または有害な可能性のある JSON ペイロードを回避することで、セキュリティの側面がより強化されます。
外部 API クライアントがプロセス API を直接呼び出すことは決して許可されず、また、このような悪意のある有害な JSON ペイロードは常にこのポリシーを使用するエクスペリエンス API レイヤーでのみ停止されるため、同じポリシーがプロセス レイヤー API に再度適用される可能性は最も低くなります。

最新問題: **6**
展示品を参照してください。
3 つのビジネス プロセスを実装する必要があり、実装では複数の異なる SaaS アプリケーションと通信する必要があります。
これらのプロセスは、個別の (サイロ化された) LOB によって所有され、主に互いに独立していますが、いくつかのビジネス エンティティを共有しています。各 LOB には 1 つの開発チームと独自の予算があります。この組織のコンテキストでは、データ モデルの冗長性を最小限に抑えてこれらのビジネス プロセスを実装する API の API データ モデルを選択する最も効果的な方法は何ですか。
A) ビジネスプロセスの一貫した部分と関連するビジネスエンティティの定義に一致する複数の境界付きコンテキストデータモデルを構築する
B) 確立されたマイクロサービスとアジャイルAPI中心のプラクティスに従うために、各APIごとに異なるデータモデルを構築する
C) XMLスキーマを使用してすべてのAPIデータモデルを構築し、組織全体で一貫性と再利用性を高める
D) 3つのビジネスプロセスのすべてのデータタイプを統合し、データモデルの一貫性と冗長性を確保した、集中型の標準データモデル（エンタープライズデータモデル）を構築する
A. オプションA
B. オプションB
C. オプションC
D. オプションD
Answer: A (メーッセージを残す**)**
正解: ビジネス プロセスの一貫した部分と関連するビジネス エンティティの定義に一致する境界付きコンテキスト データ モデルをいくつか構築します。
****
>> XML スキーマ/アジャイル API 中心のプラクティスを使用して API データ モデルを構築するオプションは、質問で示されているシナリオとは無関係です。したがって、これら 2 つは無効です。

>> チームと LOB がサイロで作業し、それぞれが異なるイニシアチブ、予算などを持っているため、EDM (エンタープライズ データ モデル) の構築は実行可能ではなく、このシナリオには適していません。EDM の構築には、チーム全体の間での徹底的な調整が必要ですが、このシナリオでは明らかに不可能と思われます。
したがって、このシナリオに最適なのは、ビジネス プロセスの一貫した部分と関連するビジネス エンティティの定義に一致する複数の境界付きコンテキスト データ モデルを構築することです。

最新問題: **7**
Anypoint Platform が提供する API 呼び出しメトリクスは何を提供しますか?
A. 過去の API 呼び出しに関するデータ。さまざまな API の異常や使用パターンの特定に役立ちます。
B. 再利用レベルに基づくアプリケーションネットワークの有効性の測定
C. 特定の脅威しきい値を超える可能性のある将来のポリシー違反を積極的に特定する
D. ビジネス ユーザーと直接共有できる API からの ROI メトリック
Answer: (解答を表示する)

最新問題: **8**
API 実装における自動検出の使用を最もよく説明するものは何ですか?
A. Anypoint Exchangeが資産を発見し、再利用できるようにします。
B. Anypoint AnalyticsがAPIの使用状況を把握できるようになります
C. Anypoint StudioがAnypoint Platformで設定されたAPI定義を検出できるようにします。
D. API ManagerがAPI実装を認識し、ポリシーを適用できるようにします。
Answer: (解答を表示する)

最新問題: **9**
コード中心の API ドキュメント環境では、API コンシューマーが、代表的なシナリオの一部として 1 つ以上の API の呼び出しを示す API クライアント ソース コードを調査および実行できるようにする必要があります。
Anypoint Platform を使用して、このようなコード中心の API ドキュメント環境を提供する最も効果的な方法は何ですか?
A. APIがAnypoint ExchangeエントリとAPIコンソールを通じて適切に文書化されていることを確認し、これらのページをすべてのAPIコンシューマーと共有します。
B. APIノートブックを作成し、関連するAnypoint Exchangeエントリに含めます。
C. Anypoint Exchangeエントリを介して関連するAPIを検出可能にする
D. 関連するAPIごとにモックサービスを有効にし、Anypoint Exchangeエントリ経由で公開します。
Answer: (解答を表示する)

最新問題: **10**

展示を参照してください。組織は Mule スタンドアロン ランタイムを実行しており、Active Directory を Anypoint Platform 外部 ID プロバイダーとして構成しています。組織には他のシステム コンポーネントのための予算がありません。
特定の内部ユーザー グループへのアクセスを最も効果的に制限するには、組織内のすべての API インスタンスにどのようなポリシーを適用する必要がありますか?
A. 基本認証 - LDAP ポリシーを適用します。内部 Active Directory がユーザー認証用の LDAP ソースとして構成されます。
B. クライアントID強制ポリシーを適用します。特定のユーザーグループは、特定のクライアント資格情報を使用するようにクライアントアプリケーションを構成します。
C. IPホワイトリストポリシーを適用します。特定のユーザーのワークステーションのみがホワイトリストに追加されます。
D. OAuth 2.0アクセストークン強制ポリシーを適用します。内部Active DirectoryがOAuthサーバーとして構成されます。
Answer: (解答を表示する**)**
基本認証 - LDAP ポリシーを適用します。内部 Active Directory がユーザー認証用の LDAP ソースとして構成されます。
****
>> IP ホワイトリストはこの目的には適していません。さらに、ユーザーのワークステーションは、ネットワーク内で必ずしも静的 IP を持つとは限りません。
>> OAuth 2.0 の適用には、組織のシステム コンポーネントに含まれていないクライアント プロバイダーが必要です。
>> すべてのユーザーが個別のクライアント資格情報を作成し、使用方法に合わせて構成できるようにするのは効果的なアプローチではありません。
効果的な方法は、基本認証 - LDAP ポリシーを適用することです。内部 Active Directory は、ユーザーを認証するための LDAP ソースとして構成されます。

最新問題**: 11**
すべてのデータ処理を特定の管轄区域（米国や EU など）内で実行することを要求する法的規制に対処する場合の API 実装について正しいのは何ですか?
A. Anypoint MQではなく、Active MQなどの管轄地域の外部メッセージングシステムを使用する必要があります。
B. 転送中も保存中もすべてのデータが暗号化されていることを確認する必要があります。
C. オブジェクトストアは米国東部地域にのみデプロイされたサービスに依存しているため、使用を避ける必要があります。
D. これらは、Anypoint Platform コントロールプレーンによって管理される Anypoint Platform ランタイムプレーンにデプロイされ、両方のプレーンが同じ管轄内にある必要があります。
Answer: (解答を表示する**)**

最新問題**: 12**

API 実装を Anypoint VPC にデプロイする必要があるのはいつですか?
A. API実装が永続オブジェクトストアに書き込む必要がある場合
B. API実装が、顧客管理のAWSインスタンス内のCloudHubの外部にデプロイされた公開サービスを呼び出す必要がある場合
C. API実装をMule Mavenプラグインを使用して本番AWS VPCにデプロイする必要がある場合
D. API実装が、パブリックアクセスを許可しない制限された顧客ホストネットワークのサブネット内でアクセス可能である必要がある場合
Answer: B (メーッセージを残す**)**

最新問題**: 13**
Anypoint VPC のテクノロジー アーキテクチャについて正しいのは何ですか?
A. Anypoint VPCのプライベートIPアドレス範囲はCloudHubによって自動的に選択されます
B. Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックは、プライベート ネットワーク内にとどまることができます。
C. 各CloudHub環境には個別のAnypoint VPCが必要です
D. VPC ピアリングを使用すると、基盤となる AWS VPC をオンプレミス (非 AWS) のプライベートネットワークにリンクできます。
Answer: B (メーッセージを残す**)**
正解: Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックは、プライベート ネットワーク内にとどまることができます。
****
>> Anypoint VPC のプライベート IP アドレス範囲は、CloudHub によって自動的に選択されるわけではありません。CIDR ブロックを使用して VPC を作成するときに、弊社によって選択されます。
CIDR ブロック: クラスレス ドメイン間ルーティング (CIDR) 表記での Anypoint VPC のサイズ。たとえば、10.111.0.0/24 に設定すると、Anypoint VPC には 10.111.0.0 から 10.111.0.255 までの 256 個の IP アドレスが付与されます。
理想的には、Anypoint VPC に選択する CIDR ブロックはプライベート IP スペースからのものであり、他の Anypoint VPC の CIDR ブロックや企業ネットワークで使用されている CIDR ブロックと重複してはなりません。
各 CloudHub 環境には個別の Anypoint VPC が必要です。Anypoint VPC が作成されると、複数の環境で同じ VPC を選択できます。ただし、一般的には、非本番環境と本番環境に常に個別の Anypoint VPC を用意することがベスト プラクティスであり、推奨されます。
>> 基盤となる AWS VPC をオンプレミス (非 AWS) のプライベート ネットワークにリンクするために Anypoint VPN を使用します。VPC ピアリングではありません。
参照：
与えられた選択肢の中で唯一正しいのは、Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックがプライベート ネットワーク内に留まることができるということです。
https://docs.mulesoft.com/runtime-manager/vpc-connectivity-methods-concept

最新問題: **14**
CloudHub 専用ロードバランサーを使用する必要がある条件は何ですか?
A. 複数の CloudHub ワーカー間での API 呼び出しを負荷分散する必要がある場合
B. 顧客がホストする Mule ランタイムにデプロイされた API 実装にカスタム DNS 名が必要な場合
C. 同じ Mule アプリケーションの別々のデプロイメント間でクロスリージョン負荷分散が必要な場合
D. API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
Answer: B (メーッセージを残す**)**

最新問題: **15**
特定のビジネス プロセスを実装するために、粗粒度ではなく細粒度の API デプロイメント モデルを使用した場合の一般的な結果は何ですか。
A. ビジネスプロセスをサポートするアプリケーションネットワーク内の接続数の減少
B. アプリケーションネットワーク内で検出可能なAPI関連アセットの数の増加
C. API の範囲と複雑さが小さくなるため、エンドユーザーへの応答時間が短縮されます。
D. 細粒度APIごとに消費するリソースが少なくなるため、タワー全体のリソース使用量が削減されます。
Answer: B (メーッセージを残す**)**
正解: アプリケーション ネットワーク内で検出可能な API 関連アセットの数が増える。
****
>> 粗粒度のアプローチと比較した場合、細粒度のアプローチでは応答時間が速くなりません。
>> 実際、粒度の粗い API モデルを持つネットワークでは、粒度の細かい API モデルを持つネットワークよりも応答時間が短くなります。その理由は次のとおりです。
きめ細かなアプローチ:
1. 粗粒度に比べてAPIの数が多くなる
2. したがって、ビジネス プロセスの機能性を実現するには、さらにオーケストレーションを行う必要があります。
3. つまり、多くの API 呼び出しが必要になります。そのため、より多くの接続を確立する必要があります。したがって、大量の機能が組み込まれた API が少ない粗粒度のアプローチと比較すると、ホップ、ネットワーク I/O、統合ポイントの数が増えるのは明らかです。
4. そのため、これらすべての追加ホップと追加されたレイテンシにより、細粒度のアプローチでは、粗粒度のアプローチに比べて応答時間が少し長くなります。
5. レイテンシと接続が追加されるだけでなく、API の数が増えるため、きめ細かいアプローチで使用されるリソースも増えます。
そのため、きめ細かな API は、ネットワーク内で再利用可能な資産をより多く公開し、検出できるようにするという点で優れています。ただし、ネットワーク ホップと応答時間に関して多少の妥協をしながら、統合ポイント、接続、リソースを管理するために、より多くのメンテナンスが必要になります。

最新問題: **16**
Anypoint Platform が提供する API 呼び出しメトリクスは何を提供しますか?
A. ビジネス ユーザーと直接共有できる API からの ROI メトリック
B. 再利用レベルに基づくアプリケーションネットワークの有効性の測定
C. 過去の API 呼び出しに関するデータ。さまざまな API の異常や使用パターンの特定に役立ちます。
D. 特定の脅威しきい値を超える可能性のある将来のポリシー違反を積極的に特定する
Answer: B (メーッセージを残す**)**
説明/参照:
https://usermanual.wiki/Document/APAAppNetstudentManual02may2018.991784750.pdf

最新問題: **17**
組織では、さまざまなクラウドベースの SaaS システムと複数のオンプレミス システムを使用しています。オンプレミス システムは組織のアプリケーション ネットワークの重要な部分であり、組織のイントラネット内からのみアクセスできます。
クラウドベースの SaaS システムとオンプレミス システムの両方との統合をサポートするために Anypoint Platform を構成して使用する最適な方法は何ですか?
A) Anypoint Platform Private Cloud Edition コントロールプレーンによって管理される Anypoint VPC で CloudHub でデプロイされた Mule ランタイムを使用する
B)MuleSoft がホストする Anypoint Platform コントロール プレーンによって管理される共有ワーカー クラウドで、CloudHub でデプロイされた Mule ランタイムを使用する
C) Anypoint Platform Private Cloud Edition コントロールプレーンによって管理され、外部ネットワークアクセスが一切ない完全に分離されたオンプレミスの Mule ランタイムのインストールを使用します。
D)MuleSoft がホストする Anypoint Platform コントロール プレーンによって管理される、CloudHub でデプロイされたオンプレミス Mule ランタイムと手動でプロビジョニングされたオンプレミス Mule ランタイムの組み合わせを使用する
A. オプションA
B. オプションB
C. オプションC

