



Informatica® Intelligent Cloud Services
October 2022

Secure Agent サービス

© 著作権 Informatica LLC 2021, 2022

本ソフトウェアおよびマニュアルは、使用および開示の制限を定めた個別の使用許諾契約のもとでのみ提供されています。本マニュアルのいかなる部分も、いかなる手段（電子的複写、写真複写、録音など）によっても、Informatica LLC の事前の承諾なしに複製または転載することは禁じられています。

米政府の権利プログラム、ソフトウェア、データベース、および関連文書や技術データは、米国政府の顧客に配信され、「商用コンピュータソフトウェア」または「商業技術データ」は、該当する連邦政府の取得規制と代理店固有の補足規定に基づきます。このように、使用、複製、開示、変更、および適応は、適用される政府の契約に規定されている制限およびライセンス条項に従うものとし、政府契約の条項によって適当な範囲において、FAR 52.227-19、商用コンピュータソフトウェアライセンスの追加権利を規定します。

Informatica、Informatica Cloud、Informatica Intelligent Cloud Services、PowerCenter、PowerExchange、および Informatica ロゴは、米国およびその他の国における Informatica LLC の商標または登録商標です。Informatica の商標の最新リストは、Web (<https://www.informatica.com/trademarks.html>) にあります。その他の企業名および製品名は、それぞれの企業の商標または登録商標です。

本ソフトウェアまたはドキュメンテーション（あるいはその両方）の一部は、第三者が保有する著作権の対象となります。必要な第三者の通知は、製品に含まれています。

本マニュアルの情報は、予告なしに変更されることがあります。このドキュメントで問題が見つかった場合は、infa_documentation@informatica.com までご報告ください。

Informatica 製品は、それらが提供される契約の条件に従って保証されます。Informatica は、商品性、特定目的への適合性、非侵害性の保証等を含めて、明示的または黙示的ないかなる種類の保証をせず、本マニュアルの情報を「現状のまま」提供するものとします。

発行日: 2022-12-01

目次

序文	6
Informatica のリソース.....	6
Informatica マニュアル.....	6
Informatica Intelligent Cloud Services Web サイト.....	6
Informatica Intelligent Cloud Services コミュニティ.....	6
Informatica Intelligent Cloud Services マーケットプレイス.....	7
データ統合コネクタのドキュメント.....	7
Informatica ナレッジベース.....	7
Informatica Intelligent Cloud Services Trust Center.....	7
Informatica グローバルカスタマサポート.....	7
 第 1 章 : Secure Agent サービス	8
 第 2 章 : API Microgateway Service	11
API Microgateway サービスのプロパティの編集.....	12
Secure Agent または Secure Agent グループでの API Microgateway サービスの有効化.....	13
 第 3 章 : CMI ストリーミングエージェント	15
CMI ストリーミングエージェントのプロパティ.....	15
 第 4 章 : 共通統合コンポーネント	18
共通統合コンポーネントプロパティ.....	18
 第 5 章 : データベース取り込みサービス	21
データベース取り込みサービスのプロパティ.....	21
データベース取り込みエージェントの環境変数.....	25
 第 6 章 : Discovery Agent アプリケーション	26
Discovery Agent アプリケーションのプロパティ.....	26
 第 7 章 : データ統合サーバー	31
データ統合サーバーの回復機能.....	31
データ統合サーバーのプロパティ.....	32
 第 8 章 : エラスティックサーバー	34
エラスティックサーバーのプロパティ.....	34
 第 9 章 : ファイル統合サービス	36
 第 10 章 : GitRepoConnectApp	37
ローカルリポジトリのベースディレクトリ.....	37

GitRepoConnectApp のプロパティ.....	38
第 11 章：一括取り込み（ファイル）.....	40
第 12 章：メタデータ基盤アプリケーション.....	43
メタデータ基盤アプリケーションのプロパティ.....	43
第 13 章：プロセスサーバー.....	49
プロセスサーバーのプロパティ.....	49
デフォルト接続データベースのプロパティ.....	55
ログレベル.....	55
プロセスサーバーのサイズ決定に関する推奨事項.....	56
Secure Agent との通信.....	57
プロセスサーバーのための Secure Agent の設定.....	58
単一の Secure Agent へのデプロイ.....	58
Secure Agent グループへのデプロイ.....	59
Secure Agent クラスタへのデプロイ.....	60
PostgreSQL データベースのインストールとアップグレードの前提条件.....	61
Windows での PostgreSQL データベースの管理.....	62
Windows での PostgreSQL データベースのバックアップ.....	62
Windows での PostgreSQL データベースのリストア.....	62
Windows での PostgreSQL データベースのリセット.....	63
Windows での PostgreSQL サーバーの起動.....	63
Windows での PostgreSQL サーバーの停止.....	63
Windows での PostgreSQL サーバーステータスの取得.....	63
Windows での PostgreSQL データベースのクリーンアップ.....	64
Windows での PostgreSQL データベースの再インデックス化.....	64
Windows でのトランザクションログのリセット.....	64
Linux での PostgreSQL データベースの管理.....	65
Linux での PostgreSQL データベースのバックアップ.....	65
Linux での PostgreSQL データベースのリストア.....	66
Linux での PostgreSQL データベースのリセット.....	66
Linux での PostgreSQL サーバーの起動.....	66
Linux での PostgreSQL サーバーの停止.....	66
Linux での PostgreSQL サーバーステータスの取得.....	67
Linux での PostgreSQL データベースのクリーンアップ.....	67
Linux での PostgreSQL データベースの再インデックス化.....	67
Linux でのトランザクションログのリセット.....	68
PostgreSQL データベースのアップグレード.....	68
レプリケーション技術を使用した PostgreSQL データベースのアップグレード.....	68
PostgreSQL 構成ファイル.....	69
PostgreSQL ログローテーションの設定.....	69
プロセスサーバーに対するパブリック証明書とプライベートキーの設定.....	70

スループットを向上させるためのスレッドプールプロファイルの設定.	71
第 14 章 : Secure Agent サービスプロパティの設定.....	73
索引.....	75

序文

「Secure Agent サービス」を使用して、Informatica Intelligent Cloud ServicesSM Secure Agent がデータ処理に使用するマイクロサービスについて確認します。サービスプロパティを設定する方法を確認します。

Informatica のリソース

Informatica は、Informatica Network やその他のオンラインポータルを通じてさまざまな製品リソースを提供しています。リソースを使用して Informatica 製品とソリューションを最大限に活用し、その他の Informatica ユーザーや各分野の専門家から知見を得ることができます。

Informatica マニュアル

Informatica マニュアルポータルでは、最新および最近の製品リリースに関するドキュメントの膨大なライブラリを参照できます。マニュアルポータルを利用するには、<https://docs.informatica.com> にアクセスしてください。

製品マニュアルに関する質問、コメント、ご意見については、Informatica マニュアルチーム (infa_documentation@informatica.com) までご連絡ください。

Informatica Intelligent Cloud Services Web サイト

Informatica Intelligent Cloud Services Web サイト (<http://www.informatica.com/cloud>) にアクセスできます。このサイトには、Informatica Cloud 統合サービスに関する情報が含まれます。

Informatica Intelligent Cloud Services コミュニティ

Informatica Intelligent Cloud Services コミュニティを使用して、技術的な問題について議論し、解決します。また、技術的なヒント、マニュアルの更新情報、FAQ（よくある質問）への答えを得ることもできます。

次の Informatica Intelligent Cloud Services コミュニティにアクセスします。

<https://network.informatica.com/community/informatica-network/products/cloud-integration>

開発者は、次の Cloud 開発者コミュニティで詳細情報を確認したり、ヒントを共有したりできます。

<https://network.informatica.com/community/informatica-network/products/cloud-integration/cloud-developers>

Informatica Intelligent Cloud Services マーケットプレイス

Informatica マーケットプレイスにアクセスすると、データ統合コネクタ、テンプレート、およびマップレットを試用したり購入したりできます。

<https://marketplace.informatica.com/>

データ統合コネクタのドキュメント

データ統合コネクタのドキュメントには、マニュアルポータルからアクセスできます。マニュアルポータルを利用するには、<https://docs.informatica.com> にアクセスしてください。

Informatica ナレッジベース

Informatica ナレッジベースを使用して、ハウツー記事、ベストプラクティス、よくある質問に対する回答など、製品リソースを見つけることができます。

ナレッジベースを検索するには、<https://search.informatica.com> にアクセスしてください。ナレッジベースに関する質問、コメント、ご意見の連絡先は、Informatica ナレッジベースチーム (KB_Feedback@informatica.com) です。

Informatica Intelligent Cloud Services Trust Center

Informatica Intelligent Cloud Services Trust Center は、Informatica のセキュリティポリシーおよびリアルタイムでのシステムの可用性について情報を提供します。

Trust Center (<https://www.informatica.com/trust-center.html>) にアクセスします。

Informatica Intelligent Cloud Services Trust Center にサブスクライブして、アップグレード、メンテナンス、およびインシデントの通知を受信します。[Informatica Intelligent Cloud Services Status](#) ページには、すべての Informatica Cloud 製品の実稼働ステータスが表示されます。メンテナンスの更新はすべてこのページに送信され、停止中は最新の情報が表示されます。更新と停止の通知がされるようにするには、Informatica Intelligent Cloud Services の 1 つのコンポーネントまたはすべてのコンポーネントについて更新の受信をサブスクライブします。すべてのコンポーネントにサブスクライブするのが、更新を逃さないようにするための最良の方法です。

登録するには、<https://status.informatica.com/> に移動し、**[更新を購読登録]** をクリックします。その後、電子メール、SMS テキストメッセージ、Webhook、RSS フィードとして、またはこの 4 つを任意に組み合わせて送信された通知を受信することを選択ができます。

Informatica グローバルカスタマサポート

電話またはオンラインでカスタマサポートセンターに連絡できます。

オンラインサポートについては、Informatica Intelligent Cloud Services の **[サポート要求の送信]** をクリックしてください。またオンラインサポートを使用して問題を記録することもできます。オンラインサポートを利用するには、ログインが必要です。<https://network.informatica.com/welcome> でログイン要求できます。

Informatica グローバルカスタマサポートの電話番号は、Informatica の Web サイト <https://www.informatica.com/services-and-training/support-services/contact-us.html> に掲載されています。

第 1 章

Secure Agent サービス

Secure Agent サービスは、Secure Agent がデータ処理に使用するプラグブルマイクロサービスです。例えば、Secure Agent はデータ統合サーバーを使用してデータ統合ジョブを実行し、プロセスサーバーを使用してアプリケーション統合を実行してオーケストレーションジョブを処理します。各 Secure Agent サービスは、エージェントで実行されている他のサービスとは独立して実行されます。

独立したサービスアーキテクチャには、次の利点があります。

- コネクタまたはパッケージを追加したときに、Secure Agent が再起動しない。
- サービスは、別のサービスの再起動時に影響を受けない。例えば、データ統合サーバーを再起動しても、プロセスオーケストレーションジョブは引き続き実行されます。
- アップグレード中のダウンタイムは最小化されます。アップグレードプロセスは、Secure Agent の新しいバージョンをインストールし、コネクタパッケージを更新し、データ統合サーバーの構成の変更を適用します。ダウンタイムを最小化するために、古いエージェントは引き続き使用可能なままで、アップグレード中にデータ統合ジョブを実行し続けます。Secure Agent の新しいバージョンは、アップグレードプロセスの完了後に開始されるジョブを実行します。

Secure Agent で実行されるサービスは、Secure Agent グループで有効なライセンスと Informatica Intelligent Cloud Services によって異なります。

以下の表に、エージェントで実行される Secure Agent サービスと、それらのサービスを使用する Informatica Intelligent Cloud Services を示します。

Secure Agent サービス	説明	次により使用
API マイクロゲートウェイサービス	Secure Agent で実行されるアプリケーションの統合プロセスを管理します。	アプリケーションの統合、API マネージャ
B2B プロセッサ	B2B Gateway のインバウンドおよびアウトバウンドプロセスフローを実行します。 注: Informatica グローバルカスタマサポートから指示された場合を除き、このサービスの値は変更しないでください。	B2B Gateway
CIH プロセッサ	プライベートパブリケーションリポジトリを使用する組織のために、Cloud Integration Hub のパブリケーションおよびサブスクリプションを実行します。	Cloud Integration Hub
CMI ストリーミングエージェント	一括取り込みサービスでストリーミング取り込みジョブを実行します。	一括取り込みサービス
共通統合コンポーネント	シェルスクリプトまたはバッチコマンドをタスクフローのコマンドタスクステップで実行します。	データ統合

Secure Agent サービス	説明	次により使用
データベース取り込み	一括取り込みサービスでアプリケーション取り込みジョブとデータベース取り込みジョブを実行します。	一括取り込みサービス
データ統合サーバー	マッピング、タスク、およびタスクフローインスタンスなどのデータ統合ジョブを実行します。	B2B Gateway Cloud Integration Hub、 Azure のデータアクセラレータ、 データ統合 データプロファイリング
Discovery Application Agent	ステージングされたプロファイリング結果をデータ統合サーバーまたは詳細クラスタで読み取り、プロファイリング結果をメタデータコマンドセンターにアップロードします。	メタデータコマンドセンター
EDC 検索エージェント	Azure のデータアクセラレータのため、およびデータ統合でのデータカタログ検出のために、Enterprise Data Catalog データアセットを検出します。 注: Informatica グローバルカスタマサポートから指示された場合を除き、このサービスの値は変更しないでください。	Azure のデータアクセラレータ、 データ統合
エラスティックサーバー	詳細クラスタおよびそのクラスタで実行するジョブを管理します。	データ統合
ファイル統合サービス	リモートサーバーとのファイルの送信または受信、あるいはその両方に、HTTPS、AS2 および SFTP などのファイル転送プロトコルを使用します。	B2B Gateway、データ統合
GitRepoConnectApp	組織がオンプレミスのソース管理リポジトリを使用している場合、Informatica Intelligent Cloud Services とソース管理リポジトリ間の通信を管理します。	ソース管理を使用するすべての Informatica Intelligent Cloud Services
一括取り込み	ファイル取り込みタスクおよびファイルリスナジョブを実行します。 注: Informatica グローバルカスタマサポートから指示された場合を除き、このサービスの値は変更しないでください。	一括取り込みサービス
メタデータ基盤アプリケーション	組織内で設定されているソースシステムからメタデータを抽出し、抽出したメタデータを Secure Agent を介してメタデータコマンドセンターにアップロードします。	メタデータコマンドセンター

Secure Agent サービス	説明	次により使用
OI データコレクタ	PowerCenter、Data Engineering Integration、および Data Quality から運用データとドメイン関連のメタデータを収集するオペレーションインサイトデータコレクタを実行します。 注: Informatica グローバルカスタマサポートから指示された場合を除き、このサービスの値は変更しないでください。	オペレーションインサイト
プロセスサーバー	アプリケーション統合プロセス、コネクタ、および接続を実行します。	アプリケーションの統合、 アプリケーション統合コンソール

それぞれの Secure Agent サービスには、Tomcat 設定や Tomcat JRE 設定などの一意の設定プロパティセットがあります。パフォーマンスを最適化するため、または Informatica グローバルカスタマサポートから指示された場合は、サービスを設定したり、サービスのプロパティを変更しなければならないことがあります。Secure Agent サービスは、エージェントで実行されている他のサービスとは独立して実行されます。

第 2 章

API Microgateway Service

API Microgateway Service は、組織のオンプレミスの Secure Agent 上で実行されるアプリケーションの統合プロセスを管理します。API Microgateway サービスを使用して、管理対象 API を API Microgateway プロキシとして公開します。

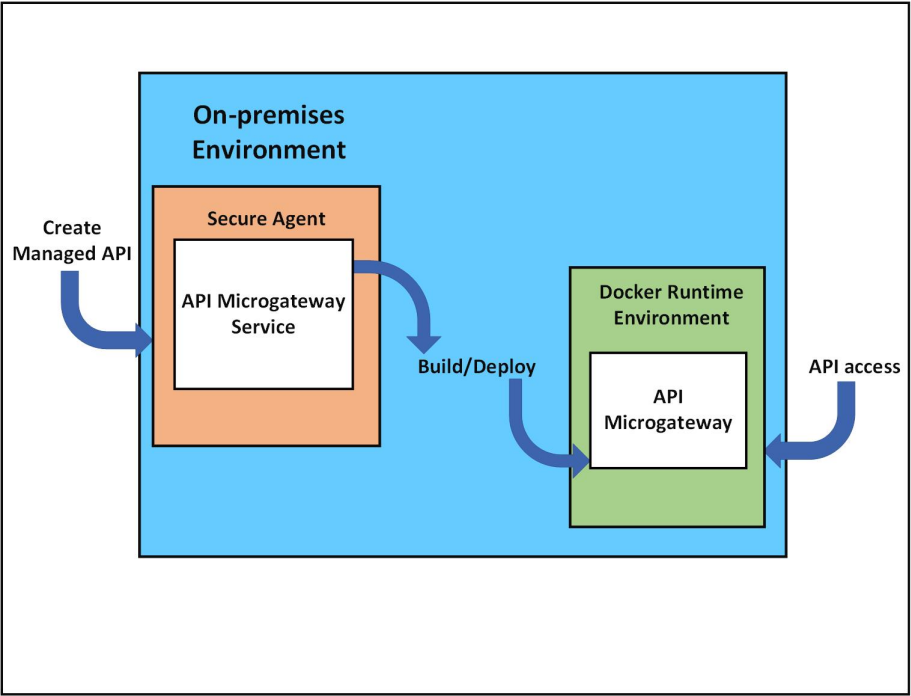
次の方法を使用して、API Microgateway サービスでパブリッシュする管理対象 API へのアクセスを制御できます。

- IP フィルタリングポリシー
- レート制限ポリシー
- 基本認証または OAuth 2.0 認証

API Microgateway サービスは、API Microgateway プロキシを作成およびデプロイするための REST API を提供します。API コンシューマは、組織のオンプレミス環境に API Microgateway プロキシとしてデプロイされた管理対象 API にアクセスします。アプリケーションの統合プロセスは、REST サービスの URL および SOAP サービスの URL のエンドポイントを公開します。

API Microgateway サービスを使用して、管理する API エンドポイントへの API Microgateway プロキシを構築します。API Microgateway Service は、組織の Secure Agent マシン上に不変の Docker イメージとして API Microgateway を構築します。次に、API Microgateway サービスを使用して、API アクセス用の Secure Agent Docker ランタイム環境のコンテナに Docker イメージをデプロイします。API Microgateway は、リクエストをアプリケーションの統合エンドポイントに転送する前に設定した API アクセスポリシーを適用します。

次の図は、オンプレミス環境で管理対象 API を公開する API Microgateway サービスおよび API Microgateway コンポーネントを示しています。



Secure Agent Docker ランタイム環境は、Blue-Green デプロイメントストラテジを使用して Docker イメージをホストし、API Microgateway コンポーネントの更新中のダウンタイムをゼロにします。