D. オプションD
Answer: B (メーッセージを残す**)**
CloudHubでデプロイされたMuleと手動でプロビジョニングされたオンプレミスMuleを組み合わせて使用する
MuleSoft がホストするプラットフォーム コントロール プレーンによって管理されるランタイム。
****
与えられたシナリオから得られる重要な詳細:
>> 組織はクラウドベースとオンプレミスの両方のシステムを使用しています
>> オンプレミスのシステムには、組織のイントラネット内からのみアクセスできます。上記の重要な詳細に基づいて、与えられた選択肢を評価してみましょう。
>> CloudHub でデプロイされた Mule ランタイムは、MuleSoft がホストするコントロール プレーンを使用してのみ制御できます。Private Cloud Edition のコントロール プレーンを使用して CloudHub Mule ランタイムを制御することはできません。したがって、これを提案するオプションは無効です。
>> MuleSoft がホストする Anypoint Platform によって管理される共有ワーカー クラウドで CloudHub でデプロイされた Mule ランタイムを使用することは、与えられたシナリオとはまったく無関係であり、愚かな選択です。したがって、これを提案するオプションは無効です。
>> Anypoint Platform Private Cloud Edition コントロール プレーンによって管理され、外部ネットワーク アクセスなしで完全に分離された Mule ランタイムのオンプレミス インストールを使用すると、オンプレミス統合が機能します。ただし、外部アクセスがない場合、SaaS ベースのアプリとの統合は実行できません。さらに、CloudHub でホストされるアプリは、SaaS ベースのアプリケーションとの統合に最適です。したがって、これを提案するオプションが最善の方法です。
これらの混合/ハイブリッド統合をサポートするために Anypoint Platform を設定して使用する最適な方法は、MuleSoft がホストする Platform コントロール プレーンによって管理される、CloudHub でデプロイされたオンプレミス Mule ランタイムと手動でプロビジョニングされたオンプレミス Mule ランタイムを組み合わせて使用することです。

最新問題**: 18**
Anypoint VPC のテクノロジー アーキテクチャについて正しいのは何ですか?
A. Anypoint VPCのプライベートIPアドレス範囲はCloudHubによって自動的に選択されます
B. Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックは、プライベート ネットワーク内にとどまることができます。
C. 各CloudHub環境には個別のAnypoint VPCが必要です
D. VPC ピアリングを使用すると、基盤となる AWS VPC をオンプレミス (非 AWS) のプライベートネットワークにリンクできます。
Answer: B (メーッセージを残す**)**
Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックは、プライベート ネットワーク内にとどまることができます。
****

>> Anypoint VPC のプライベート IP アドレス範囲は、CloudHub によって自動的に選択されるわけではありません。CIDR ブロックを使用して VPC を作成するときに、弊社によって選択されます。
CIDR ブロック: クラスレス ドメイン間ルーティング (CIDR) 表記での Anypoint VPC のサイズ。たとえば、10.111.0.0/24に設定すると、Anypoint VPCには10.111.0.0から256個のIPアドレスが付与されます。
10.111.0.255。
理想的には、Anypoint VPC に選択する CIDR ブロックはプライベート IP スペースからのものであり、他の Anypoint VPC の CIDR ブロックや企業ネットワークで使用されている CIDR ブロックと重複してはなりません。
各 CloudHub 環境には個別の Anypoint VPC が必要です。Anypoint VPC が作成されると、複数の環境で同じ VPC を選択できます。ただし、一般的には、非本番環境と本番環境に常に個別の Anypoint VPC を用意することがベスト プラクティスであり、推奨されます。
>> Anypoint VPN を使用して、基盤となる AWS VPC をオンプレミス (非 AWS) のプライベート ネットワークにリンクします。
VPC ピアリングではありません。

最新問題**: 19**
展示品を参照してください。
顧客がホストする Mule ランタイムを MuleSoft がホストする Anypoint Platform コントロールプレーン (ハイブリッド展開) と併用する場合、正しいのは何ですか?
A. Anypoint Runtime Manager は、Mule アプリケーションをデプロイするために Mule ランタイムへのネットワーク接続を開始します。
B. MuleSoft がホストする共有ロードバランサは、Mule ランタイムへの API 呼び出しの負荷分散に使用できます。
C. API実装は、コントロールプレーンと通信できない場合でも、顧客がホストするMuleランタイムで正常に実行できます。
D. Anypoint Runtime Manager は、ノード障害が発生した場合に新しい Mule ランタイムインスタンスを作成することで、コントロールプレーンの HA を自動的に確保します。
Answer: C (メーッセージを残す**)**
API 実装は、コントロール プレーンと通信できない場合でも、顧客がホストする Mule ランタイムで正常に実行できます。
****
>> 顧客がホストするランタイム上のAPIの負荷分散に共有ロードバランサを使用することはできません
>> ハイブリッド展開モデルの場合、オンプレミスは最初に Runtime Manager エージェントを使用して Runtime Manager に接続されます。したがって、最初にオンプレミスから Runtime Manager への接続が開始されます。その後、すべての制御は Runtime Manager から実行できます。

>> Anypoint Runtime Manager は自動 HA を保証することはできません。クラスター/サーバー グループなどは事前に構成する必要があります。
与えられた選択肢の中で唯一 TRUE となるのは、コントロール プレーンと通信できない場合でも、API 実装は顧客がホストする Mule ランタイムで正常に実行できるというものです。このステートメントを正当化する参考資料が以下にあります。
参考文献:
https://docs.mulesoft.com/runtime-manager/deployment-strategies#hybrid-deployments
https://help.mulesoft.com/s/article/On-Premise-Runtimes-Disconnected-From-US-Control-Plane-2018 年 6 月 18 日
https://help.mulesoft.com/s/article/Runtime-Manager-cannot-manage-On-Prem-Applications-and-Servers-from-U
https://help.mulesoft.com/s/article/On-premise-Runtimes-Appear-Disconnected-in-Runtime-Manager-May-29th-

最新問題**: 20**
Anypoint Exchange では、API プロデューサーによって、承認されたセマンティック バージョン管理プラクティスに従って API がバージョン 3.1.1 から 3.2.0 に更新され、その変更は API のパブリック ポータルを通じて通知されました。
新しいバージョンでは API エンドポイントは変更されません。
API クライアントの開発者はこの変更にどのように対応すべきでしょうか?
A. 更新はプロジェクトリスクとして識別され、このAPIを使用する機能の完全な回帰テストを実行する必要があります。
B. APIクライアントコードは、新しい機能を利用する必要がある場合にのみ変更する必要があります。
C. APIプロデューサーは、古いバージョンを新しいバージョンと並行して実行するように要求される必要があります。
D. 既存の機能の変更を理解するには、API プロデューサーに連絡する必要があります。
Answer: A (メーッセージを残す**)**

最新問題**: 21**
どの Mule アプリケーションで、その Mule アプリケーションによって公開されるエンドポイントに Anypoint Platform によって API ポリシーを適用できますか?
A. HTTP/1x 経由でリクエストを受け入れる Mule アプリケーション。
B. TCP 経由で JSON リクエストを受け入れるが、応答を提供する必要がない Mule アプリケーション。
C. WebSocket 経由で JSON リクエストを受け入れる Mule アプリケーション。
D. HTTP/2 経由で gRPC リクエストを受け入れる Mule アプリケーション
Answer: D (メーッセージを残す**)**
説明/参照:

最新問題: **22**
API 実装は CloudHub 上の単一のワーカーにデプロイされ、外部 API クライアント (CloudHub 外部) によって呼び出されます。
API 実装が API 呼び出しに応答しなくなったらすぐにトリガーされることが保証されるアラートを設定するにはどうすればよいでしょうか?
A. 呼び出し元の API クライアント内で API 呼び出し例外を処理し、API が利用できない場合はその API クライアントからアラートを発生させます。
B. Anypoint Runtime Manager で 「ワーカーが応答しない」アラートを設定します。
C. 指定された期間内に API がリクエストを受信しなかった場合にアラートを作成します。
D. API 内にハートビート/ヘルスチェックを実装し、Anypoint Platform の外部から呼び出して、ハートビートが応答しない場合にアラートを出します。
Answer: (解答を表示する**)**

最新問題: **23**
ある企業では、EU コントロール プレーンと顧客がホストする Mule ランタイムを組み合わせたハイブリッド Anypoint Platform 展開モデルを使用しています。ステージング環境で Mule API 実装のテストに成功した後、Mule API 実装は環境固有のプロパティで設定され、本番環境に昇格する必要があります。MuleSoft が Mule API 実装を構成し、本番環境への昇格を自動化するために推奨する方法は何ですか?
A. 各環境のプロパティ ファイルを Mule API 実装のデプロイ可能なアーカイブにバンドルし、Anypoint CLI または Anypoint Platform REST API を使用して Mule API 実装を本番環境に昇格します。
B. API マネージャーのプロパティ タブで Mule API 実装のプロパティを変更し、API マネージャーを使用して Mule API 実装を本番環境に昇格します。
C. Anypoint Exchange で Mule API 実装のプロパティを変更し、Runtime Manager を使用して Mule API 実装を本番環境に昇格します。
D. API ポリシーを使用してステージング環境にデプロイされた Mule API 実装のプロパティを変更し、別の API ポリシーを使用して Mule API 実装を本番環境にデプロイします。
Answer: A (メーッセージを残す**)**
正解: 各環境のプロパティ ファイルを Mule API 実装のデプロイ可能なアーカイブにバンドルし、Anypoint CLI または Anypoint Platform REST API を使用して Mule API 実装を本番環境に昇格します。
****
>> Anypoint Exchange は、アセットの検出とドキュメント化を目的としています。Mule API 実装のプロパティを変更するための規定はまったくありません。
>> API マネージャーは、API インスタンス、その契約、ポリシー、SLA を管理するためのものです。また、API 実装のプロパティを変更するための規定もありません。

>> API ポリシーは API の非機能要件に対処するためのものであり、API 実装のプロパティを変更するための規定はありません。
したがって、開発実践の一環としてこれを行う正しい方法および推奨される方法は、各環境のプロパティ ファイルを Mule API 実装にバンドルし、環境ごとにそれぞれのファイルをポイントして参照することです。

最新問題: **24**
展示品を参照してください。
API 主導の接続性とアプリケーション ネットワークの意味で有効な API とは何でしょうか?
A) TCP 経由の Java RMI
B) TCP 経由の Java RMI
C) HOP 経由の CORBA
D) UDP経由のXML
A. オプションA
B. オプションB
C. オプションC
D. オプションD
Answer: D (メーッセージを残す**)**
正解: HTTP 経由の XML
****
>> API 主導の接続性とアプリケーション ネットワークでは、最も効果的な API とネットワークを構築するために、HTTP ベースのプロトコル上に API を配置することが求められています。
>> HTTPベースのAPIにより、プラットフォームはさまざまなポリシーを適用して多くのNFRに対応できます。
>> HTTP ベースの API を使用すると、HTTP ベースの w3c ルールに準拠した多くの標準的で効果的な実装パターンを実装することもできます。

最新問題: **25**
API 主導の接続性の定義に最も適したものは次のどれですか?
A. API 主導の接続性は、単なるアーキテクチャやテクノロジーではなく、組織内で効率的な IT 提供のために人材とプロセスを編成する方法でもあります。
B. API主導の接続性は、エクスペリエンス、プロセス、システム層をカバーする3層アーキテクチャです。
C. API主導の接続性は、エクスペリエンス、プロセス、システム層ベースのAPIを実装することを可能にする技術です。
Answer: (解答を表示する**)**
API 主導の接続性は、単なるアーキテクチャやテクノロジーではなく、組織内で効率的な IT 提供のために人材とプロセスを編成する方法でもあります。
****

最新問題: **26**
ダウンタイムが繰り返し発生することが知られている Order API を呼び出す必要がある API 実装が設計されています。
このため、Order API が利用できない場合は、フォールバック API が呼び出されます。
フォールバック API の呼び出しを設計する際に、どのようなアプローチが最高の回復力を提供しますか?
A. Anypoint Exchangeで適切な既存のフォールバックAPIを検索し、注文APIに加えてこのフォールバックAPIへの呼び出しを実装します。
B. API マネージャーで注文 API の別のエントリを作成し、プライマリ注文 API が利用できない場合にこの API をフォールバック API として呼び出します。
C. Order APIが利用できない場合は、HTTP 307 Temporary Redirectステータスコードを介してクライアントリクエストをフォールバックAPIにリダイレクトします。
D. HTTP リクエスター コンポーネントに、Order API を呼び出すオプションを設定して、Order API から HTTP 4xx または 5xx 応答ステータス コードが返されるたびにフォールバック API を呼び出すようにします。
Answer: A (メーッセージを残す**)**
正解: Anypoint Exchange で適切な既存のフォールバック API を検索し、注文 API に加えてこのフォールバック API への呼び出しを実装します。
****
>> API クライアントが HTTP 3xx 一時リダイレクト ステータス コードを受信し、別の API を呼び出すためにフォールバック ロジックを実装する必要があるという事前承認済みの合意が API クライアントとの間でない限り、これは理想的でも適切なアプローチでもありません。
>> API マネージャーで同じ Order API の別のエントリを作成すると、同じ API 実装の上に別のインスタンスが作成されるだけです。したがって、同じ API のクローンをフォールバック API として使用しても効果はありません。フォールバック API は、理想的にはプライマリ API とは異なる API 実装である必要があります。
>> 現在、Anypoint HTTP Connector では、応答として特定の HTTP ステータス コードを受信したときにフォールバック API を呼び出すことができるオプションは提供されていません。
指定されたオプションの中で TRUE となる唯一のステートメントは、適切な既存のフォールバック API を Anypoint エクスチェンジで検索し、注文 API に加えてこのフォールバック API への呼び出しを実装することです。