API Microgateway サービスのプロパティの編集

API Microgateway サービスのプロパティは Administrator で編集します。

次の図は、**【システム構成の詳細】** 領域で編集できる API Microgateway サービスのプロパティを示しています。

▼ System Configuration Details		
Service:	API Microgateway Service ▼	
Type:	All Types ▼	
Type	Name	Value
AGENT_RUNTIME_SETTINGS	project-name	'project1'
AGENT_RUNTIME_SETTINGS	docker-registry-name	'infa.agent.apimgw'
DOCKER_CONTAINER_SETTINGS	blue	http-port: '16090' https-port: '16095'
DOCKER_CONTAINER_SETTINGS	green	http-port: '17090' https-port: '17095'
DOCKER_CONTAINER_SETTINGS	haproxy	http-port: '6090' https-port: '6095'

次の表に、API Microgateway サービスの設定を示します。

タイプ	名前	説明
AGENT_RUNTIME_SETTINGS	project-name	API 設定を保存するプロジェクトの名前。新しいプロジェクトを作成する場合など、必要に応じて名前を変更できます。 注: プロジェクト名には、/または/0 という文字を含めることはできません。プロジェクト名に制限された文字が含まれている場合、プロジェクトの作成は失敗します。
AGENT_RUNTIME_SETTINGS	docker-registry-name	Secure Agent マシン上の名前付きおよびタグ付きの API Microgateway Docker イメージをすべて含むローカルの Docker レジストリの名前。 注: Docker イメージとタグの名前には、次の文字を含めることはできません: - , _ . Docker イメージまたはタグ名に制限された文字が含まれている場合、イメージの構築は失敗します。
DOCKER_CONTAINER_SETTINGS	blue	Secure Agent マシンに最初にデプロイされる Docker イメージコンテナ (green と交互になります)。 blue コンテナの次のポートを変更することができます。 - http-port。デフォルト値は 16090 です。 - https-port。デフォルト値は 16095 です。
DOCKER_CONTAINER_SETTINGS	green	Secure Agent マシンに 2 番目にデプロイされる Docker イメージコンテナ (blue と交互になります)。 green コンテナの次のポートを変更することができます。 - http-port。デフォルト値は 17090 です。 - https-port。デフォルト値は 17095 です。
DOCKER_CONTAINER_SETTINGS	haproxy	Secure Agent マシン上の Docker イメージコンテナのルーター。blue と green のコンテナ間でトラフィックを切り替えます。 haproxy コンテナの次のポートを変更することができます。 - http-port。デフォルト値は 6090 です。 - https-port。デフォルト値は 6095 です。

注: API Microgateway を停止するには、3 つの Docker イメージコンテナをすべて停止します。

Secure Agent または Secure Agent グループでの API Microgateway サービスの有効化

API Microgateway サービスを実行する Secure Agent を変更する場合は、管理者で Secure Agent または Secure Agent グループに対して API Microgateway サービスを有効にします。Secure Agent グループに対して API Microgateway サービスを有効にすると、このサービスは、グループ内のすべての Secure Agent と、後でグループに追加するすべての Secure Agent に対して有効になります。

1. **【ランタイム環境】** ページに移動し、Secure Agent または Secure Agent グループの **【アクション】** メニューから **【サービスの有効化または無効化、コネクタ】** を選択します。

【エージェントのコンポーネントの有効化/無効化】 ウィンドウが表示されます。

2. サービスのリストから **【API Microgateway】** を選択し、**【保存】** をクリックします。

API Microgateway サービスが Secure Agent または Secure Agent グループに対して有効になります。

第 3 章

CMI ストリーミングエージェント

CMI ストリーミングエージェントを使用して、ストリーミング取り込みタスクを定義し、展開します。ストリーミング取り込みタスクを一括取り込みサービスで設定します。

CMI ストリーミングエージェントは、オンプレミスシステムで実行され、一括取り込みストリーミングサービスと連携して動作します。オンプレミスシステムで、CMI ストリーミングエージェントは一括取り込みストリーミングで展開されたジョブを実行します。エージェントは各ジョブのステータスおよび統計情報を更新します。

Linux でエージェントのインストールディレクトリ名にスペースが含まれている場合、CMI ストリーミングエージェントが起動しません。エージェントは接続タイムアウトステータスを返します。再起動を数回試行した後に、エージェントはエラー状態になります。

注: Informatica Intelligent Cloud Services 一括取り込みサービスの 2020 年 4 月のリリースよりも前、CMI ストリーミングエージェントは、ストリーミング取り込みエージェントという名前でした。

CMI ストリーミングエージェントのプロパティ

CMI ストリーミングエージェントの動作を変更または最適化するには、ランタイム環境でエージェントプロパティを設定します。CMI ストリーミングエージェントのプロパティは、Secure Agent の編集時に **【システム構成の詳細】** 領域で設定します。

CMI ストリーミングエージェントのエンジン、エージェント、およびスクリプトのプロパティを設定できます。

次の図は、CMI ストリーミングエージェントのプロパティの一部を示しています。

▼ System Configuration Details

Service:

CMI Streaming Agent

Type:

All Types

Type	Name	Value
Engine	MaxLogFileSize	'5MB'
Engine	LogLevel	'DEBUG'
Agent	DataflowPullInterval	60
Agent	JVM	'-Xms256M -Xmx256M'
Agent	LogLevel	'DEBUG'
Agent	MaxLogFileSize	'10MB'
Agent	MaxNumberOfBackups	5
Scripts	LogLevel	'DEBUG'
Scripts	MaxFileSize	'5MB'
Scripts	MaxBackupIndex	5

CMI ストリーミングエージェントの次のプロパティを設定できます。

タイプ	プロパティ名	説明
エンジン	MaxLogFileSize	エンジンが作成可能なログファイルの最大サイズ。 デフォルトは 5 MB です。
エンジン	LogLevel	エンジンのログレベル。
エージェント	DataflowPullInterval	エージェントがタスクで更新を確認するまでの間隔。 デフォルトは 60 秒です。
エージェント	JVM	エージェントの JVM プロパティのリスト。例：[-Xms256M -Xmx256M]
エージェント	LogLevel	エージェントのログレベル。
エージェント	MaxLogFileSize	エージェントが作成可能なログファイルの最大サイズ。 デフォルトは 10MB です。

タイプ	プロパティ名	説明
エージェント	MaxNumberOfBackups	エージェントのバックアップログファイルの最大数。 デフォルトは 5 です。
スクリプト	LogLevel	スクリプトのログレベル。
スクリプト	MaxFileSize	最大ファイルサイズ。この最大ファイルサイズに達した後、ログはロールオーバーされ、新しいファイルが作成されます。 デフォルトは 10MB です。
スクリプト	MaxBackupIndex	ロールオーバー後に保持するバックアップファイルの最大数。 デフォルトは 5 です。

第 4 章

共通統合コンポーネント

共通統合コンポーネントサービスは、タスクフローのコマンドタスクステップ内で指定されたコマンドを実行する Secure Agent サービスです。

共通統合コンポーネントサービスを表示または設定するには、共通統合コンポーネントサービスとコマンド実行者パッケージのライセンスを所有している必要があります。

いくつかのサービスプロパティを設定して、共通統合コンポーネントサービスのパフォーマンスを最適化できます。サービスプロパティは、Secure Agent の編集時に変更できます。

共通統合コンポーネントサービスが処理するすべての要求は、次のディレクトリに記録されます。

<Secure Agent インストールディレクトリ>\apps\Common_Integration_Components\logs\<バージョン>

各コマンドタスクのログファイルは、次のディレクトリ内で参照できます。

<Secure Agent インストールディレクトリ>\apps\Common_Integration_Components\logs\command\<コマンドジョブ ID>

サーバーレスランタイム環境では、Secure Agent は各コマンドタスクのログファイルを Amazon S3 にプッシュします。

共通統合コンポーネントプロパティ

共通統合コンポーネントサービスの動作を変更する、または最適化するには、Secure Agent の編集時にプロパティを **【システム構成の詳細】** セクションで設定します。

次の画像は、共通統合コンポーネントサービスのプロパティを示しています。

▼ System Configuration Details

Service: Common Integration Components ▼

Type: All Types ▼

Type	Name
Tomcat	NetworkTimeoutPeriod
Tomcat	JRE_OPTS
Platform	LCM_JRE_OPTS
SYSTEM_CFG	HTTP_CONNECTION_TIMEOUT_SECONDS
SYSTEM_CFG	HTTP_SOCKET_TIMEOUT_SECONDS
COMMAND_CFG	MaximumConcurrentJobs

以下の共通統合コンポーネントサービスのプロパティを設定できます。

タイプ	名前	説明
Tomcat	JRE_OPTS	Apache Tomcat プロセスの JRE VM オプション。
プラットフォーム	LCM_JRE_OPTS	Apache Tomcat プロセスを開始する、停止する、またはステータスを取得するための JRE オプション。 注: Informatica グローバルカスタマサポートから指示されない限り、このプロパティ値は変更しないでください。
SYSTEM_CFG	HTTP_CONNECTION_TIMEOUT_SECONDS	Secure Agent が Informatica Intelligent Cloud Services と通信するための HTTP 接続を設定するために待機する秒単位での最大時間。 デフォルトは 60 です。 注: Informatica グローバルカスタマサポートから指示されない限り、このプロパティ値は変更しないでください。
SYSTEM_CFG	HTTP_SOCKET_TIMEOUT_SECONDS	Secure Agent と Informatica Intelligent Cloud Services との間の HTTP 接続でデータパケットの転送中の秒単位での最大アイドル時間。 デフォルトは 60 です。 注: Informatica グローバルカスタマサポートから指示されない限り、このプロパティ値は変更しないでください。

タイプ	名前	説明
COMMAND_CFG	MaximumConcurrentJobs	<p>単一の Secure Agent によって実行できる同時コマンドタスクの最大数。</p> <p>1 つの Secure Agent グループ内の各 Secure Agent のデフォルト値は 10 です。</p> <p>例えば、1 つの Secure Agent グループ内に 3 つの Secure Agent がある場合、このサービスが処理できる同時コマンドタスクの最大数は 30 です。</p> <p>最大制限を超えたすべてのコマンド実行要求はキュー入れられ、Secure Agent が使用可能になると実行されます。</p>
<p>注: Informatica グローバルカスタマサポートから指示されない限り、共通統合コンポーネントサービスの他のプロパティ値は変更しないでください。</p>		

第 5 章

データベース取り込みサービス

一括取り込みアプリケーションと一括取り込みデータベースはデータベース取り込みエージェントサービスを使用して、取り込みジョブを実行します。

Secure Agent をランタイム環境にダウンロードしてデータベース取り込みサービスを有効にすると、Secure Agent が実行されているオンプレミスのシステムにデータベース取り込みパッケージがプッシュされます。その後、必要に応じて Secure Agent で実行されるデータベース取り込みサービスのプロパティを設定できます。

データベース取り込みサービスのプロパティ

Secure Agent グループが使用するデータベース取り込みサービスの動作を変更または最適化するには、ランタイム環境にデータベース取り込みプロパティを設定します。

プロパティを設定するには、ランタイム環境を開き、**[編集]** をクリックします。**[システム構成の詳細]** で、**[データベース取り込み]** サービスと **[DBMI_AGENT_CONFIG]** タイプを選択します。

次の表に、データベース取り込みエージェントサービスのプロパティを示します。

プロパティ	説明
maxTaskUnits	<p>Secure Agent が実行されているオンプレミスマシンで同時に実行できるアプリケーション取り込みタスクとデータベース統合タスクの最大数。</p> <p>Secure Agent マシンの適切なタスクユニット数を計算するには、コア数を 3 または 4 で割ることをお勧めします。例えば、8 コアのマシンを使用している場合は、このプロパティを 2 に設定できます。その後、CPU 使用率を監視し、必要に応じてプロパティ値を調整してパフォーマンスチューニングを行います。</p> <p>初期ロード処理中、このプロパティは同時にアンロードできるテーブルの数を決定します。残りのテーブルはキューに入れられ、リソースが使用可能になるとアンロード処理を開始します。</p> <p>注: 1 つのジョブで多くのテーブルを処理できます。処理できるテーブルの総数の制限となるのは、使用可能なメモリのみです。1KB の行サイズに基づく初期ロードタスクには、平均してテーブルごとに 25MB の RAM が必要です。</p> <p>増分ロード処理中、このプロパティは同時に実行できるアプリケーション取り込みジョブとデータベース統合ジョブの数を決定します。</p> <p>このプロパティを Secure Agent マシンのコア数よりも大きい値に設定すると、タスク実行の並列処理が増える可能性があります。タスク実行時にパフォーマンスのボトルネックが発生する可能性もあります。</p>
serviceLogRetentionPeriod	<p>最終更新がファイルに書き込まれた後に、各内部データベース取り込みサービスログファイルが保持される日数。この保持期間が経過すると、ログファイルは削除されます。デフォルト値は 7 日です。</p> <p>サービスログは、それらが作成された Secure Agent ホスト (<infaagent>/apps/Database_Ingestion/logs) に保持されます。</p> <p>注: このプロパティは一括取り込みアプリケーションと一括取り込みデータベースの両方に適用されます。</p>
taskLogRetentionPeriod	<p>最終更新がジョブログファイルに書き込まれた後、各ジョブログファイルを保存する日数。この保持期間が経過すると、ログファイルは削除されます。デフォルト値は 7 日です。</p>

プロパティ	説明
ociPath	<p>Oracle ソースおよびターゲットの場合、Oracle Call Interface (OCI) ファイル oci.dll または libclntsh.so へのパス。実行中の DBMI エージェントの場合、この値は Windows の PATH 環境変数で指定されたパス、または Linux の LD_LIBRARY_PATH 環境変数で指定されたパスに付加されます。</p> <p>注: このプロパティは一括取り込みデータベースにのみ適用されます。</p>
serviceUrl	<p>データベース取り込みサービスが Informatica Intelligent Cloud Services クラウドへの接続に使用する URL。</p> <p>注: このプロパティは一括取り込みアプリケーションと一括取り込みデータベースの両方に適用されます。</p>
logLevel	<p>データベース取り込みサービスが生成するログに含める詳細レベル。次のオプションがあります。</p> <ul style="list-style-type: none"> - トレース - デバッグ - 情報 - 警告 - エラー <p>デフォルト値はトレースです。</p> <p>注: このプロパティは一括取り込みアプリケーションと一括取り込みデータベースの両方に適用されます。</p>
taskExecutionHeapSize	<p>タスク実行サービスの最大ヒープサイズ（ギガバイト単位）。この値は、maxTaskUnits プロパティとともに、Secure Agent で実行可能な同時アプリケーション取り込みタスクと同時データベース統合タスクの数に影響します。より多くのタスクを同時に実行するには、ヒープサイズを増やすことをお勧めします。この値に続けてギガバイトの場合は「g」と入力します（例: 「9g」）。デフォルト値は「8g」です。</p> <p>注: このプロパティは一括取り込みアプリケーションと一括取り込みデータベースの両方に適用されます。</p>
useProxy	<p>このプロパティを true に設定すると、ターゲットへの接続時およびターゲットへのデータの書き込み時に DBMI エージェントがプロキシを通過できるようになります。次に、DBMI エージェントは、Secure Agent プロキシ構成のプロキシ設定を使用します。デフォルトでは、プロキシ設定は使用されません。</p> <p>注: このプロパティは一括取り込みアプリケーションと一括取り込みデータベースの両方に適用されます。</p>

プロパティ	説明
intermediateStorageDirectory	<p>増分ロードジョブおよび初期ロードと増分ロードの組み合わせジョブの場合、関連するタスク定義で【永続ストレージの有効化】オプションが選択されているときに、データを含む中間ファイルが保存されるローカルルートディレクトリです。</p> <p>注: このプロパティは一括取り込みデータベースにのみ適用されます。</p>
storageBackupDirectory	<p>増分ロードジョブおよび初期ロードと増分ロードの組み合わせジョブの場合、関連するタスク定義で【永続ストレージの有効化】オプションが選択されているときに、バックアップファイルが保存されるディレクトリへのパスです。</p> <p>注: このプロパティは一括取り込みデータベースにのみ適用されます。</p>
storageProperties	<p>増分ロードジョブおよび初期ロードと増分ロードの組み合わせジョブの場合、関連するタスク定義で【永続ストレージの有効化】オプションが選択されているときに使用されるキー=値のペアのカンマ区切りリストです。このプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。</p> <p>注: このプロパティは一括取り込みデータベースにのみ適用されます。</p>
task_container.jvm.allowExceptionForInvalidEncodedData	<p>無効なエンコーディングを UTF-8 に報告する文字変換エラーを受け取った場合にソースデータを修復または修正しないようにするには、ソースからデータをアンロードしようとしたときにデータベース取り込みジョブが失敗しないようにこのプロパティを <code>false</code> に設定します。この設定を使用するとデータベース取り込みサービスによって同等の Java プロパティが DataDirect JDBC ドライバに渡されるため、例外が発生しなくなります。このプロパティを設定した後に、データベース取り込みサービスを再開する必要があります。</p> <p>注: このプロパティは一括取り込みデータベースにのみ適用されます。</p>
testProperty	<p>このプロパティは設定しないでください。このプロパティは、Informatica グローバルカスタマサポートおよび技術スタッフによる内部使用を目的としています。このプロパティは、【タイプ】 フィールドで DBMI_AGENT_ENV を選択した場合にのみ表示されます。</p>

データベース取り込みエージェントの環境変数

データベース取り込みエージェントの動作を変更または最適化するには、次の環境変数を定義します。

環境変数	説明
DBMI_REPLACE_UNSUPPORTED_CHARS	<p>Microsoft Azure Synapse Analytics ターゲットの場合に、アプリケーション取り込みジョブまたはデータベース取り込みジョブが、ターゲットが正しく処理できない文字データ内の文字を置き換えるかどうかを指定します。文字の置き換えを有効にするには、この環境変数を <code>true</code> に設定します。</p> <p><code>DBMI_REPLACE_UNSUPPORTED_CHARS=true</code></p> <p>設定後、一括取り込みアプリケーションまたは一括取り込みデータベースは、<code>DBMI_UNSUPPORTED_CHARS_REPLACEMENT</code> 環境変数に指定されている文字を使用して、サポートされていない文字を置き換えます。</p>
DBMI_UNSUPPORTED_CHARS_REPLACEMENT	<p><code>DBMI_REPLACE_UNSUPPORTED_CHARS</code> 環境変数が <code>true</code> に設定されている場合に、Microsoft Azure Synapse Analytics ターゲットが正しく処理できないソースデータ内の文字を置き換える文字を指定します。</p> <p>デフォルト値: ? (疑問符)</p> <p>注: この環境変数は一括取り込みデータベースに対してのみ定義します。</p>
DBMI_WRITER_CONN_POOL_SIZE	<p>アプリケーション取り込みジョブまたはデータベース統合ジョブが変更データをターゲットにプロパゲートするために使用する接続の数を示します。デフォルト値は 8 です。有効な値は 4~8 です。</p>
DBMI_WRITER_RETRIES_MAX_COUNT	<p>データベース統合ジョブがソースデータを Amazon S3 または Microsoft Azure Data Lake Storage Gen2 ターゲットにロードしている最中にネットワークの問題が発生した場合に、データベース統合ジョブで初期ロードまたは増分ロードを続行する要求を再試行する最大回数を指定します。再試行がすべて失敗した場合、ジョブは失敗となります。</p> <p>デフォルト値は 5 です。</p>
DBMI_WRITER_RETRIES_INTERVAL_IN_MILLIS	<p>ネットワークで問題が発生した場合に、データベース統合ジョブが Amazon S3 または Microsoft Azure Data Lake Storage Gen2 ターゲットへの初期ロードまたは増分ロードを続行する要求を再試行する前に待機する間隔（ミリ秒単位）を指定します。</p> <p>デフォルト値は 1000 です。</p>