最新問題: **27**
API 実装は CloudHub 上の単一のワーカーにデプロイされ、外部 API クライアント (CloudHub 外) によって呼び出されます。その API 実装が API 呼び出しに応答しなくなったらすぐにトリガーされることが保証されるアラートを設定するにはどうすればよいでしょうか。
A. API 内にハートビート/ヘルスチェックを実装し、Anypoint プラットフォームの外部から呼び出して、ハートビートが応答しない場合に警告を発します。
B. Anypoint Runtime Manager で 「ワーカーが応答しない」アラートを設定する

C. 呼び出し元のAPIクライアント内でAPI呼び出し例外を処理し、APIが利用できない場合にそのAPIクライアントからアラートを発生させます。
D. 指定された期間内に API がリクエストを受信しなかった場合にアラートを作成します。
Answer: B (メーッセージを残す**)**
正解: Anypoint Runtime Manager で 「ワーカーが応答していません」アラートを設定します。
****
>> すべてのオプションは、最終的に、アプリケーションが応答を停止したときに必要なアラートを生成するのに役立ちます。
>> ただし、API 呼び出し内で例外を処理し、API クライアントからアラートを発生させるのは不適切で愚かなことです。API 実装を呼び出す API クライアントは多数ある可能性があり、すべてのクライアントでこの設定を一貫して行うことは理想的ではありません。現実的な方法ではありません。
>> API 内でヘルス チェック/ハートビートを実装し、外部から呼び出してヘルスを判定するのは問題ないように思えますが、追加の設定が必要であり、同時に、API 実装でヘルス チェック API を呼び出す外部ツール間で断続的なネットワークの問題が発生すると、誤報が発生する可能性が非常に高くなります。API 実装自体には問題がないかもしれませんが、他の要因により誤報が発生する可能性があります。
>> 指定された期間内に API がリクエストを受信しなかった場合に API Manager でアラートを作成すると、実際に現実的なアラートが生成されますが、API クライアントからのリクエストが実際にない場合にも、誤ったアラートが送信される可能性があります。
この要件を満たす最善かつ正しい方法は、Runtime Manager に 「ワーカーが応答しない」という条件でアラートを設定することです。これにより、ワーカーが応答しなくなるとすぐにアラートが生成されます。

最新問題**: 28**
承認されたセマンティック バージョン管理プラクティスに従って、API プロデューサーによって Anypoint Exchange で API がバージョン 3.1.1 から 3.2.0 に更新され、変更内容が API パブリック ポータルを通じて通知されました。
新しいバージョンでは API エンドポイントは変更されません。API クライアントの開発者はこの変更にどのように対応すればよいでしょうか?
A. 既存の機能の変更を理解するには、API プロデューサーに連絡する必要があります。
B. APIプロデューサーは、新しいバージョンと並行して古いバージョンを実行するように要求される必要があります。
C. APIクライアントは自身のコードを更新し、完全な回帰を行う必要がある
D. APIクライアントコードは、新しい機能を利用する必要がある場合にのみ変更する必要があります。
Answer: D (メーッセージを残す**)**

最新問題**: 29**

システム API はプライマリ環境と災害復旧 (DR) 環境に展開され、各環境で DNS 名が異なります。プロセス API はシステム API のクライアントであり、システム API によってレート制限されており、各環境で制限が異なります。システム API の DR 環境では、プライマリ環境が提供するレート制限の 20% しか提供されません。これらの条件と制約を考慮すると、プロセス API の全体的なエラーを減らすための最適な API フォールト トレラント呼び出し戦略は何でしょうか。
A. プライマリ環境にデプロイされたシステム API を呼び出します。断続的な障害を回避するために、プロセス API にタイムアウトと再試行のロジックを追加します。それでも障害が発生する場合は、DR 環境にデプロイされたシステム API を呼び出します。
B. プライマリ環境にデプロイされたシステム API を呼び出します。DR 環境にデプロイされたシステム API を呼び出すことで、断続的な障害を処理するためにプロセス API に再試行ロジックを追加します。
C. プライマリ環境にデプロイされたシステムAPIとDR環境にデプロイされたシステムAPIを並行して呼び出し、断続的な障害を回避するためにプロセスAPIにタイムアウトと再試行のロジックを追加し、結果を結合するためのロジックをプロセスAPIに追加します。
D. プライマリ環境にデプロイされたシステム API を呼び出します。断続的な障害を回避するために、プロセス API にタイムアウトと再試行ロジックを追加します。それでも障害が発生する場合は、DR 環境にデプロイされたプロセス API のコピーを呼び出します。
Answer: (解答を表示する**)**
正解: プライマリ環境にデプロイされたシステム API を呼び出します。断続的な障害を回避するために、プロセス API にタイムアウトと再試行ロジックを追加します。それでも障害が発生する場合は、DR 環境にデプロイされたシステム API を呼び出します。
****
この質問には、DR 環境のシステム API がプライマリ環境が提供するレート制限の 20% しか提供しないという重要な考慮事項が 1 つあります。したがって、比較すると、プライマリ環境と比較して、DR 環境 API への呼び出しは非常に少なくなります。これを念頭に置いて、適切かつ最適なフォールト トレラント呼び出し戦略を分析しましょう。
1. DR 環境には 20% の制限があるため、両方のシステム API を並行して呼び出すことは絶対に実行可能なアプローチではありません。毎回並行して呼び出すと、DR 環境のレート制限が簡単かつ急速に使い果たされ、必要なときに真の断続的なエラー シナリオが発生する機会がなくなる可能性があります。
2. もう一つのオプションは、プライマリ環境のシステム API を呼び出すときに、プロセス API にタイムアウトと再試行ロジックを追加することを提案することです。これは今のところは良いことです。ただし、すべての再試行が失敗した場合、オプションは DR 環境でプロセス API のコピーを呼び出すことを提案しますが、これは正しくなく、推奨されません。フォールバックの対象となるのはシステム API のみであり、プロセス API 全体ではありません。プロセス API には通常、DR のプロセス API を呼び出して再度繰り返したくない他の多くの API を呼び出す大量のオーケストレーションがあります。したがって、このオプションは正しくありません。
3. もう 1 つのオプションは、最初にプライマリ環境のシステム API を再試行するのではなく、プロセス API に再試行 (タイムアウトなし) ロジックを追加して、DR 環境のシステム API を直接再

試行することを提案することです。これはまったく適切なフォールバックではありません。適切なフォールバックは、最初にプライマリ環境ですべての再試行が実行され、使い果たされた後にのみ発生するはずです。ただし、ここでのオプションは、メイン API を試行せずに、最初の失敗自体でフォールバック API を直接再試行することを提案しています。したがって、このオプションも正しくありません。
これにより、適切かつ最適なオプションが 1 つ残ります。
- プライマリ環境にデプロイされたシステムAPIを呼び出す
- プロセス API にタイムアウトと再試行のロジックを追加します。
- すべての再試行後も失敗した場合は、DR 環境にデプロイされたシステム API を呼び出します。

最新問題: **30**
アップストリーム API とその実装を設計する際、開発チームは、ダウンストリーム API を呼び出すときにタイムアウトを設定しないようアドバイスされています。ダウンストリーム API には信頼できる SLA がないためです。これは、そのアップストリーム API の唯一のダウンストリーム API 依存関係です。
ダウンストリーム API がクラッシュすることなく中断なく実行されると仮定します。このアドバイスの影響は何でしょうか?
A. アップストリームAPIのSLAは提供できません
B. ダウンストリームAPIの呼び出しはタイムアウトせずに完了します
C. アップストリームAPI実装が実行されるMuleランタイムによって、デフォルトのタイムアウト500ミリ秒が自動的に適用されます。
D. 下流のAPI実装が実行されるMuleランタイムによって、1000ミリ秒未満のToad依存のタイムアウトが適用されます。
Answer: (解答を表示する**)**
正解: アップストリーム API の SLA は提供できません。
****
>> まず最初に、HTTP コネクタのデフォルトの HTTP 応答タイムアウトは 10000 ミリ秒 (10 秒) です。500 ミリ秒ではありません。
>> Mule ランタイムは、このような 負荷依存」のタイムアウトを適用しません。現在、Mule にはそのような動作はありません。
>> HTTP コネクタにはデフォルトで 10000 ミリ秒のタイムアウトがあるため、信頼性の低い SLA 時間により、ダウンストリーム API の呼び出しがタイムアウトせずに完了することを常に保証することはできません。応答時間が 10 秒を超えると、リクエストがタイムアウトする可能性があります。
これによる主な影響は、アップストリーム API の適切な SLA を提供できないことです。

最新問題: **31**
どの Mule アプリケーションで、その Mule アプリケーションによって公開されるエンドポイントに Anypoint Platform によって API ポリシーを適用できますか?

A) HTTP/1.x 経由でリクエストを受け入れる Mule アプリケーション
B) TCP経由でJSONリクエストを受け入れるが、応答を返す必要がないMuleアプリケーション
C) WebSocket経由でJSONリクエストを受け入れるMuteアプリケーション
D) HTTP/2 経由で gRPC リクエストを受け入れる Mule アプリケーション
A. オプションD
B. オプションB
C. オプションC
D. オプションA
Answer: D (メーッセージを残す**)**

最新問題**: 32**
組織は、最新の API (MuleSoft の定義による) を使用して再利用可能な IT 資産の消費を重視する IT 運用モデルに移行するという戦略的決定を下します。
この新しい IT 運用モデルに関連して、各最新 API を最もよく表すものは何ですか?
A. 各最新 API は製品のように扱われ、特定の対象ユーザー (たとえば、モバイル アプリ開発者) 向けに設計される必要があります。
B. 各最新 API は REST および HTTP ベースである必要があります。
C. 最新の API にはそれぞれ独自のソフトウェア開発ライフサイクルがあるため、ドキュメント化と自動化の必要性が軽減されます。
D. 最新の API はどれも簡単に使用できる必要があるため、SAML や JWT などの複雑な認証メカニズムは避ける必要があります。
Answer: A (メーッセージを残す**)**

最新問題**: 33**
Anypoint Platform REST API、Anypoint CU、Mule Maven プラグインなどのツールを使用して Anypoint Platform とのやり取りを自動化することについて正しいのは何ですか?
A. Anypoint Platform API と Anypoint CU へのアクセスは、Anypoint Platform のロールと権限を通じて個別に制御できるため、特定のユーザーは Anypoint CLI にアクセスでき、他のユーザーはプラットフォーム API にアクセスできます。
B. Anypoint Platform API は CloudHub とのやり取りのみを自動化できますが、顧客がホストする Mule ランタイムへの展開には Mule Maven プラグインが必要です。

C. デフォルトでは、Anypoint CLI と Mule Maven プラグインは Mule ランタイムに含まれていないため、デプロイされた Mule アプリケーションでは使用できません。
D. APIポリシーをAnypoint Platform APIに適用して、特定のLOBのみが特定の機能にアクセスできるようにすることができます。
Answer: C (メーッセージを残す)
デフォルトでは、Anypoint CLI と Mule Maven プラグインは Mule ランタイムに含まれていないため、デプロイされた Mule アプリケーションでは使用できません。
****
>> カスタムで記述した API インスタンスのように、Anypoint Platform API に API ポリシーを適用することはできません。したがって、これを示唆するオプションは FALSE です。
>> Anypoint Platform API は、CloudHub と顧客がホストする Mule ランタイムの両方とのやり取りを自動化するために使用できます。CloudHub だけではありません。したがって、これに反対するオプションは FALSE です。
>> Mule Maven プラグインは、顧客がホストする Mule ランタイムへのデプロイメントに必須ではありません。CI/CD の自動化をスムーズにするのに役立つだけです。ただし、デプロイメントの必須要件ではありません。したがって、これに反対するオプションは FALSE です。
>> プラットフォームには、一部のユーザーに Anypoint CLI を、他のユーザーに Anypoint Platform API を個別にアクセス制御するための特別なロールや権限はありません。適切な一般的なロール/権限 (API 所有者、Cloudhub 管理者など) があれば、任意のオプション (Anypoint CLI または Platform API) を使用できます。したがって、これを示唆するオプションは FALSE です。
選択肢の中で唯一正しいのは、Anypoint CLI と Mule Maven プラグインは Mule ランタイムに含まれていないため、デプロイされた Mule アプリケーションでは使用できません、ということです。
Maven は Studio の一部ですが、開発には他の Maven インストールを使用することもできます。
CLI は利便性のみを目的としています。ランタイムにアプリをインストールする多くの方法のうちの 1 つです。
これらは、展開または自動化のプロセス以外の何の一部でもありません。

最新問題**: 34**
API クライアントは、既存の API 実装から 1 つのメソッドを呼び出します。API 実装は後で更新されます。API 実装にどのような変更を加えると、API クライアントの呼び出しロジックも更新する必要が生じますか?
A. APIクライアントによって呼び出されたメソッドのレスポンスのデータ型が変更された場合
B. APIクライアントが使用するリソースに新しいメソッドが追加されたとき
C. APIクライアントによって呼び出されたメソッドに新しい必須フィールドが追加されたとき
D. APIクライアントによって呼び出されたメソッドに子メソッドが追加された場合
Answer: C (メーッセージを残す)
APIクライアントによって呼び出されたメソッドに新しい必須フィールドが追加された場合
****