注: 環境変数を定義または変更したら、データベース取り込みエージェントを再起動して、変更を有効にします。

第 6 章

Discovery Agent アプリケーション

Discovery Agent アプリケーションは、データ統合（CDI 実行エンジン）またはクラスタ（CDI-E 実行エンジン）のステージングされたプロファイル結果を読み取り、Secure Agent を介してプロファイル結果をメタデータコマンドセンターにアップロードします。

いくつかのサービスプロパティを設定して、Discovery Agent アプリケーションサービスのパフォーマンスを最適化できます。

Discovery Agent アプリケーションのプロパティ

Discovery Agent アプリケーションサービスの動作を変更する、または最適化するには、Secure Agent の編集時にプロパティを **【システム構成の詳細】** セクションで設定します。

次の画像は、Discovery Agent アプリケーションサービスのプロパティを示しています。

System Configuration Details

[Reset All](#)

Service: Discovery Agent Application ▼

Type: All Types ▼

Type	Name
APP_CFG	discovery_agent_app_maxParallelTasks
APP_CFG	discovery_agent_app_agentMaxRetryAttempts
APP_CFG	discovery_agent_app_agentInitialBackoffInterval
APP_CFG	discovery_agent_app_agentMaxBackoffInterval
APP_CFG	discovery_agent_app_agentShutdownWaitTimeMillis
APP_CFG	discovery_agent_app_profileResultRetention
APP_CFG	discovery_agent_app_JVM_ARGS
PLUGIN_CFG	plugin_JVM_ARGS
APP_LOG4J	name

以下の Discovery Agent アプリケーションサービスのプロパティを設定できます。

タイプ	名前	説明	サンプル値	デフォルト値
APP_CFG	discovery_agent_app_maxParallelTasks	結果アップロードタスクのために生成されるスレッドの最大数。	3	4
APP_CFG	discovery_agent_app_agentMaxRetryAttempts	結果アップロードの最大再試行回数。	6	5
APP_CFG	discovery_agent_app_agentInitialBackoffInterval	-	50	100
APP_CFG	discovery_agent_app_agentMaxBackoffInterval	-	35000	35000

タイプ	名前	説明	サンプル値	デフォルト値
APP_CFG	discovery_agent_app_agentShutdownWaitTimeMillis	-	60000	60000
APP_CFG	discovery_agent_app_profileResultRetention	アップロード後にプロファイル結果を保持または削除するオプション。 デフォルトの動作では、結果が削除されます。	true	false
APP_CFG	discovery_agent_app_JVM_ARGS	該当なし	-	-
PLUGIN_CFG	plugin_JVM_ARGS	デバッグパラメータを追加する、またはJVMメモリ構成パラメータを変更するオプション。	agentlib:jdwp=transport=dt_socket,address=8083,server=y,suspend=y	-
APP_LOG4J	name	すべてのSecureAgentアプリケーションの共通プロパティ。	AppLogPropertiesConfig	AppLogPropertiesConfigXmx2048m
APP_LOG4J	rootLogger_level	すべてのSecureAgentアプリケーションの共通プロパティ。		info
APP_LOG4J	rootLogger_appenderRefs	すべてのSecureAgentアプリケーションの共通プロパティ。	applog	applog
APP_LOG4J	rootLogger_appenderRefs_applog_ref	すべてのSecureAgentアプリケーションの共通プロパティ。	LOGFILE	LOGFILE
APP_LOG4J	appenders	すべてのSecureAgentアプリケーションの共通プロパティ。	applog	applog

タイプ	名前	説明	サンプル値	デフォルト値
APP_LOG4J	appender_app_log_type	すべてのSecureAgentアプリケーションの共通プロパティ。	RollingFile	RollingFile
APP_LOG4J	appender_app_log_name	すべてのSecureAgentアプリケーションの共通プロパティ。	LOGFILE	LOGFILE
APP_LOG4J	appender_app_log_filePattern	すべてのSecureAgentアプリケーションの共通プロパティ。	%d{MM-dd-yy-HH-mm-ss}-%i	%d{MM-dd-yy-HH-mm-ss}-%i
APP_LOG4J	appender_app_log_layout_type	すべてのSecureAgentアプリケーションの共通プロパティ。	PatternLayout	PatternLayout
APP_LOG4J	appender_app_log_layout_pattern	すべてのSecureAgentアプリケーションの共通プロパティ。	%d %d{z} %p [%c] - %m%n	%d %d{z} %p [%c] - %m%n
APP_LOG4J	appender_app_log_policies_type	すべてのSecureAgentアプリケーションの共通プロパティ。	ポリシー	ポリシー
APP_LOG4J	appender_app_log_policies_size_type	すべてのSecureAgentアプリケーションの共通プロパティ。	SizeBasedTriggeringPolicy	SizeBasedTriggeringPolicy
APP_LOG4J	appender_app_log_policies_size_size	すべてのSecureAgentアプリケーションの共通プロパティ。	5 MB	10 MB
APP_LOG4J	appender_app_log_strategy_type	すべてのSecureAgentアプリケーションの共通プロパティ。	DefaultRolloverStrategy	DefaultRolloverStrategy

タイプ	名前	説明	サンプル値	デフォルト値
APP_LOG4J	appender_app log_strategy_ max	すべての SecureAgent アプリケーションの共通プロパティ。	5	5
APP_LOGBACK	logback_log_file_pattern	すべての SecureAgent アプリケーションの共通プロパティ。	{yyyy-ww}	{yyyy-ww}
APP_LOGBACK	logback_log_max_file_size	すべての SecureAgent アプリケーションの共通プロパティ。	5 MB	10 MB
APP_LOGBACK	logback_log_max_history	すべての SecureAgent アプリケーションの共通プロパティ。	20	10
APP_LOGBACK	logback_root_level	すべての SecureAgent アプリケーションの共通プロパティ。	デバッグ	情報

第 7 章

データ統合サーバー

データ統合サーバーは、マッピング、タスク、およびタスクフローインスタンスなどのデータ統合ジョブを実行する Secure Agent サービスです。

詳細クラスタが、詳細モードのマッピングのデータロジックを処理する場合、データ統合サーバーは詳細クラスタのサブタスクをエラスティックサーバーに委ねます。

いくつかのサービスプロパティを設定して、データ統合サーバーのパフォーマンスを最適化できます。例えば、ネットワークの回復機能設定または Secure Agent の接続タイムアウト期間を変更できます。サービスプロパティは、Secure Agent の編集時に変更できます。

データ統合サーバーの回復機能

ネットワークの一時的な問題が発生している際、Secure Agent が接続の再確立を試みている間、データ統合タスクを続行できます。データ統合サーバーのネットワークの回復機能プロパティを設定できます。

Secure Agent が接続の再確立を試みる方法は、次のデータ統合サーバーのプロパティで決定されます。

NetworkTimeoutPeriod

Secure Agent で Informatica Intelligent Cloud Services との通信の再確立を試行する時間の長さを決定します。期間の終わりに通信が確立されていない場合、実行されていた進行中のデータ統合タスクは停止します。デフォルト値は 300 秒です。

NetworkRetryInterval

Secure Agent が指定されたタイムアウト期間内に Informatica Intelligent Cloud Services への接続を試行する頻度を決定します。デフォルト値は 5 秒です。

例えば、デフォルト設定の場合、ネットワークが停止すると、Secure Agent は Informatica Intelligent Cloud Services との通信の再確立を 300 秒間試行します。その 300 秒の期間中、Secure Agent は 5 秒ごとに Informatica Intelligent Cloud Services への接続を試行します。300 秒の期間内に Secure Agent が通信を再確立すれば、進行中のデータ統合タスクは影響を受けません。Secure Agent は、300 秒の期間内に通信を再確立できない場合、進行中のデータ統合タスクを停止します。

データ統合サーバーのプロパティ

データ統合サーバーの動作を変更または最適化するには、データ統合サーバーのプロパティを設定します。データ統合サーバーのプロパティは、Secure Agent の編集時に **【システム構成の詳細】** 領域で設定します。

次の画像は、データ統合サーバーのプロパティを示しています。

▼ System Configuration Details		
Service:	Data Integration Server ▼	
Type:	All Types ▼	
Type	Name	Value
Tomcat	NetworkTimeoutPeriod	300
Tomcat	NetworkRetryInterval	5
Tomcat JRE	INFA_SSL	
Tomcat JRE	INFA_MEMORY	'-Xms32m -Xmx512m -XX:MaxPermSize=128m'
Tomcat JRE	JRE_OPTS	'-Xrs'
Tomcat JRE	JAVA_LIBS	
Tomcat Log4j	log4j_rootLogger	'INFO, tomcatLog'
Tomcat Log4j	log4j_appender_tomcatLog	'org.apache.log4j.FileAppender'
Tomcat Log4j	log4j_appender_tomcatLog_layout	'org.apache.log4j.PatternLayout'
Tomcat Log4j	log4j_appender_tomcatLog_layout_ConversionPattern	'%d %d{z} %p [%c] - %m%n'

設定可能なデータ統合サーバーのプロパティを次に示します。

タイプ	名前	説明
Tomcat	NetworkTimeoutPeriod	Secure Agent が Informatica Intelligent Cloud Services との通信の再確立を試行するまでの時間（秒）。デフォルトは 300 です。
Tomcat	NetworkRetryInterval	Secure Agent が、指定したタイムアウト期間内に Informatica Intelligent Cloud Services への接続を試行する頻度（秒）。デフォルトは 5 です。
Tomcat JRE	JRE_OPTS	Apache Tomcat プロセスの JRE VM オプション。
Tomcat JRE	INFA_MEMORY	Apache Tomcat プロセスの仮想マシンメモリに対して設定される JRE VM オプション。
DTM	AgentConnectionTimeout	Secure Agent 通信で、タイムアウトするまでに待機を要求する秒数。デフォルトは 5 です。