>> 一般的に、API 契約が破綻した場合、API クライアントのロジックを更新する必要があります。
>> API に新しいメソッドまたは子メソッドが追加されても、API クライアントは既存のメソッドを引き続き使用できるため、動作が中断されることはありません。したがって、これら 2 つのオプションは無効です。
>> 残っているのは、 応答のデータ型が変更された場合」と 新しい必須フィールドが追加された場合」の 2 つです。
>> 応答のデータ型を変更すると、API 契約が破棄されます。ただし、質問は 呼び出し」ロジックに関するものであり、応答処理ロジックに関するものではありません。API クライアントは引き続き API を正常に呼び出して応答を受け取ることができますが、応答の一部のフィールドのデータ型は異なります。
>> 新しい必須フィールドを追加すると、API の呼び出し契約が破棄されます。新しい必須フィールドを追加すると、API 契約により、API クライアント/API コンシューマーと API プロバイダーの間で締結された RAML または API 仕様の合意が破棄されます。そのため、API クライアントの呼び出しロジックも更新する必要があります。

最新問題: **35**
Anypoint Platform 組織は、ID 管理とクライアント管理のために外部 ID プロバイダー (IdP) を使用して設定されています。Anypoint Platform API に対してコマンドを実行するには、Anypoint CLI にどのような資格情報またはトークンを提供する必要がありますか?
A. ID管理のためにIdPが提供する資格情報
B. クライアント管理のためにIdPによって提供される資格情報
C. クライアント管理用に IdP から提供された資格情報を使用して生成された OAuth 2.0 トークン
D. ID管理のためにIdPから提供された資格情報を使用して生成されたOAuth 2.0トークン
Answer: A (メーッセージを残す**)**
ID管理のためにIdPが提供する資格情報
****

最新問題: **36**
バックエンド システムの制限により、システム API は 1 秒あたり最大 500 件のリクエストしか処理できません。バックエンド システムの過負荷を回避するためにシステム API に適用する最適な API ポリシーのタイプは何ですか?
A. レート制限
B. HTTP キャッシュ
C. レート制限 - SLA ベース
D. スパイクコントロール
Answer: D (メーッセージを残す**)**
スパイクコントロール
****

>> まず第一に、HTTP キャッシュ ポリシーは、バックエンド システムの過負荷を回避することとは目的が異なります。したがって、これは OUT です。
>> レート制限とスロットリング/スパイク制御ポリシーは API アクセスを制限するように設計されていますが、目的は異なります。
>> レート制限は、アクセスにハード制限を適用することで API を保護します。
>> スロットリング/スパイク制御は、トラフィックのスパイクを平滑化することで API アクセスを調整します。
そのため、スパイクコントロールが適切な選択肢となります。

最新問題: **37**
展示品を参照してください。
組織は、すべての CloudHub デプロイメントに対して 1 つの特定の CloudHub (AWS) リージョンを使用します。
組織の Mule アプリケーションがそのリージョンの CloudHub にデプロイされている場合、CloudHub ワーカーはどのようにしてアベイラビリティーゾーン (AZ) に割り当てられますか?
A. ワーカーは、そのリージョン内の利用可能な AZ にランダムに分散されます。
B. 特定の環境に属するワーカーは、そのリージョン内の同じAZに割り当てられます。
C. AZはMuleアプリケーションのデプロイメント構成の一部として選択されます
D. Mule アプリケーションに対して AZ がランダムに選択され、Mule アプリケーションのすべての CloudHub ワーカーがその 1 つの AZ に割り当てられます。
Answer: D (メーッセージを残す**)**

最新問題: **38**
複数の CloudHub ワーカーにデプロイされた Mule アプリケーションとして実装された、非同期で実行される長時間実行プロセスのトランザクション状態を追跡するための、Anypoint Platform で最もパフォーマンスの高いすぐに使用できるソリューションは何ですか?
A. java.util.WeakHashMap
B. ファイルベースのストレージ
C. Redis 分散キャッシュ
D. 永続オブジェクトストア
Answer: D (メーッセージを残す**)**

最新問題: **39**
Anypoint Platform REST API、Anypoint CU、Mule Maven プラグインなどのツールを使用して Anypoint Platform とのやり取りを自動化することについて正しいのは何ですか?
A. デフォルトでは、Anypoint CLI と Mule Maven プラグインは Mule ランタイムに含まれていないため、デプロイされた Mule アプリケーションでは使用できません。
B. APIポリシーをAnypoint Platform APIに適用して、特定のLOBのみが特定の機能にアクセスできるようにすることができます。

C. Anypoint Platform API は CloudHub とのやり取りのみを自動化できますが、顧客がホストする Mule ランタイムへの展開には Mule Maven プラグインが必要です。
D. Anypoint Platform API と Anypoint CU へのアクセスは、Anypoint Platform のロールと権限を通じて個別に制御できるため、特定のユーザーは Anypoint CLI にアクセスでき、他のユーザーはプラットフォーム API にアクセスできます。
Answer: A (メーッセージを残す**)**

最新問題**: 40**
プロセス API の実装を変更する必要があります。
この変更が API クライアントに与える影響を最小限に抑える有効なアプローチは何ですか?
A. プロセスAPIの実装に必要な変更を実装し、可能な限りプロセスAPIのRAML定義が変更されないようにします。
B. 現在のプロセスAPIのRAML定義を更新し、更新されたRAML定義へのリンクを送信してAPIクライアント開発者に通知します。
C. APIコンシューマーが新しいプロセスAPIまたはAPIバージョンへの移行の準備ができていることを確認するまで、変更を延期します。
D. Process API の変更を新しい API 実装に実装し、古い API 実装が HTTP ステータス コード 301 - Moved Permanently を返すようにして、API クライアントに新しい API 実装を呼び出す必要があることを通知します。
Answer: (解答を表示する**)**

最新問題**: 41**
CloudHub 専用ロードバランサーを使用する必要がある条件は何ですか?
A. API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
B. 顧客がホストする Mule ランタイムにデプロイされた API 実装にカスタム DNS 名が必要な場合
C. 同じ Mule アプリケーションの別々のデプロイメント間でクロスリージョン負荷分散が必要な場合
D. 複数の CloudHub ワーカー間での API 呼び出しを負荷分散する必要がある場合
Answer: C (メーッセージを残す**)**

最新問題**: 42**
一部の HTTP リクエストに対する応答は、リクエストで使用される HTTP 動詞に応じてキャッシュできます。
HTTP 仕様によれば、どの HTTP 動詞に対してこれを実行しても安全ですか?
A. GET、PUT、オプション
B. PUT、POST、DELETE
C. GET、オプション、HEAD
D. GET、HEAD、POST
Answer: B (メーッセージを残す**)**

最新問題: **43**
ダウンタイムが繰り返し発生することが知られている Order API を呼び出す必要がある API 実装が設計されています。
このため、Order API が利用できない場合は、フォールバック API が呼び出されます。
フォールバック API の呼び出しを設計する際に、どのようなアプローチが最高の回復力を提供しますか?
A. Anypoint Exchangeで適切な既存のフォールバックAPIを検索し、注文APIに加えてこのフォールバックAPIへの呼び出しを実装します。
B. API マネージャーで注文 API の別のエントリを作成し、プライマリ注文 API が利用できない場合にこの API をフォールバック API として呼び出します。
C. Order APIが利用できない場合は、HTTP 307 Temporary Redirectステータスコードを介してクライアントリクエストをフォールバックAPIにリダイレクトします。
D. HTTP リクエスター コンポーネントに、Order API を呼び出すオプションを設定して、Order API から HTTP 4xx または 5xx 応答ステータス コードが返されるたびにフォールバック API を呼び出すようにします。
Answer: (解答を表示する**)**
Anypoint Exchangeで適切な既存のフォールバックAPIを検索し、注文APIに加えてこのフォールバックAPIへの呼び出しを実装します。
****
>> API クライアントが HTTP 3xx 一時リダイレクト ステータス コードを受信し、別の API を呼び出すためにフォールバック ロジックを実装する必要があるという事前承認済みの合意が API クライアントとの間でない限り、これは理想的でも適切なアプローチでもありません。
>> API マネージャーで同じ Order API の別のエントリを作成すると、同じ API 実装の上に別のインスタンスが作成されるだけです。したがって、同じ API のクローンをフォールバック API として使用しても効果はありません。
フォールバック API は、理想的には、プライマリ API とは異なる API 実装である必要があります。
>> 現在、Anypoint HTTP Connector では、応答として特定の HTTP ステータス コードを受信したときにフォールバック API を呼び出すことができるオプションは提供されていません。
指定されたオプションの中で TRUE となる唯一のステートメントは、適切な既存のフォールバック API を Anypoint エクスチェンジで検索し、注文 API に加えてこのフォールバック API への呼び出しを実装することです。

最新問題: **44**
Anypoint Platform でクライアント管理に外部 ID プロバイダーを使用する場合の主な要件は何ですか?
A. Anypoint Platform にサインインするにはシングルサインオンが必要です

B. アプリケーションネットワークには、アイデンティティプロバイダと対話するシステムAPIが含まれている必要があります。
C. Anypoint Platform によって管理される OAuth 2.0 で保護された API を呼び出すには、API クライアントは同じアイデンティティ プロバイダによって発行されたアクセス トークンを送信する必要があります。
D. Anypoint Platform によって管理される API は、SAML 2.0 ポリシーによって保護される必要があります。
Answer: C (メーッセージを残す**)**
説明/参照: https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html

最新問題**: 45**
イノベーションとクロックスピードを向上させるために MuleSoft が組織に推奨する IT 運用モデルの主な変更点は何ですか?
A. アセットの生産と同様に消費も促進します。これにより、開発者は他のプロジェクトからアセットを発見して再利用できるようになり、標準化が促進されます。
B. マスターデータ管理 (MDM) システムを使用して資産を公開します。これによりプロジェクトが標準化され、開発者は他のプロジェクトから資産をすばやく発見して再利用できるようになります。
C. 再利用可能な API に SOA を実装して、消費よりも生産に重点を置きます。これにより、XML および WSDL 形式が標準化され、意思決定が迅速化されます。
D. 毎日多くの小さな決定を下す、無駄のない機敏な組織を構築します。これにより意思決定が迅速化され、各事業部門がプロジェクトの所有権を取得できるようになります。
Answer: (解答を表示する**)**
アセットの生産と同様に消費も促進します。これにより、開発者は他のプロジェクトからアセットを発見して再利用できるようになり、標準化が促進されます。
****
>> MuleSoft が推奨し普及させた新しい IT 運用モデルの主なモットーは、API 主導の接続性と呼ばれる API 戦略を通じて、提供方法を実稼働モデルから実稼働 + 消費モデルに変更することです。
>> 構築された資産は、LOB や組織全体で再利用できるように、検出可能でセルフサービス可能である必要があります。
>> MuleSoft の IT 運用モデルでは、SDLC モデル (Agile/Lean など) や MDM についてはまったく触れられていません。したがって、これらを提案するオプションは無効です。
参考文献:
https://blogs.mulesoft.com/biz/connectivity/what-is-a-center-for-enablement-c4e/
https://www.mulesoft.com/resources/api/secret-to-managing-it-projects

最新問題**: 46**

新しいアップストリーム API は、平均 500 ミリ秒、最大 800 ミリ秒 (99 パーセンタイル) の応答時間の SLA を提供するように設計されています。対応する API 実装では、非常に類似した複雑さの 3 つのダウンストリーム API を順番に呼び出す必要があります。
これらのダウンストリーム API の最初のものは、応答時間について次の SLA を提供します: 中央値: 100 ミリ秒、80 パーセンタイル: 500 ミリ秒、95 パーセンタイル: 1000 ミリ秒。
可能であれば、新しいアップストリーム API の望ましい SLA を満たすために、最初のダウンストリーム API の呼び出しに対してアップストリーム API でタイムアウトを設定するにはどうすればよいでしょうか。
A. タイムアウトを 50 ミリ秒に設定します。これにより、その API の呼び出しがさらにタイムアウトしますが、再試行の余地がさらに増えます。
B. タイムアウトを 100 ミリ秒に設定します。これにより、他の 2 つのダウンストリーム API が完了するまでに 400 ミリ秒が残ります。
C. 上流 API の希望する SLA を満たすためにタイムアウトは不可能です。最初の下流 API と異なる SLA を交渉するか、代替 API を呼び出す必要があります。
D. タイムアウトを設定しないでください。このAPIの呼び出しは必須なので、応答するまで待つ必要があります。
Answer: (解答を表示する**)**
タイムアウトを100msに設定すると、他の2つのダウンストリームAPIが完了するまでに400msの猶予が残ります。
****
与えられたシナリオから得られる重要な詳細:
>> アップストリーム API の設計上の SLA は 500 ミリ秒 (中央値) です。最大 SLA 応答時間は無視します。
>> この API は 3 つのダウンストリーム API を順番に呼び出しますが、これらはすべて同様の複雑さです。
>> 最初のダウンストリームAPIは、100msの中央SLAを提供しています。80パーセンタイル：500ms、95パーセンタイル：
1000ミリ秒。
上記の詳細に基づいて:
>> 50 ミリ秒のタイムアウトを設定することを提案するオプションは除外できます。なぜなら、提供される SLA 自体の中央値が 100 ミリ秒の場合、ほとんどの呼び出しがタイムアウトになり、再試行に時間が浪費され、最終的にすべての再試行で使い果たされるからです。再試行がいくつか成功したとしても、残りの時間では、2 番目と 3 番目のダウンストリーム API が時間内に応答するのに十分な余裕がありません。
>> このAPIの呼び出しは必須なので、応答するまで待たなければならないため、タイムアウトを設定しないことを提案するオプションはばかげています。タイムアウトを設定しないことは、適切な実装パターンに反することになります。さらに、最初のAPIが提供された中央SLA 100ms以内に応答しない場合は、おそらく500ms（80パーセンタイル）または1000ms（95パーセンタイル）で

応答するでしょう。どちらの場合も、最初のダウンストリームAPIから正常な応答を得ても何の役にも立ちません。なぜなら、この時点ですでにアップストリームAPI SLAは
500 ミリ秒を超えました。2 番目と 3 番目のダウンストリーム API を呼び出す時間が残っていません。
>> アップストリーム API の望ましい SLA を満たすためにタイムアウトが不可能であるというのは真実ではありません。
最初のダウンストリーム API は平均 SLA 100 ミリ秒を提供しているため、ほとんどの場合、その時間内に応答が返されます。したがって、残りの 2 つのダウンストリーム API 呼び出しに 400 ミリ秒の余裕が残るため、ほとんどの呼び出しではタイムアウトを 100 ミリ秒に設定するのが理想的です。

最新問題**: 47**
CloudHub 専用ロードバランサーを使用する必要がある条件は何ですか?
A. 同じ Mule アプリケーションの別々のデプロイメント間でクロスリージョン負荷分散が必要な場合
B. 顧客がホストする Mule ランタイムにデプロイされた API 実装にカスタム DNS 名が必要な場合
C. 複数の CloudHub ワーカー間での API 呼び出しを負荷分散する必要がある場合
D. API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
Answer: (解答を表示する**)**
正解: API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
****
事実/メモリのヒント: CloudHub 専用ロードバランサには多くの利点がありますが、検討する際に心に留めておくべき重要な点が 2 つあります。
>> CloudHub にデプロイされたアプリにカスタム DNS 名を持つ URL エンドポイントを設定する
>> HTTPS と双方向 (相互) 認証の両方に対してカスタム証明書を構成します。
これに関して提供されているオプションについて
>> 私たち
DLB を使用して、同じ Mule アプリケーションの個別のデプロイメント間でクロスリージョン負荷分散を実行することはできません。

>> 複数の DLB URL が同じ Mule アプリを指すようにマッピング ルールを設定できます。ただし、その逆 (複数の Mule アプリが同じ DLB URL を持つ) は不可能です。
>> DLB は Cloudhub にデプロイされた Mule アプリのカスタム DNS 名の設定に役立ちますが、顧客がホストする Mule ランタイムにデプロイされたアプリには当てはまりません。
>> DLB を使用して複数の CloudHub ワーカー間で API 呼び出しの負荷を分散できることは事実ですが、必須ではありません。SLB (共有ロード バランサ) を使用しても同じこと (負荷分散) を実現できます。これを実現するために必ずしも DLB が必要というわけではありません。
したがって、シナリオに適合し、DLB を使用する必要がある唯一の適切なオプションは、API 実装と API クライアント間で TLS 相互認証が必要な場合です。

最新問題: **48**
展示品を参照してください。
新しいプロモーション プロセス API の RAML 定義が提案され、Anypoint Exchange に公開されました。
プロモーション API の重要な消費者となるマーケティング部門には、満たさなければならない重要な要件と期待があります。
Anypoint Platform の機能を使用して、マーケティング部門をこの初期の API 設計フェーズに参加させる最も効果的な方法は何ですか?
A) マーケティング部門に、自動生成されたAPIコンソールを使用してAPIのモック実装を操作するように依頼します。
B) マーケティング部門のDBAと設計ワークショップを開催し、マーケティングITシステムのデータベーススキーマをRAMLに変換する。
C) Anypoint Studioを使用してAPIをMuleアプリケーションとして実装し、そのAPI実装をCloudHubにデプロイして、マーケティング部門に操作を依頼します。
D) APIデザイナーから統合テストスイートをエクスポートし、マーケティング部門にそのスイートのテストを実行させて合格することを確認する。
A. オプションB
B. オプションD
C. オプションC
D. オプションA
Answer: B (メーッセージを残す**)**

最新問題: **49**
展示品を参照してください。
顧客がホストする Mule ランタイムを MuleSoft がホストする Anypoint Platform コントロールプレーン (ハイブリッド展開) と併用する場合、正しいのは何ですか?
A. MuleSoft がホストする共有ロードバランサは、Mule ランタイムへの API 呼び出しの負荷分散に使用できます。

B. Anypoint Runtime Manager は、Mule アプリケーションをデプロイするために Mule ランタイムへのネットワーク接続を開始します。
C. Anypoint Runtime Manager は、ノード障害が発生した場合に新しい Mule ランタイムインスタンスを作成することで、コントロールプレーンの HA を自動的に確保します。
D. API 実装は、コントロール プレーンと通信できない場合でも、顧客がホストする Mule ランタイムで正常に実行できます。
Answer: (解答を表示する**)**

最新問題**: 50**
API 実装が更新されます。API の RAML 定義もいつ更新する必要がありますか?
A. API実装がMuleランタイムの古いバージョンから新しいバージョンに移行された場合
B. API実装が最適化され、平均応答時間が改善された場合
C. API実装がリクエストまたはレスポンスメッセージの構造を変更する場合
D. API実装がオンプレミスに展開された従来のバックエンドシステムとのやり取りから、最新のクラウドベース $aaS) システムへの変更の場合
Answer: C (メーッセージを残す**)**

最新問題**: 51**
CloudHub 専用ロードバランサーを使用する必要がある条件は何ですか?
A. 同じ Mule アプリケーションの別々のデプロイメント間でクロスリージョン負荷分散が必要な場合
B. 顧客がホストする Mule ランタイムにデプロイされた API 実装にカスタム DNS 名が必要な場合
C. 複数の CloudHub ワーカー間での API 呼び出しを負荷分散する必要がある場合
D. API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
Answer: D (メーッセージを残す**)**
API実装とAPIクライアント間でサーバー側負荷分散TLS相互認証が必要な場合
****
事実/メモリのヒント: CloudHub 専用ロードバランサには多くの利点がありますが、検討する際に心に留めておくべき重要な点が 2 つあります。
>> CloudHub にデプロイされたアプリにカスタム DNS 名を持つ URL エンドポイントを設定する
>> HTTPS と双方向 (相互) 認証の両方に対してカスタム証明書を構成します。
この質問に対して提供されているオプションは次のとおりです。
>> DLB を使用して、同じ Mule アプリケーションの個別のデプロイメント間でリージョン間の負荷分散を実行することはできません。
>> 複数の DLB URL が同じ Mule アプリを指すようにマッピング ルールを設定できます。ただし、その逆 (複数の Mule アプリが同じ DLB URL を持つ) は不可能です。
>> DLB は Cloudhub にデプロイされた Mule アプリのカスタム DNS 名の設定に役立ちますが、顧客がホストする Mule ランタイムにデプロイされたアプリには当てはまりません。

>> DLB を使用して複数の CloudHub ワーカー間で API 呼び出しの負荷を分散できることは事実ですが、必須ではありません。SLB (共有ロード バランサ) を使用しても同じこと (負荷分散) を実現できます。これを実現するために必ずしも DLB が必要というわけではありません。
したがって、シナリオに適合し、DLB を使用する必要がある唯一の適切なオプションは、API 実装と API クライアント間で TLS 相互認証が必要な場合です。

最新問題: **52**
展示品を参照してください。
開発者は、クライアント ID 適用ポリシーによって管理されるステージング環境にデプロイされた API を呼び出すクライアント アプリケーションを構築しています。
API を正常に呼び出すには何が必要ですか?
A. STAGING 環境で API を所有する Anypoint Platform アカウントのクライアント ID とシークレット
B. Anypoint Platform アカウントのステージング環境のクライアント ID とシークレット
C. STAGING環境のAPIインスタンスのAnypoint Exchangeから取得したクライアントIDとシークレット
D. Anypoint Platform から取得した有効な OAuth トークンとそれに関連付けられたクライアント ID およびシークレット
Answer: C (メーッセージを残す**)**
正解: STAGING 環境の API インスタンスの Anypoint Exchange から取得したクライアント ID とシークレット
****
>> APIにアクセスするためにAnypoint Platformアカウントまたは個々の環境のクライアントIDとシークレットを使用することはできません。
>> 問題の API に適用されるポリシーの種類は 「クライアント ID 強制ポリシー」であるため、OAuth トークン ベースのアクセスは機能しません。
API にアクセスする正しい方法は、作業する特定の環境の API インスタンスに対して Anypoint Exchange から取得したクライアント ID とシークレットを使用することです。
参考文献:
API マネージャーでの API インスタンス契約の管理
https://docs.mulesoft.com/api-manager/1.x/request-access-to-api-task
https://docs.mulesoft.com/exchange/to-request-access
https://docs.mulesoft.com/api-manager/2.x/policy-mule3-client-id-based-policies

最新問題: **53**
組織は、最新の API (MuleSoft の定義による) を使用して再利用可能な IT 資産の消費を重視する IT 運用モデルに移行するという戦略的決定を下します。
この新しい IT 運用モデルに関連して、各最新 API を最もよく表すものは何ですか?
A. 最新のAPIにはそれぞれ独自のソフトウェア開発ライフサイクルがあり、ドキュメント作成や自動化の必要性が軽減されます。

B. 各モデム API は製品のように扱われ、特定の対象ユーザー (たとえば、モバイル アプリ開発者) 向けに設計される必要があります。
C. 各モダンAPIは簡単に使用できる必要があるため、SAMLやJWTなどの複雑な認証メカニズムは避けるべきです。
D. 最新のAPIはRESTとHTTPベースでなければならない
Answer: B (メーッセージを残す**)**
正解:
1. 各最新APIは製品のように扱われ、特定の対象ユーザー（モバイルアプリ開発者など）向けに設計される必要がある
****

最新問題**: 54**
どの Mule アプリケーション展開シナリオで Anypoint Platform Private Cloud Edition または Anypoint Platform for Pivotal Cloud Foundry を使用する必要がありますか?
A. 複数のデータセンターにわたってすべてのアプリケーションの高可用性を実現する必要がある場合
B. すべてのAPIがプライベートであり、パブリッククラウドに公開されないことが求められる場合
C. 規制要件により、メタデータを含むすべてのデータ項目のオンプレミス処理が義務付けられている場合
D. アプリケーションネットワーク内のすべてのバックエンドシステムが組織のイントラネットに展開されている場合
Answer: C (メーッセージを残す**)**
規制要件により、メタデータを含むすべてのデータ項目のオンプレミス処理が義務付けられている場合。
****
以下の場合、Anypoint Platform PCE または PCF を使用する必要はありません。したがって、これらのオプションは無効です。
>> CloudHub を使用すると、複数のデータセンターにわたってすべてのアプリケーションの高可用性を実現できます。
>> Anypoint VPN と CloudHub からのトンネリングを使用して、組織のイントラネットに展開されているアプリケーション ネットワーク内のすべてのバックエンド システムに接続できます。
>> Anypoint VPC とファイアウォール ルールを使用して、すべての API をプライベートにし、パブリック クラウドに公開しないようにすることができます。
指定されたオプションの中で、Anypoint Platform PCE/PCF の使用が必要な唯一の有効な理由は、規制要件により、メタデータを含むすべてのデータ項目のオンプレミス処理が義務付けられている場合です。

最新問題**: 55**

複数の CloudHub ワーカーにデプロイされた Mule アプリケーションとして実装された、非同期で実行される長時間実行プロセスのトランザクション状態を追跡するための、Anypoint Platform で最もパフォーマンスの高いすぐに使用できるソリューションは何ですか?
A. Redis 分散キャッシュ
B. java.util.WeakHashMap
C. 永続オブジェクトストア
D. ファイルベースのストレージ
Answer: C (メーッセージを残す**)**
正解: 永続オブジェクトストア
****
>> Redis分散キャッシュはパフォーマンスに優れていますが、Anypoint Platformですぐに使用できるソリューションではありません
>> ファイルストレージは、Anypoint Platform ではパフォーマンスが高くなく、すぐに使用できるソリューションでもありません。
>> java.util.WeakHashMap は、Java コードを使用して最初から完全にカスタム化されたキャッシュ実装を必要とし、実行されている JVM に制限されます。つまり、複数のワーカーで実行されている場合、キャッシュの状態はワーカーに対応していません。このタイプのキャッシュはワーカーに対してローカルです。したがって、これはすぐに使用できるものではなく、クラウドハブ上の複数のワーカー間でワーカー対応でもありません。https://www.baeldung.com/java-weakhashmap
>> 永続オブジェクト ストアは、Anypoint Platform によって提供されるすぐに使用できるソリューションであり、CloudHub で実行されている複数のワーカー間でパフォーマンスが高く、ワーカーを認識します。https://docs.mulesoft.com/object-store/ したがって、永続オブジェクト ストアが正しい答えです。