タイプ	名前	説明
DTM	JVMOption1 - JVMOption5	<p>最大および最小の JVM ヒープサイズ、インテリジェント構造検出の最大レコードサイズ、特定のコネクタのプロキシ設定などの、データ統合サーバーの詳細プロパティを設定する JVM オプション。例えば、最大 JVM ヒープサイズをデフォルト値の 512 MB から 2048 MB に変更するには、JVMOption1 を '-Xmx2048m' に設定します。</p> <p>デフォルトでは、JVMOption1 - JVMOption5 を使用して、最大 5 つの詳細プロパティを設定できます。追加のプロパティを設定するには、JVMOption6 や JVMOption7 などの名前を付けた、データ統合サーバー用のカスタム DTM プロパティを追加します。オプション番号が連続していて、番号を飛ばしていないことを確認してください。</p> <p>設定できる JVM オプションの詳細については、データ統合のヘルプ、適切なコネクタのヘルプ、または Informatica Network の Knowledge Base を参照してください。</p>
<p>注: Informatica グローバルカスタマサポートから指示された場合を除き、データ統合サーバーの他のプロパティ値は変更しないでください。</p>		

第 8 章

エラスティックサーバー









エラスティックサーバーは、詳細クラスタとクラスタで実行されるジョブを管理する Secure Agent サービスです。

エラスティックサーバーがログファイルに書き込む詳細のレベルを指定するために、サービスプロパティを設定できます。クラスタオペレータロールと Secure Agent ロールをセットアップするために設定する必要のあるサービスプロパティもあります。詳細については、[詳細クラスタの説明](#)を参照してください。

エラスティックサーバーのプロパティ

エラスティックサーバーの動作を変更するには、Secure Agent の編集時に **【システム構成の詳細】** 領域でエラスティックサーバーのプロパティを設定します。

次の図は、エラスティックサーバーのプロパティを示しています。

System Configuration Details Reset All				
Service:	Elastic Server ▼			
Type:	All Types ▼			
Type	Name	Value	Sensitive	
LOG4J_CFG	log4j_app_log_level	'INFO'	<input type="checkbox"/>	 
AWS_CFG	agent_role_external_id_key		<input type="checkbox"/>	 
AWS_CFG	privileged_role_arn_key	arn:aws:iam::<account-id>:role/cluster_operator_role	<input type="checkbox"/>	 
AWS_CFG	role_session_duration_secs_key		<input type="checkbox"/>	 
AWS_CFG	aws_regional_endpoint_enabled	'false'	<input type="checkbox"/>	 
AZURE_CFG	azure_agent_role_identity_client_id		<input type="checkbox"/>	 

設定できるエラスティックサーバーのプロパティを次に示します。

タイプ	名前	説明
LOG4J_CFG	log4j_app_log_level	<p>エラスティックサーバーがログファイルに書き込む詳細のレベル。「INFO」などの文字列としてログレベルを入力します。</p> <p>ログレベルを大きくすると、エラスティックサーバーがログファイルに書き込むメッセージに、より優先度の高いログレベルのメッセージが含まれます。例えば、ログレベルが INFO の場合、ログには FATAL、ERROR、WARNING、および INFO コードのメッセージが記録されます。</p> <p>有効な値は次のとおりです。</p> <ol style="list-style-type: none"> 1. FATAL。サービスがシャットダウンする、または利用不可能になる修復不能なシステム障害が含まれます。 2. ERROR。接続の失敗、メタデータの保存または取得の失敗、サービスのエラーが含まれます。 3. WARNING。修復可能なシステム障害または警告が含まれます。 4. INFO。システムおよびサービスの変更にに関するメッセージが含まれます。 5. TRACE。ユーザー要求の失敗がログとして記録されます。 6. DEBUG。ユーザー要求がログとして記録されます。
AWS_CFG	agent_role_external_id_key	<p>Secure Agent がクラスタオペレータロールを使用する場合に Secure Agent で指定する外部 ID。クラスタオペレータロールの信頼関係で外部 ID を設定する場合に必要です。</p> <p>このプロパティは、AWS 環境でのみ有効です。</p>
AWS_CFG	privileged_role_arn_key	<p>クラスタオペレータロールの ARN。</p> <p>AWS 環境で個別のクラスタオペレータロールと Secure Agent ロールを設定する場合に必要です。</p> <p>このプロパティは、AWS 環境でのみ有効です。</p>
AWS_CFG	role_session_duration_secs_key	<p>AWS AssumeRole API のセッション時間（秒単位）。デフォルトのセッション時間は 1,800 秒（30 分）です。</p> <p>クラスタオペレータロールに設定されている最大 CLI/API セッション期間をオーバーライドします。エラスティックサーバーに設定されているセッション期間がクラスタオペレータロールのセッション期間よりも長い場合、Secure Agent がクラスタオペレータロールを使用できない場合があります。</p> <p>このプロパティは、AWS 環境でのみ有効です。</p>
AZURE_CFG	azure_agent_role_identity_client_id	<p>マネージド ID agent_identity のクライアント ID。agent_identity がユーザー割り当てのマネージド ID であり、Secure Agent マシンに少なくとも 1 つの他のマネージド ID がある場合に必要です。</p> <p>このプロパティは、Azure 環境でのみ有効です。</p>

第 9 章

ファイル統合サービス

ファイル統合サービスを使用して、組織とリモートファイルサーバーの間でファイルを転送します。

ファイル統合サービスは、エージェントが AS2 などの高度なファイル転送プロトコルの実行に使用する Secure Agent のサービスです。

組織でリモートパートナーからファイルを受信できるようにする前に、ファイルサーバーを設定しておく必要があります。管理者の「ファイルサーバー」ページで、ファイル統合サービスに関連付けられる組織のファイルサーバーを設定します。設定には、ファイルサーバーの詳細、暗号化の方法、および許可されるファイルタイプなどのプロパティが含まれます。

ファイル統合サービスを停止または開始するには、サービスを使用するファイルサーバーを停止または開始します。

ファイルサーバーの設定については、「ファイルサーバー」を参照してください。

ファイル統合サービスを使用するには、組織が適切なライセンスを持っている必要があります。ファイル統合サービスを設定するには、管理者ロールが割り当てられている必要があります。

第 10 章

GitRepoConnectApp

組織がオンプレミスのソース管理リポジトリを使用している場合、GitRepoConnectApp サービスは Informatica Intelligent Cloud Services およびソース管理リポジトリ間の通信を管理します。

Secure Agent は、Secure Agent マシンへのリモートソース管理リポジトリのローカルコピーの作成時に GitRepoConnectApp サービスを使用します。また、このサービスを使用して、リモートリポジトリからソース管理操作に関する情報を取得します。

ローカルリポジトリのベースディレクトリ

ソース管理リポジトリがオンプレミスの場合、Secure Agent は Informatica Intelligent Cloud Services アセットを格納するリポジトリブランチのローカルコピーを作成します。Secure Agent マシンでローカルリポジトリの場所を設定できます。

デフォルトでは、Secure Agent は次のディレクトリにローカルリポジトリを作成します。

```
<Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/<ベースディレクトリ>/<クライアント URL>/<組織 ID>/<ブランチ>/<リモートリポジトリ名>
```

このファイルパスでは、ベースディレクトリは GitRepoConnectApp サービスの **[git_local_repository_path]** プロパティによって管理されます。

デフォルトでは、**git_local_repository_path** は `../data/git_repository/` に設定されています。そのため、Secure Agent は次のディレクトリにローカル Git リポジトリを作成します。

```
<Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/data/git_repository/<クライアント URL>/<組織 ID>/<ブランチ>/<リモートリポジトリ名>
```

[git_local_repository_path] プロパティを編集することで、ベースディレクトリを変更できます。例えば、このプロパティを `../MYREPO/PROD` に設定した場合、Secure Agent は、次のディレクトリにローカル Git リポジトリを作成します。

```
<Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/MYREPO/PROD/<クライアント URL>/<組織 ID>/<ブランチ>/<リモートリポジトリ名>
```

バックスラッシュ文字 (\) を含むベースディレクトリを指定するには、別のバックスラッシュ文字でエスケープします。

[git_local_repository_path] プロパティを設定する場合は、次のガイドラインを使用してください。

- 親ディレクトリ (..) を省略した場合このプロパティを設定すると、Secure Agent は GitRepoConnectApp サービスのバージョンにサブディレクトリを作成します。リポジトリのローカルコピーは、次のディレクトリに保存されます。

<Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/<GitRepoConnectApp バージョン>/<ベースディレクトリ>/<クライアント URL>/<組織 ID>/<ブランチ>/<リモートリポジトリ名>

この場合、Secure Agent は GitRepoConnectApp サービスが更新されるたびに新しいローカルリポジトリディレクトリを作成するため、Secure Agent マシンのディスク領域が大量に消費される可能性があります。

- このプロパティを設定する場合は、ローカルリポジトリディレクトリが複数のエージェントによって共有されないようにしてください。それぞれの Secure Agent マシンには、リポジトリの独自のローカルコピーが必要です。

GitRepoConnectApp のプロパティ

GitRepoConnectApp サービスの動作を変更または最適化するには、サービスのプロパティを設定します。Secure Agent の編集時に、[システム構成の詳細] 領域でサービスプロパティを設定します。