最新問題**: 56**
API 実装における自動検出の使用を最もよく説明するものは何ですか?
A. API ManagerがAPI実装を認識し、ポリシーを適用できるようにします。
B. Anypoint StudioがAnypoint Platformで設定されたAPI定義を検出できるようにします。
C. Anypoint Exchangeが資産を発見し、再利用できるようにします。
D. Anypoint AnalyticsがAPIの使用状況を把握できるようになります。
Answer: A (メーッセージを残す**)**
これにより、API Manager は API 実装を認識し、ポリシーを適用できるようになります。
****
>> API 自動検出は、デプロイされたアプリケーションをプラットフォーム上で作成された API とペアリングすることで、API マネージャーから API を管理するメカニズムです。
>> API 管理には、追跡、ポリシーの適用（適用する場合）、API 分析のレポートが含まれます。
>> 自動検出プロセスにとって重要なのは、API 名とバージョンを指定して API を識別することです。
参考文献:

https://docs.mulesoft.com/api-manager/2.x/api-auto-discovery-new-concept
https://docs.mulesoft.com/api-manager/1.x/api-auto-discovery
https://docs.mulesoft.com/api-manager/2.x/api-auto-discovery-new-concept

最新問題: **57**
ある組織には、HTTP POST 経由で JSON データを受け入れる API がいくつかあります。これらの API はすべて公開されており、いくつかのモバイル アプリケーションや Web アプリケーションに関連付けられています。
組織はこれらの API に対して認証やコンプライアンス ポリシーを使用することを望んでいませんが、同時に、悪意のある人物が API 実装を実行しているアプリケーションやサーバーを何らかの形で侵害する可能性のあるペイロードを送信する可能性があることを懸念しています。
この脅威への露出に対処できる、すぐに使用できる Anypoint Platform ポリシーは何ですか?
A. すべてのAPI呼び出しにHTTPS相互認証を使用して悪意のある行為者をシャットアウトします
B. すべてのAPIにIPブラックリストポリシーを適用します。ブラックリストにはすべての悪質な行為者が含まれます。
C. 悪意のあるデータが使用される前にそれを検出するヘッダー挿入および削除ポリシーを適用する
D. 潜在的な脅威ベクトルを検出するために、すべてのAPIにJSON脅威保護ポリシーを適用します。
Answer: (解答を表示する**)**
正解: 潜在的な脅威ベクトルを検出するために、すべての API に JSON 脅威保護ポリシーを適用する
****
>> 通常、API が特定の消費者 (既知の消費者/顧客) 向けに設計および開発されている場合は、トラフィックがその消費者/顧客からのみ発生するように、同じものを IP ホワイトリストに登録します。
>> ただし、このシナリオでは、API が一般に公開されており、非常に多くのモバイル アプリケーションや Web アプリケーションで使用されているため、すべての悪意のある行為者を特定してブラックリストに登録することは不可能です。
>> したがって、JSON 脅威保護ポリシーは、そのような悪意のある行為者による不正な JSON ペイロードを防ぐための最善の機会です。

最新問題: **58**
API 主導の接続のどのレイヤーにビジネス ロジック オーケストレーションが存在します?
A. システム層
B. エクスペリエンスレイヤー
C. プロセス層
Answer: (解答を表示する**)**
プロセスレイヤー
****

>> エクスペリエンス レイヤーは、エンド ユーザー エクスペリエンスの強化を目的としています。このレイヤーは、さまざまな API クライアント/消費者のニーズを満たすためのものです。
>> システム層は、本質的にモジュール化されたAPI専用であり、バックエンドシステムのさまざまな個別の機能を実装/公開します。
>> プロセス レイヤーは、1 つまたは複数のシステム レイヤー モジュラー API を呼び出して、単純または複雑なビジネス オーケストレーション ロジックが記述される場所です。したがって、プロセス レイヤーが正しい答えです。

最新問題: **59**
ある組織は、Azure 環境で MuleSoft がホストするランタイム プレーン機能 (HTTP 負荷分散、ゼロ ダウンタイム、水平および垂直スケーリングなど) を必要としています。これらの機能を実現するための組織の労力を最小限に抑えるランタイム プレーンはどれですか。
A. Anypoint ランタイム ファブリック
B. Pivotal Cloud Foundry 向け Anypoint プラットフォーム
C. クラウドハブ
D. 顧客ホスト型と MuleSoft ホスト型の Mule ランタイムのハイブリッドな組み合わせ
Answer: (解答を表示する**)**
正解: Anypoint Runtime Fabric
****
>> 顧客が既に Azure 環境を持っている場合、一部の Mule ランタイムを Azure でホストし、一部を MuleSoft でホストするハイブリッド モデルを採用するのは、まったく理想的なアプローチではありません。これは不要であり、役に立ちません。
>> CloudHub は Mulesoft がホストするランタイム プレーンであり、AWS 上にあります。CloudHub を顧客の Azure 環境にポイントするようにカスタマイズすることはできません。
>> Pivotal Cloud Foundry向けAnypoint Platformは、Pivotal Cloud Foundryが提供するインフラストラクチャ専用です。
>> Anypoint Runtime Fabric は、Mule アプリケーションと API ゲートウェイの展開とオーケストレーションを自動化するコンテナ サービスであるため、正しい答えです。Runtime Fabric は、AWS、Azure、仮想マシン (VM)、ベアメタル サーバー上の顧客管理インフラストラクチャ内で実行されます。
-Anypoint Runtime Fabric の機能には次のようなものがあります。
- アプリケーションごとに個別の Mule ランタイムを実行することで、アプリケーション間の分離を実現します。
- 同じリソース セットで複数のバージョンの Mule ランタイムを実行する機能。
- 複数のレプリカにわたってアプリケーションをスケーリングします。
- 自動化されたアプリケーションフェイルオーバー。
-Anypoint Runtime Manager によるアプリケーション管理。

最新問題: **60**

質問10: スキップ
API 実装は、要求元の API クライアントに 3 つの X-RateLimit-* HTTP 応答ヘッダーを返します。これらの応答ヘッダーは API クライアントにどのような種類の情報を示しますか?
A. スロットリングの結果生じるエラーコード
B. 次のリクエストで送信される相関ID
C. HTTPレスポンスサイズ
D. API実装によって許可された残りの容量
Answer: (解答を表示する**)**
API 実装によって許可される残りの容量。
****
>>
参考:https://docs.mulesoft.com/api-manager/2.x/rate-limiting-and-throttling-sla-based-policies#response-he

最新問題**: 61**
Anypoint Platform が提供する API 呼び出しメトリクスは何を提供しますか?
A. ビジネス ユーザーと直接共有できる API からの ROI メトリック
B. 再利用レベルに基づくアプリケーションネットワークの有効性の測定
C. 過去の API 呼び出しに関するデータ。さまざまな API の異常や使用パターンの特定に役立ちます。
D. 特定の脅威しきい値を超える可能性のある将来のポリシー違反を積極的に特定する
Answer: (解答を表示する**)**
過去の API 呼び出しに関するデータにより、さまざまな API の異常や使用パターンを特定できます。
****
Anypoint Platform が提供する API 呼び出しメトリクス:
>> 投資収益率 (ROI) に関連する情報は提供されません。したがって、それを提案するオプションは無効です。
>> API がどのように再利用されるか、API が効果的に使用されているかどうかなどに関する情報は提供されません。
>> 将来のポリシー違反を積極的に特定するのに役立つような予測情報は提供されません。
したがって、このようなメトリックから取得できるデータ/情報は、過去の API 呼び出しに関するもので、さまざまな API にわたる異常や使用パターンを識別するのに役立ちます。

最新問題: **62**
パブリック Anypoint Exchange ポータルを通じて API を共有する前に確認すべきことは何ですか?
A. 公開アクセスが必要なAPIインスタンスの可視性レベルは、パブリック可視性に設定する必要があります。
B. APIへのアクセスが必要なユーザーは、Anypoint Platformの適切なロールに追加する必要があります。
C. APIは少なくとも初期実装が展開され、ユーザーが操作できるようにアクセス可能で機能する必要があります。
D. データが侵害されないように、サポートされている認証/承認メカニズムのいずれかを使用して API を保護する必要があります。
Answer: A (メーッセージを残す)
説明
https://docs.mulesoft.com/exchange/to-share-api-asset-to-portal

最新問題: **63**
大量の統合ロジックを含み、製品 API の呼び出しを伴う注文 API を設計する必要があります。
製品 API は組織全体で頻繁に使用され、CTO のオフィスにある専用の開発チームによって開発されているため、注文 API と製品 API 間の力関係は 顧客/サプライヤー」の関係になります。
Order API 内で Product API の API データ モデルを処理するには、どのような戦略を使用する必要がありますか?
A. 製品 API の開発チームに、注文 API の API データ モデルを採用するよう説得し、注文 API の統合ロジックが 1 つの一貫した内部データ モデルで動作できるようにします。
B. Order API の統合ロジックを実装するときに、Product API の API データ型を直接操作して、Order API が Product API と同じ (変更されていない) データ型を使用するようにします。
C. Order API に、Product API データ モデルを Order API の内部データ型に変換する破損防止レイヤーを実装します。
D. 組織全体のデータ モデリング イニシアチブを開始し、製品 API と注文 API の両方で使用されるエンタープライズ データ モデルを作成します。
Answer: C (メーッセージを残す)
製品 API の開発チームに、注文 API の API データ モデルを採用するよう説得し、注文 API の統合ロジックが 1 つの一貫した内部データ モデルで動作できるようにします。
****
与えられたシナリオから注目すべき重要な詳細:

>> 注文 API と製品 API 間の力関係は顧客/サプライヤーです。したがって、以下の 力関係」のルールに従って、呼び出し元 (この場合は注文 API) は呼び出された側 (製品 API チーム) に機能を要求し、製品 API チームはそれらの要求に対応する必要があります。

最新問題: **64**
以下のオプションから正しいオーナーとレイヤーの組み合わせを選択してください
A. 1. アプリ開発者はエクスペリエンスレイヤーAPIを所有し、それに重点を置いています
2. 中央IT部門はプロセスレイヤーAPIを所有し、それに重点を置いています
3. LOB ITはシステム層APIを所有し、それに重点を置いています
B. 1. 中央IT部門はエクスペリエンスレイヤーAPIを所有し、それに重点を置いています
2. LOB ITはプロセスレイヤーAPIを所有し、それに重点を置いています
3. アプリ開発者はシステムレイヤーAPIを所有し、それに重点を置いています
C. 1. アプリ開発者はエクスペリエンスレイヤーAPIを所有し、それに重点を置いています
2. LOB ITはプロセスレイヤーAPIを所有し、それに重点を置いています
3. 中央IT部門はシステム層APIを所有し、それに重点を置いている
Answer: C (メーッセージを残す**)**
1. アプリ開発者はエクスペリエンスレイヤーAPIを所有し、それに重点を置いています
2. LOB ITはプロセスレイヤーAPIを所有し、それに重点を置いています
3. 中央IT部門はシステム層APIを所有し、それに重点を置いている
参考文献:
https://blogs.mulesoft.com/biz/api/ experience-api-ownership/
https://blogs.mulesoft.com/biz/api/process-api-ownership/
https://blogs.mulesoft.com/biz/api/system-api-ownership/

最新問題: **65**
アプリケーション ネットワークは再構成可能であり、 曲がっても壊れない」ため、変更に対応できるように構築されています。
A. 真
B. 偽
Answer: (解答を表示する**)**
****
>> アプリケーション ネットワークは使い捨てのアーキテクチャです。
>> つまり、アーキテクチャ全体とそのコンポーネントに影響を与えることなく変更できるということです。
>> 要件や設計変更に応じて曲がるが、壊れない

最新問題: **66**
ある組織では、今日の引用をキャッシュする Quote of the Day API を実装しています。

どのようなシナリオで、オブジェクト ストア コネクタを介して GoudHub オブジェクト ストアを使用して、キャッシュの状態を永続化できますか?
A. API実装のCloudHubデプロイメントが3つあり、キャッシュ状態を共有する必要がある3つの別々のCloudHubリージョンにある場合
B. 2つのAnypoint Platformビジネスグループによる API実装の2つのCloudHubデプロイメントが同じCloudHubリージョンにあり、キャッシュ状態を共有する必要がある場合
C. CloudHubへのAPI実装のデプロイメントが1つあり、顧客がホストするMuleランタイムへのottVデプロイメントが1つあり、キャッシュ状態を共有する必要がある場合
D. API実装のCloudHubデプロイメントが1つあり、キャッシュ状態を共有する必要がある3つのCloudHubワーカーがある場合
Answer: D (メーッセージを残す**)**
正解: キャッシュ状態を共有する必要のある 3 つの CloudHub ワーカーに API 実装の CloudHub デプロイメントが 1 つある場合。
****
シナリオの主な詳細:
>> オブジェクトストアコネクタ経由でCloudHubオブジェクトストアを使用する
上記の詳細を考慮すると:
>> CloudHub オブジェクト ストアは、CloudHub Mule アプリケーションと 1 対 1 の関係にあります。
>> オブジェクト ストア コネクタを使用して、異なるリージョンやビジネス グループで実行されている複数の Mule アプリケーション、または顧客がホストする Mule ランタイム間でアプリケーションの CloudHub オブジェクト ストアを共有することはできません。
>> 本当に必要で、非常に必要な場合、Anypoint Platform は、Object Store REST API を使用して別のアプリケーションの CloudHub Object Store へのアクセスを許可する方法をサポートします。ただし、Object Store コネクタは使用しません。
したがって、オブジェクト ストア コネクタを介して CloudHub オブジェクト ストアを使用してキャッシュの状態を永続化できる唯一のシナリオは、キャッシュの状態を共有する必要のある複数の CloudHub ワーカーに対して API 実装の 1 つの CloudHub デプロイメントがある場合です。

最新問題**: 67**
展示品を参照してください。
エンドツーエンドのビジネス プロセスをエクスペリエンス、プロセス、システム API のコラボレーションに分解する最適な方法は何ですか?
A) エンドユーザー アプリケーションのカスタマイズをエクスペリエンス API レベルではなくプロセス API レベルで処理する
B) システムAPIが、特定されたプロセスAPIまたはエクスペリエンスAPIで現在必要とされていないデータを返すことを許可する
C) 3 つのレイヤー (エクスペリエンス、プロセス、システム API) ごとに 1 つの API を作成し、常に階層化アプローチを使用します。

D) プロセス API を使用して複数のシステム API への呼び出しを調整しますが、他のプロセス API への呼び出しは調整しません。
A. オプションC
B. オプションD
C. オプションB
D. オプションA
Answer: (解答を表示する**)**

最新問題**: 68**
Anypoint Platform REST API、Anypoint CU、Mule Maven プラグインなどのツールを使用して Anypoint Platform とのやり取りを自動化することについて正しいのは何ですか?
A. デフォルトでは、Anypoint CLI と Mule Maven プラグインは Mule ランタイムに含まれていないため、デプロイされた Mule アプリケーションでは使用できません。
B. APIポリシーをAnypoint Platform APIに適用して、特定のLOBのみが特定の機能にアクセスできるようにすることができます。
C. Anypoint Platform API は CloudHub とのやり取りのみを自動化できますが、顧客がホストする Mule ランタイムへの展開には Mule Maven プラグインが必要です。
D. Anypoint Platform API と Anypoint CU へのアクセスは、Anypoint Platform のロールと権限を通じて個別に制御できるため、特定のユーザーは Anypoint CLI にアクセスでき、他のユーザーはプラットフォーム API にアクセスできます。
Answer: C (メーッセージを残す**)**

最新問題**: 69**
組織は、最新の API (MuleSoft の定義による) を使用して再利用可能な IT 資産の消費を重視する IT 運用モデルに移行するという戦略的決定を下します。
この新しい IT 運用モデルに関連して、各最新 API を最もよく表すものは何ですか?
A. 最新のAPIにはそれぞれ独自のソフトウェア開発ライフサイクルがあり、ドキュメント作成や自動化の必要性が軽減されます。
B. 各モデム API は製品のように扱われ、特定の対象ユーザー (たとえば、モバイル アプリ開発者) 向けに設計される必要があります。
C. 各モダンAPIは簡単に使用できる必要があるため、SAMLやJWTなどの複雑な認証メカニズムは避けるべきです。
D. 最新のAPIはRESTとHTTPベースでなければならない
Answer: (解答を表示する**)**
正解:
1. 各最新APIは製品のように扱われ、特定の対象ユーザー（モバイルアプリ開発者など）向けに設計される必要がある
****

フォームの下部
フォームの先頭

最新問題: **70**
プロセス API に適用される可能性が最も低い API ポリシーは何ですか?
A. カスタム回路ブレーカー
B. レート制限
C. JSON 脅威保護
D. クライアントIDの強制
Answer: C (メーッセージを残す**)**

最新問題: **71**
ダウンタイムが繰り返し発生することが知られている Order API を呼び出す必要がある API 実装が設計されています。
このため、Order API が利用できない場合は、フォールバック API が呼び出されます。
フォールバック API の呼び出しを設計する際に、どのようなアプローチが最高の回復力を提供しますか?
A. Anypoint Exchangeで適切な既存のフォールバックAPIを検索し、注文APIに加えてこのフォールバックAPIへの呼び出しを実装します。
B. API マネージャーで注文 API の別のエントリを作成し、プライマリ注文 API が利用できない場合にこの API をフォールバック API として呼び出します。
C. Order APIが利用できない場合は、HTTP 307 Temporary Redirectステータスコードを介してクライアントリクエストをフォールバックAPIにリダイレクトします。
D. HTTP リクエスター コンポーネントに、Order API を呼び出すオプションを設定して、Order API から HTTP 4xx または 5xx 応答ステータス コードが返されるたびにフォールバック API を呼び出すようにします。
Answer: A (メーッセージを残す**)**

最新問題: **72**
API 実装の準備が整い、API が API Manager に登録されたら、Anypoint Exchange 上の API へのアクセスをリクエストするのは誰ですか?
A. なし
B. 両方
C. API クライアント
D. API コンシューマー
Answer: (解答を表示する**)**
正解: API コンシューマー
****

>> APIクライアントは、APIコンシューマのクライアント資格情報を使用するコードまたはプログラムですが、アクセスを取得するためにAnypoint Exchangeと直接やり取りすることはありません。

>> API コンシューマーは登録して API へのアクセスをリクエストする必要があり、API クライアントはそれらのクライアント資格情報を使用して API にアクセスする必要があります。つまり、API コンシューマーは Anypoint Exchange から API へのアクセスをリクエストする必要がある人です。

最新問題: **73**
ある組織では、今日の引用をキャッシュする Quote of the Day API を実装しています。
A. API実装のCloudHubデプロイメントが3つあり、キャッシュ状態を共有する必要がある3つの別々のCloudHubリージョンにある場合
B. 2つのAnypoint Platformビジネスグループによる API実装の2つのCloudHubデプロイメントが同じCloudHubリージョンにあり、キャッシュ状態を共有する必要がある場合
C. CloudHubへのAPI実装のデプロイメントが1つあり、顧客がホストするMuleランタイムへのottVデプロイメントが1つあり、キャッシュ状態を共有する必要がある場合
D. オブジェクト ストア コネクタを介して GoudHub オブジェクト ストアを使用してキャッシュの状態を永続化できるシナリオは何ですか?
E. API実装のCloudHubデプロイメントが1つあり、キャッシュ状態を共有する必要がある3つのCloudHubワーカーがある場合
Answer: (解答を表示する**)**

最新問題: **74**
ある組織では、InfoSec チームが Anypoint Platform 関連のデータ トラフィックを調査しています。
Anypoint Platform で監視およびアラートに使用できるデータのほとんどはどこから発生するのでしょうか?
A. デプロイメント モデルに応じて、Mule ランタイムまたは API 実装から
B. 共有ロードバランサ、VPC、Muleランタイムなど、Anypoint Platformのさまざまなコンポーネントから
C. データの種類に応じて、Mule ランタイムまたは API マネージャーから
D. デプロイメントモデルに関係なく、Mule ランタイムから
Answer: D (メーッセージを残す**)**
正解: デプロイメントモデルに関係なく、Mule ランタイムから
****
>> モニタリングとアラートのメトリックは、デプロイメント モデルに関係なく、常に Mule ランタイムから生成されます。
>> 一部のメトリック (Runtime Manager) は Mule Runtime から生成され、一部 (API 呼び出し/API 分析) は API Manager から生成されているように見えるかもしれません。ただし、これは現実的には正しくありません。その理由は、API Manager は API インスタンスの管理ツールに過ぎませんが、API に適用されるすべてのポリシーは最終的に Mule Runtime (埋め込みまたは API プロキシ) でのみ実行されるためです。
>> 同様に、すべての API 実装も Mule ランタイム上で実行されます。

したがって、デプロイメント モデルが MuleSoft ホスト型、顧客ホスト型、またはハイブリッド型であるかどうかに関係なく、監視とアラートに必要な 1 日の大半は Mule Runtimes からのみ生成されます。

最新問題: **75**
システム API の API データ モデルは、バックエンド システムのデータ モデルを最小限に改良して、対応するバックエンド システムによって公開されるデータ モデルを適切に模倣できる場合とはどのような場合でしょうか。
A. システムAPIを対応するデータモデルを持つ境界付きコンテキストに割り当てることができる場合
B. 対応するバックエンドシステムが近い将来に置き換えられることが予想される場合
C. バックエンドシステムからの限定的な分離のみを伴う実用的なアプローチが適切であると判断された場合
D. 組織全体で広く使用されている既存のエンタープライズデータモデルがある場合
Answer: D (メーッセージを残す**)**

最新問題: **76**
一部の HTTP リクエストに対する応答は、リクエストで使用される HTTP 動詞に応じてキャッシュできます。
HTTP 仕様によれば、どの HTTP 動詞に対してこれを実行しても安全ですか?
A. PUT、POST、DELETE
B. GET、HEAD、POST
C. GET、PUT、オプション
D. GET、オプション、HEAD
Answer: D (メーッセージを残す**)**
GET、オプション、ヘッド
http://restcookbook.com/HTTP%20Methods/べき等性/

最新問題: **77**
次の順序のうち正しいものはどれですか?

A. APIクライアントはAPIを呼び出すロジックを実装します >> APIコンシューマはAPIへのアクセスを要求します >> API実装はリクエストをAPIにルーティングします
B. API コンシューマが API へのアクセスを要求 >> API クライアントが API を呼び出すロジックを実装 >> API がリクエストをルーティング >> API 実装
C. API コンシューマーは API を呼び出すロジックを実装します >> API クライアントは API へのアクセスを要求します >> API 実装は要求を API にルーティングします
D. APIクライアントはAPIを呼び出すロジックを実装します >> APIコンシューマはAPIへのアクセスを要求します >> APIはリクエストをルーティングします >> API実装
Answer: B (メーッセージを残す**)**
正解: API コンシューマが API へのアクセスを要求 >> API クライアントが API を呼び出すロジックを実装 >> API がリクエストをルーティング >> API 実装
****
>> API コンシューマーは、API を呼び出すロジックを実装しません。単なるロールです。したがって、"API コンシューマーは API を呼び出すロジックを実装します」というオプションは無効です。
>> API 実装はリクエストをルーティングしません。これは、ターゲット システムの機能が公開されるロジックの最終部分です。したがって、リクエストは他のエンティティによって API 実装にルーティングされる必要があります。したがって、"API 実装はリクエストを API にルーティングします」というオプションは無効です。
>> オプションの 1 つのステートメントは正しいですが、順序が間違っています。順序は、"API クライアントが API を呼び出すロジックを実装します >> API コンシューマーが API へのアクセスを要求します >> API が要求を API 実装にルーティングします」と示されています。ここで、オプションのステートメントは有効ですが、順序が間違っています。
>> 正しいオプションとシーケンスは、API コンシューマーが最初に Anypoint Exchange 上の API へのアクセスを要求し、クライアント資格情報を取得するものです。次に、API クライアントは、API コンシューマーによって要求されたアクセス クライアント資格情報を使用して API を呼び出すロジックを記述し、その要求は API マネージャーによって管理される API を介して API 実装にルーティングされます。

最新問題**: 78**
Anypoint Platform でクライアント管理に外部 ID プロバイダーを使用する場合の主な要件は何ですか?
A. Anypoint Platform にサインインするにはシングルサインオンが必要です
B. アプリケーションネットワークには、アイデンティティプロバイダと対話するシステムAPIが含まれている必要があります。
C. Anypoint Platform によって管理される OAuth 2.0 で保護された API を呼び出すには、API クライアントは同じアイデンティティ プロバイダによって発行されたアクセス トークンを送信する必要があります。

D. Anypoint Platform によって管理される API は、SAML 2.0 ポリシーによって保護される必要があります。
Answer: C (メーッセージを残す)
https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html 説明:
正解: Anypoint Platform によって管理される OAuth 2.0 で保護された API を呼び出すには、API クライアントは同じ ID プロバイダによって発行されたアクセス トークンを送信する必要があります。
****
>> クライアント管理には外部のIDプロバイダを使用しているため、Anypoint Platformにサインインするためにシングルサインオンは必要ありません。
>> クライアント管理に外部のアイデンティティプロバイダを使用しているため、Anypoint Platform によって管理されるすべての API を SAML 2.0 ポリシーで保護する必要はありません。
>> クライアント管理に外部 ID プロバイダーを使用しているため、アプリケーション ネットワークに ID プロバイダーと対話するシステム API を含める必要があるというのは正しくありません。指定されたオプションで正しいのは、「Anypoint Platform によって管理される OAuth 2.0 で保護された API を呼び出すには、API クライアントは同じ ID プロバイダーによって発行されたアクセス トークンを送信する必要があります」のみです。参照:
https://docs.mulesoft.com/api-manager/2.x/external-oauth-2.0-token-validation-policy
https://blogs.mulesoft.com/dev/api-dev/api-security-ways-to-authenticate-and-authorize/

最新問題**: 79**
以下にリストされている Anypoint Platform 機能のうち、API および API 呼び出し/コンシューマーのカテゴリに該当するものはどれですか?
2つ選択してください。
A. API 操作と管理
B. API ランタイムの実行とホスティング
C. API 消費者エンゲージメント
D. API 設計と開発
Answer: D (メーッセージを残す)
正解: API の設計と開発、API ランタイムの実行とホスティング
****
>> API 設計と開発 - Anypoint Studio、Anypoint Design Center、Anypoint Connectors
>> API ランタイムの実行とホスティング - Mule ランタイム、CloudHub、ランタイム サービス
>> API 運用と管理 - Anypoint API Manager、Anypoint Exchange
>> API コンシューマー管理 - API 契約、パブリック ポータル、Anypoint Exchange、API ノートブック
Explanation:
正解: API 運用と管理、API コンシューマーエンゲージメント
****
>> API 設計と開発 - Anypoint Studio、Anypoint Design Center、Anypoint Connectors

>> API ランタイムの実行とホスティング - Mule ランタイム、CloudHub、ランタイム サービス
>> API 運用と管理 - Anypoint API Manager、Anypoint Exchange
>> API コンシューマー管理 - API 契約、パブリック ポータル、Anypoint Exchange、API ノートブック
フォームの下部
フォームの先頭