以下の図に、GitRepoConnectApp のプロパティを示します。

▼ System Configuration Details		
Service:	GitRepoConnectApp ▼	
Type:	All Types ▼	
Type	Name	Value
LOG4J	rootLogger	'INFO'
GIT_REPO_CONNECT_APP_CONF	host	'localhost'
GIT_REPO_CONNECT_APP_CONF	address	'127.0.0.1'
GIT_REPO_CONNECT_APP_CONF	git_local_repository_path	'../data/git_repository/'
GIT_REPO_CONNECT_APP_CONF	JVM_MIN_MEMORY	'32m'
GIT_REPO_CONNECT_APP_CONF	JVM_MAX_MEMORY	'256m'

以下のサービスのプロパティを設定できます。

タイプ	名前	説明
GIT_REPO_CONNECT_APP_CONF	git_local_repository_path	Secure Agent マシンのローカル Git リポジトリのベースディレクトリ。 ベースディレクトリは、次のディレクトリに作成されます。 <Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/ デフォルトは../data/git_repository/です。そのため、Secure Agent は次のディレクトリにローカル Git リポジトリを作成します。 <Secure Agent インストールディレクトリ>/apps/GitRepoConnectApp/git_repository/<クライアント URL>/<組織 ID>/<ブランチ>/<リモートリポジトリ名>
GIT_REPO_CONNECT_APP_CONF	JVM_MIN_MEMORY	サービスの開始時に GitRepoConnectApp サービスに割り当てられるメモリの量。 デフォルトは 32 MB です。
GIT_REPO_CONNECT_APP_CONF	JVM_MAX_MEMORY	GitRepoConnectApp サービスに割り当てられる最大メモリ。 デフォルトは 256 MB です。
注: Informatica グローバルカスタマサポートから指示された場合を除き、このプロパティ値は変更しないでください。		

第 11 章

一括取り込み（ファイル）

Secure Agent グループが使用する一括取り込みファイルの動作を変更または最適化するには、Administrator のランタイム環境に一括取り込みプロパティを設定します。

以下のプロパティを設定する事ができます。

タイプ	名前	説明
AGENT_RUNTIME_SETTINGS	file-listener-snapshot-dir	新しいファイルリスナコンポーネントのスナップショットが追加されるディレクトリ。次のディレクトリパスを追加できます。 <ul style="list-style-type: none">- MassIngestionRuntime ディレクトリに対する相対パス。例: ../data/monitor。- 絶対パス。以下に例を示します。 <Secure agent installation directory>/apps/MassIngestionRuntime/data/monitor <i>Secure agent installation directory</i> には Secure Agent がインストールされているディレクトリの名前が入ります。 注: グループに複数の Secure Agent が存在する場合は、すべてのエージェントで共有されるスナップショットディレクトリを使用します。
AGENT_RUNTIME_SETTINGS	mi-task-workspace-dir	ファイル取り込みタスクがファイルをターゲットに転送するときに中間ステージング領域として使用するエージェント内のディレクトリです。エージェント内のカスタムの場所のディレクトリ。このパスは、共有場所、マウントされた場所、およびエージェント内のデフォルトの場所とは異なる場所にすることができます。
AGENT_RUNTIME_SETTINGS	mi-task-quarantine-dir	ウイルススキャンの実行時に検出された感染ファイルをファイル取り込みタスクが保存するディレクトリ。ディレクトリはエージェント内のカスタムの場所です。このパスは、共有場所、マウントされた場所、およびエージェント内のデフォルトの場所とは異なる場所にすることができます。 例: userdata\quarantine 注: 検疫ディレクトリを自動的にクリーンアップするには、検疫場所のエージェントプロパティを /tmp/informatica/fmi/quarantine のようなシステム一時ファイルの場所に設定します。
AGENT_RUNTIME_SETTINGS	file-listener-max-pool-size	ファイルリスナを実行するスレッドの最大数。デフォルトは 20 です。

タイプ	名前	説明
AGENT_RUNTIME_SETTINGS	file-listener-core-pool-size	スレッドの合計数。 デフォルトは 20 です。
AGENT_RUNTIME_SETTINGS	fmi-task-max-pool-size	ファイル取り込みタスクを実行するスレッドの最大数。 デフォルトは 50 です。
AGENT_RUNTIME_SETTINGS	fmi-task-core-pool-size	スレッドの初期数または最小数。 デフォルトは 20 です。
AGENT_RUNTIME_SETTINGS	ftp-receive-socket-buffer-size	FTP インバウンドパケットのバッファサイズ。 デフォルトは 16 バイトです。
AGENT_RUNTIME_SETTINGS	ftp-send-socket-buffer-size	FTP アウトバウンドパケットのバッファサイズ。 デフォルトは 16 バイトです。
AGENT_RUNTIME_SETTINGS	http-client-timeout	Informatica Intelligent Cloud Services へのエージェントの要求のタイムアウト時間 (秒単位)。 デフォルトは 30 秒です。
PGP_SETTINGS	public-keyring-path	パブリックキーリングを保存するディレクトリ。次のディレクトリパスを追加できます。 <ul style="list-style-type: none"> - 一括取り込みがインストールされるディレクトリに対する相対パス。以下に例を示します。 <code>../data/pubring.pkr</code> <code>pubring.pkr</code> にはパブリックキーリングを保存するファイルの名前が入ります。 - 絶対パス。以下に例を示します。 <code><Secure agent installation directory>/apps/MassIngestionRuntime/data/pubring.pkr</code> <code>pubring.pkr</code> にはパブリックキーリングを保存するファイルの名前が、<code>Secure agent installation directory</code> にはエージェントがインストールされているディレクトリの名前が入ります。
PGP_SETTINGS	secret-keyring-path	シークレットキーリングを保存するディレクトリ。次のディレクトリパスを追加できます。 <ul style="list-style-type: none"> - 一括取り込みがインストールされるディレクトリに対する相対パス。以下に例を示します。 <code>../data/secring.pkr</code> <code>secring.pkr</code> にはシークレットキーリングを保存するファイルの名前が入ります。 - 絶対パス。以下に例を示します。 <code><Secure agent installation directory>/apps/MassIngestionRuntime/data/secring.pkr</code> <code>secring.pkr</code> にはシークレットキーリングを保存するファイルの名前が、<code>Secure Agent installation directory</code> にはエージェントがインストールされているディレクトリの名前が入ります。

タイプ	名前	説明
JVM_SETTINGS	app-heap-size	一括取り込みファイルアプリケーションの最小および最大ヒープサイズ。 デフォルトは-Xms256m -Xmx2048m です。
JVM_SETTINGS	lcm-heap-size	ライフサイクル管理スクリプトの最小および最大ヒープサイズ。 デフォルトは-Xms32m -Xmx128m です。

Secure Agent を編集する場合は、**[カスタム構成の詳細]** 領域で次のプロパティを設定できます。

タイプ	名前	説明
AGENT_RUNTIME_SETTINGS	ComplexFileDisableWriteChecksum	crc ファイルを無視するには、値を True に設定します。ジョブは、Hadoop ファイル V2 をソースとし、Snowflake Cloud Data Warehouse V2 をターゲットとして正常に実行されます。

第 12 章

メタデータ基盤アプリケーション

メタデータ基盤アプリケーションサービスを使用すると、組織内で設定されているソースシステムからメタデータを抽出し、抽出したメタデータを Secure Agent を介してメタデータコマンドセンターにアップロードできます。

いくつかのサービスプロパティを設定して、メタデータ基盤アプリケーションサービスのパフォーマンスを最適化できます。

メタデータ基盤アプリケーションのプロパティ

メタデータ基盤アプリケーションサービスの動作を変更する、または最適化するには、Secure Agent の編集時にプロパティを **【システム構成の詳細】** セクションで設定します。

次の画像は、メタデータ基盤アプリケーションサービスのプロパティを示しています。

System Configuration Details Reset All

Service:

Metadata Foundation Application ▼

Type:

All Types ▼

Type	Name
APP_CFG	mfa_maxParallelTasks
APP_CFG	mfa_agentMaxRetryAttempts
APP_CFG	mfa_agentInitialBackoffInterval
APP_CFG	mfa_agentMaxBackoffInterval
APP_CFG	mfa_agentShutdownWaitTimeMillis
APP_CFG	mfa_JVM_ARGS
PLUGIN_CFG	plugin_JVM_ARGS

以下のメタデータ基盤アプリケーションサービスのプロパティを設定できます。

タイプ	名前	説明	サンプル値	デフォルト値
APP_CFG	mfa_maxParallelTasks	Secure Agentで同時実行中のタスクの数。	5	4
APP_CFG	mfa_agentMaxRetryAttempts	-	3	5
APP_CFG	mfa_agentInitialBackoffInterval	-	500	100
APP_CFG	mfa_agentMaxBackoffInterval	-	20000	35000
APP_CFG	mfa_agentShutdownWaitTimeMillis	-	30000	60000
APP_CFG	mfa_JVM_ARGS	デバッグパラメータを追加する、またはJVM メモリ構成パラメータを変更するオプション。	agentlib:jdwp=transport=dt_socket,address=8084,server=y,suspend=y	-
PLUGIN_CFG	plugin_JVM_ARGS	デバッグパラメータを追加する、またはJVM メモリ構成パラメータを変更するオプション。	agentlib:jdwp=transport=dt_socket,address=8083,server=y,suspend=y	-
TRANSFER_SVC_CFG	transfer_svc_JVM_ARGS	デバッグパラメータを追加する、またはJVM メモリ構成パラメータを変更するオプション。	agentlib:jdwp=transport=dt_socket,address=8085,server=y,suspend=y	-
TRANSFER_SVC_CFG	transfer_svc_batchSize	バッチで処理できるコンテンツファイルの数。 このオプションを使用して、アップロードサービスを使用してアップロードされるCSV ファイルの生成を最適化します。	4	1