最新問題: **80**
展示品を参照してください。
顧客がホストする Mule ランタイムを MuleSoft がホストする Anypoint Platform コントロールプレーン (ハイブリッド展開) と併用する場合、正しいのは何ですか?
A. Anypoint Runtime Manager は、Mule アプリケーションをデプロイするために Mule ランタイムへのネットワーク接続を開始します。
B. API実装は、コントロールプレーンと通信できない場合でも、顧客がホストするMuleランタイムで正常に実行できます。
C. Anypoint Runtime Manager は、ノード障害が発生した場合に新しい Mule ランタイムインスタンスを作成することで、コントロールプレーンの HA を自動的に確保します。
D. MuleSoft がホストする共有ロードバランサは、Mule ランタイムへの API 呼び出しの負荷分散に使用できます。
Answer: (解答を表示する**)**

最新問題: **81**
どの Mule アプリケーションで、その Mule アプリケーションによって公開されるエンドポイントに Anypoint Platform によって API ポリシーを適用できますか?
A) HTTP/1.x 経由でリクエストを受け入れる Mule アプリケーション
B) TCP経由でJSONリクエストを受け入れるが、応答を返す必要がないMuleアプリケーション
C) WebSocket経由でJSONリクエストを受け入れるMuteアプリケーション
D) HTTP/2 経由で gRPC リクエストを受け入れる Mule アプリケーション
A. オプションA
B. オプションD
C. オプションB
D. オプションC
Answer: C (メーッセージを残す**)**

最新問題: **82**
展示品を参照してください。
3 つのビジネス プロセスを実装する必要があり、実装では複数の異なる SaaS アプリケーションと通信する必要があります。

これらのプロセスは、個別の (サイロ化された) LOB によって所有され、主に互いに独立していますが、いくつかのビジネス エンティティを共有しています。各 LOB には 1 つの開発チームと独自の予算があります。この組織のコンテキストでは、データ モデルの冗長性を最小限に抑えてこれらのビジネス プロセスを実装する API の API データ モデルを選択する最も効果的な方法は何ですか。
A) ビジネスプロセスの一貫した部分と関連するビジネスエンティティの定義に一致する複数の境界付きコンテキストデータモデルを構築する
B) 確立されたマイクロサービスとアジャイルAPI中心のプラクティスに従うために、各APIごとに異なるデータモデルを構築する
C) XMLスキーマを使用してすべてのAPIデータモデルを構築し、組織全体で一貫性と再利用性を高める
D) 3つのビジネスプロセスのすべてのデータタイプを統合し、データモデルの一貫性と冗長性を確保した、集中型の標準データモデル（エンタープライズデータモデル）を構築する
A. オプションC
B. オプションB
C. オプションD
D. オプションA
Answer: (解答を表示する**)**

最新問題**: 83**
アップストリーム API とその実装を設計する際、ダウンストリーム API には信頼できる SLA がないため、ダウンストリーム API を呼び出すときにタイムアウトを設定しないよう開発チームにアドバイスされています。
これは、そのアップストリーム API の唯一のダウンストリーム API 依存関係です。
A. ダウンストリームAPIの呼び出しはタイムアウトせずに完了します。
B. アップストリームAPIのSLAは提供できません
C. ダウンストリーム API がクラッシュすることなく中断なく実行されると想定します。このアドバイスの影響は何でしょうか?
D. 下流のAPI実装が実行されるMuleランタイムによって、1000ミリ秒未満のToad依存のタイムアウトが適用されます。
E. アップストリームAPI実装が実行されるMuleランタイムによって、デフォルトのタイムアウト500ミリ秒が自動的に適用されます。
Answer: E (メーッセージを残す**)**

最新問題**: 84**
Anypoint Platform で API ポリシーが定義される場所と、それが API インスタンスにどのように適用されるかについて正しいのはどれですか?
A. API ポリシーは、Mule ランタイムへの API デプロイメントの一部として Runtime Manager で定義され、特定の API インスタンスにのみ適用されます。

B. API ポリシーは、特定の API インスタンスに対して API Manager で定義され、特定の API インスタンスにのみ適用されます。
C. APIポリシーはAPI Managerで定義され、すべてのAPIインスタンスに自動的に適用されます。
D. APIポリシーはAPI Managerで定義され、指定された環境内のすべてのAPIインスタンスに適用されます。
Answer: B (メーッセージを残す**)**
API ポリシーは、特定の API インスタンスに対して API Manager で定義され、特定の API インスタンスにのみ適用されます。
****
>> API 仕様が準備され、Exchange に公開されたら、API Manager にアクセスして、各 API の API インスタンスを登録する必要があります。
>> API マネージャーは、ポリシーを適用して NFR に対処するなど、API の側面の管理が行われる場所です。
>> 同じ API に対して複数のインスタンスを作成し、目的に応じて異なる方法で管理できます。
>> 1 つのインスタンスに API ポリシーのセットを適用し、同じ API の別のインスタンスに別の目的で異なるポリシーのセットを適用することができます。
>> これらの API とそのインスタンスは環境ごとに定義されます。したがって、各環境で個別に管理する必要があります。
>> プラットフォーム機能を使用して上位環境に昇格するときに、API インスタンスの同じ構成 (SLA、ポリシーなど) が昇格されることを保証できます。ただし、これはオプションのみです。必要に応じて、環境ごとに変更することもできます。
>> ランタイム マネージャーは、API 実装とその Mule ランタイムを管理する場所ですが、API 自体を管理する場所ではありません。API ポリシーは Mule ランタイムで実行されますが、ランタイム マネージャーで API ポリシーを適用することはできません。環境内の厳選されたインスタンスに対してのみ、API マネージャー経由でこれを行う必要があります。
したがって、これらの事実に基づくと、与えられた選択肢の正しい記述は、「API ポリシーは特定の API インスタンスに対して API Manager で定義され、特定の API インスタンスにのみ適用されます」です。

最新問題**: 85**
パブリック Anypoint Exchange ポータルを通じて API を共有する前に確認すべきことは何ですか?
A. 公開アクセスが必要なAPIインスタンスの可視性レベルは、パブリック可視性に設定する必要があります。
B. APIへのアクセスが必要なユーザーは、Anypoint Platformの適切なロールに追加する必要があります。
C. APIは少なくとも初期実装が展開され、ユーザーが操作できるようにアクセス可能で機能する必要があります。

D. データが侵害されないように、サポートされている認証/承認メカニズムのいずれかを使用して API を保護する必要があります。
Answer: (解答を表示する**)**
正解:パブリックにアクセスする必要がある API の API インスタンスの可視性レベルは、パブリック可視性に設定する必要があります。
****
参照：
https://docs.mulesoft.com/exchange/to-share-api-asset-to-portal

最新問題**: 86**
展示品を参照してください。
新しいプロモーション プロセス API の RAML 定義が提案され、Anypoint Exchange に公開されました。
プロモーション API の重要な消費者となるマーケティング部門には、満たさなければならない重要な要件と期待があります。
Anypoint Platform の機能を使用して、マーケティング部門をこの初期の API 設計フェーズに参加させる最も効果的な方法は何ですか?
A) マーケティング部門に、自動生成されたAPIコンソールを使用してAPIのモック実装を操作するように依頼します。
B) マーケティング部門のDBAと設計ワークショップを開催し、マーケティングITシステムのデータベーススキーマをRAMLに変換する。
C) Anypoint Studioを使用してAPIをMuleアプリケーションとして実装し、そのAPI実装をCloudHubにデプロイして、マーケティング部門に操作を依頼します。
D) APIデザイナーから統合テストスイートをエクスポートし、マーケティング部門にそのスイートのテストを実行させて合格することを確認する。
A. オプションC
B. オプションA
C. オプションD
D. オプションB
Answer: (解答を表示する**)**

最新問題**: 87**
Anypoint VPC のテクノロジー アーキテクチャについて正しいのは何ですか?
A. Anypoint VPCのプライベートIPアドレス範囲はCloudHubによって自動的に選択されます
B. Anypoint VPC にデプロイされた Mule アプリケーションとオンプレミス システム間のトラフィックは、プライベート ネットワーク内にとどまることができます。
C. 各CloudHub環境には個別のAnypoint VPCが必要です
D. VPC ピアリングを使用すると、基盤となる AWS VPC をオンプレミス (非 AWS) のプライベートネットワークにリンクできます。

Answer: B (メーッセージを残す)
説明
https://docs.mulesoft.com/runtime-manager/vpc-connectivity-methods-concept

最新問題**: 88**
Anypoint Platform でクライアント管理に外部 ID プロバイダーを使用する場合の主な要件は何ですか?
A. Anypoint Platform にサインインするにはシングルサインオンが必要です
B. アプリケーションネットワークには、アイデンティティプロバイダと対話するシステムAPIが含まれている必要があります。
C. Anypoint Platform によって管理される OAuth 2.0 で保護された API を呼び出すには、API クライアントは同じアイデンティティ プロバイダによって発行されたアクセス トークンを送信する必要があります。
D. Anypoint Platform によって管理される API は、SAML 2.0 ポリシーによって保護される必要があります。
Answer: C (メーッセージを残す)
説明
https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html

最新問題**: 89**
組織では、さまざまなクラウドベースの SaaS システムと複数のオンプレミス システムを使用しています。オンプレミス システムは組織のアプリケーション ネットワークの重要な部分であり、組織のイントラネット内からのみアクセスできます。
クラウドベースの SaaS システムとオンプレミス システムの両方との統合をサポートするために Anypoint Platform を構成して使用する最適な方法は何ですか?
A) Anypoint Platform Private Cloud Edition コントロールプレーンによって管理される Anypoint VPC で CloudHub でデプロイされた Mule ランタイムを使用する
B) MuleSoft がホストする Anypoint Platform コントロール プレーンによって管理される共有ワーカー クラウドで、CloudHub でデプロイされた Mule ランタイムを使用する
C) Anypoint Platform Private Cloud Edition コントロールプレーンによって管理され、外部ネットワークアクセスが一切ない完全に分離されたオンプレミスの Mule ランタイムのインストールを使用する
D) Cloud Hub でデプロイされたオンプレミス Mule ランタイムと、MuleSoft がホストする Anypoint Platform コントロール プレーンで管理される手動でプロビジョニングされたオンプレミス Mule ランタイムを組み合わせて使用する
A. オプションB
B. オプションD
C. オプションC
D. オプションA
Answer: A (メーッセージを残す)

最新問題: **90**
展示品を参照してください。
開発者は、クライアント ID 適用ポリシーによって管理されるステージング環境にデプロイされた API を呼び出すクライアント アプリケーションを構築しています。
API を正常に呼び出すには何が必要ですか?
A. Anypoint Platform アカウントのステージング環境のクライアント ID とシークレット
B. STAGING環境のAPIインスタンスのAnypoint Exchangeから取得したクライアントIDとシークレット
C. STAGING環境でAPIを所有するAnypoint Platformアカウントのクライアント IDとシークレット
D. Anypoint Platform から取得した有効な OAuth トークンとそれに関連付けられたクライアント ID およびシークレット
Answer: A (メーッセージを残す**)**

最新問題: **91**
展示品を参照してください。
3 つのビジネス プロセスを実装する必要があり、実装では複数の異なる SaaS アプリケーションと通信する必要があります。
これらのプロセスは、個別の (サイロ化された) LOB によって所有され、主に互いに独立していますが、いくつかのビジネス エンティティを共有しています。各 LOB には 1 つの開発チームと独自の予算があります。この組織のコンテキストでは、データ モデルの冗長性を最小限に抑えてこれらのビジネス プロセスを実装する API の API データ モデルを選択する最も効果的な方法は何ですか。
A) ビジネスプロセスの一貫した部分と関連するビジネスエンティティの定義に一致する複数の境界付きコンテキストデータモデルを構築する
B) 確立されたマイクロサービスとアジャイルAPI中心のプラクティスに従うために、各APIごとに異なるデータモデルを構築する
C) XMLスキーマを使用してすべてのAPIデータモデルを構築し、組織全体で一貫性と再利用性を高める
D) 3つのビジネスプロセスのすべてのデータタイプを統合し、データモデルの一貫性と冗長性を確保した、集中型の標準データモデル（エンタープライズデータモデル）を構築する
A. オプションA
B. オプションD
C. オプションB
D. オプションC
Answer: A (メーッセージを残す**)**

有効な **MCPA-Level-1** 問題集は GoShiken.com が提供された合格しやすい MCPA-Level-1 試験問題集！ GoShiken.com が最新の **MCPA-Level-1** 試験問題集を提供しています。GoShiken.com MCPA-Level-1 試験問題は最新で、解答が正確でございます。最新の GoShiken.com MCPA-Level-1 問題集をゲットする人はこちら: https://www.goshiken.com/MuleSoft/MCPA-Level-1-mondaishu.html (**154**30%OFF問題集溶と正解付きで **30%**ｗ特別割引コード: **Freepdfdumps**)

Valid MCPA-Level-1 Dumps shared by GoShiken.com for Helping Passing MCPA-Level-1 Exam! GoShiken.com now offer the **newest MCPA-Level-1 exam dumps**, the GoShiken.com MCPA-Level-1 exam **questions have been updated** and **answers have been corrected** get the **newest** GoShiken.com MCPA-Level-1 dumps with Test Engine here: https://www.goshiken.com/MuleSoft/MCPA-Level-1-mondaishu.html (**154** Q&As Dumps, **30%OFF Special Discount: Freepdfdumps**)