タイプ	名前	説明	サンプル値	デフォルト値
TRANSFER_SVC_CFG	transfer_svc_stagingMaxRetry	コンテンツのステージング中に失敗した場合の再試行回数。	5	3
TRANSFER_SVC_CFG	transfer_svc_parallelTaskExecutorsSize	同時実行中のタスクの数。	5	4
TRANSFER_SVC_CFG	transfer_svc_contentBatchExecutorsSize	アップロードサービスを使用した同時コンテンツアップロードの数。	5	4
TRANSFER_SVC_CFG	transfer_svc_stagingBatchExecutorsSize	同時コンテンツステージングタスクの数。	5	4
TRANSFER_SVC_CFG	transfer_svc_contentWorkersBean	ステージング、取り込み、およびログ転送を行うサービス Bean を転送します。 ステージングを無効にするには、取り込みとログ転送のみを指定します。	ingestion,log	ingestion,staging,log
TRANSFER_SVC_CFG	transfer_svc_ingestionMaxRetry	失敗した場合のアップロードと一括取り込みの再試行回数。	5	3
TRANSFER_SVC_CFG	transfer_svc_contentMaxRetentionTimeInMin	アップロード後にメタデータの抽出結果を保持または削除するオプション。 デフォルトの動作では、結果が削除されます。	600	0

タイプ	名前	説明	サンプル値	デフォルト値
TRANSFER_SVC_CFG	transfer_svc_postDriverCompletionMaxWaitTimeInSec	長時間実行中または応答しない取り込みタスクが終了した後の秒数。 デフォルトの動作では、長時間実行中または応答しない取り込みタスクは終了しません。	10	-1
APP_LOG4J	name	すべてのSecureAgentアプリケーションの共通プロパティ。	AppLogPropertiesConfig	AppLogPropertiesConfig Xmx2048m
APP_LOG4J	rootLogger_level	すべてのSecureAgentアプリケーションの共通プロパティ。	warn	info
APP_LOG4J	rootLogger_appenderRefs	すべてのSecureAgentアプリケーションの共通プロパティ。	applog	applog
APP_LOG4J	rootLogger_appenderRefs_applog_ref	すべてのSecureAgentアプリケーションの共通プロパティ。	LOGFILE	LOGFILE
APP_LOG4J	appenders	すべてのSecureAgentアプリケーションの共通プロパティ。	applog	applog
APP_LOG4J	appender_applog_type	すべてのSecureAgentアプリケーションの共通プロパティ。	RollingFile	RollingFile
APP_LOG4J	appender_applog_name	すべてのSecureAgentアプリケーションの共通プロパティ。	LOGFILE	LOGFILE

タイプ	名前	説明	サンプル値	デフォルト値
APP_LOG4J	appender_app_log_filePattern	すべてのSecureAgentアプリケーションの共通プロパティ。	%d{MM-dd-yy-HH-mm-ss}-%i	%d{MM-dd-yy-HH-mm-ss}-%i
APP_LOG4J	appender_app_log_layout_type	すべてのSecureAgentアプリケーションの共通プロパティ。	PatternLayout	PatternLayout
APP_LOG4J	appender_app_log_layout_pattern	すべてのSecureAgentアプリケーションの共通プロパティ。	%d %d{z} %p [%c] - %m%n	%d %d{z} %p [%c] - %m%n
APP_LOG4J	appender_app_log_policies_type	すべてのSecureAgentアプリケーションの共通プロパティ。	ポリシー	ポリシー
APP_LOG4J	appender_app_log_policies_size_type	すべてのSecureAgentアプリケーションの共通プロパティ。	SizeBasedTriggeringPolicy	SizeBasedTriggeringPolicy
APP_LOG4J	appender_app_log_policies_size_size	すべてのSecureAgentアプリケーションの共通プロパティ。	5 MB	10 MB
APP_LOG4J	appender_app_log_strategy_type	すべてのSecureAgentアプリケーションの共通プロパティ。	DefaultRolloverStrategy	DefaultRolloverStrategy
APP_LOG4J	appender_app_log_strategy_max	すべてのSecureAgentアプリケーションの共通プロパティ。	5	5
APP_LOGBACK	logback_log_file_pattern	すべてのSecureAgentアプリケーションの共通プロパティ。	{yyyy-ww}	{yyyy-ww}

タイプ	名前	説明	サンプル値	デフォルト値
APP_LOGBACK	logback_log_max_file_size	すべてのSecureAgentアプリケーションの共通プロパティ。	5 MB	10 MB
APP_LOGBACK	logback_log_max_history	すべてのSecureAgentアプリケーションの共通プロパティ。	20	10
APP_LOGBACK	logback_root_level	すべてのSecureAgentアプリケーションの共通プロパティ。	デバッグ	情報

第 13 章

プロセスサーバー

プロセスサーバーとは、アプリケーションの統合のプロセス、コネクタ、および接続を実行する Secure Agent サービスです。

Secure Agent にアプリケーション統合のアセットをデプロイしたら、プロセスサーバーにもデプロイします。アセットを実行すると、プロセスサーバーによって実行されます。

PostgreSQL データベースには Secure Agent のプロセスサーバーサービスが付属しており、プロセスサーバーが収集および生成したメタデータが格納されます。

システムの次の場所にある PostgreSQL ディレクトリを検索します。

```
<Secure Agent installation directory>\apps\process-engine\data\PostgreSQL
```

プロセスサーバーのプロパティ

プロセスサーバーの動作を変更または最適化するには、プロセスサーバーのプロパティを設定します。サーバー、Secure Agent グループ、Java 仮想マシン、コネクタ、データベース、およびログプロパティを設定できます。

次の図に、一部のプロセスサーバーのプロパティを示しています。

Type	Name	Value
server	host-name	'nki7.informatica.com'
server	shutdown-port	7005
server	key-alias	'localhost'
server	key-store	'./conf/keystore'
server	key-store-password	'password'
server	trust-store	'./conf/cacerts'
server	trust-store-password	'changeit'
server	ldap-enabled-realm	false
server	ldap-properties	- key: connectionURL value: ldap://\${host.name}:10389 - key: connectionName value: uid=admin,ou=system - key: connectionPassword value: \${pe.ldap.password} - key: authentication value: simple - key: userBase value: ou=people,DC=\${host.name},DC=informatica,DC=com - key: userSearch value: (uid=0) - key: roleBase value: ou=groups,DC=\${host.name},DC=informatica,DC=com - key: roleName value: cn - key: roleSearch value: (uniqueMember=0)

server	host-name	'localhost'
server	shutdown-port	7005
server	key-alias	'localhost'
server	key-store	'../conf/ae.keystore'
server	key-store-password	'password'
server	trust-store	'../conf/ae.cacerts'
server	trust-store-password	'changeit'
server	ldap-enabled-realm	false
server	ldap-properties	- key: connectionURL value: ldap://\${host.name}:10389 - key: connectionName value: uid=admin,ou=system - key: connectionPassword value: \${ps.ldap.password} - key: authentication value: simple - key: userBase value: ou=people,DC=\${host.name},DC=informatica,DC=com - key: userSearch value: (uid={0}) - key: roleBase value: ou=groups,DC=\${host.name},DC=informatica,DC=com - key: roleName value: cn - key: roleSearch value: (uniqueMember={0})
server	ssl-enabled-protocols	'TLSv1.2'
server	ephemeral-DH-key-size	2048
server	use-secure-ciphers-only	true

注: 【システム構成の詳細】 の【カスタム構成の詳細】を編集しないでください。変更内容は反映されません。

以下のサーバープロパティを設定できます。

名前	通信方式	説明
host-name	Secure Agent チャンネル	プロセスエンジンサーバーのホスト名。
shutdown-port	Secure Agent チャンネル	プロセスサーバー Tomcat のシャットダウンポート。
key-alias	HTTPS	HTTPS 通信のセキュリティキーが含まれるキーストアレコードの識別子。
key-store	HTTPS	<p>アプリケーションの統合が HTTPS 通信に使用するキーストアファイルのパスとファイル名。</p> <p>Secure Agent をインストールすると、次のデフォルトの場所にキーストアがインストールされます。</p> <p><Secure Agent インストールディレクトリ>/apps/process-engine/conf/ae.keystore</p> <p>また、相対パスを入力することもできます。例えば、現在の作業ディレクトリが Secure Agent インストールディレクトリである場合、ae.keystore ファイルを指すように次の値を入力します。</p> <p>../conf/ae.keystore</p> <p>注: ファイルパスにはスラッシュのみを含めることができます (/)。</p>
key-store-password	HTTPS	キーストアのパスワード。デフォルトパスワードは password です。

名前	通信方式	説明
trust-store	HTTPS	<p>アプリケーションの統合が HTTPS 通信に使用するトラストストアファイルのパスとファイル名。</p> <p>Secure Agent をインストールすると、デフォルトの場所にトラストストアがインストールされます。</p> <p><Secure Agent インストールディレクトリ>/apps/process-engine/conf/ae.cacerts</p> <p>また、相対パスを入力することもできます。例えば、現在の作業ディレクトリが Secure Agent インストールディレクトリである場合、ae.cacerts ファイルを指すように次の値を入力します。</p> <p>../conf/ae.cacerts</p> <p>注: ファイルパスにはスラッシュのみを含めることができます (/)。</p> <p>サービスエンドポイント認証用の公開証明書をインポートする場合は、次の場所に配置します。</p> <p><Secure Agent インストールディレクトリ>/apps/process-engine/conf/certs</p>
trust-store-password	HTTPS	トラストストアのパスワード。デフォルトのパスワードは changeit です。パスワードは変更できます。
ldap-enabled-realm	HTTP/HTTPS	認証に LDAP プロバイダを使用する場合は、このプロパティを [True] に設定します。クラスタ化された Secure Agent を使用する場合は、LDAP プロバイダを認証の一括管理用として使用します。
ldap-properties	HTTP/HTTPS	<p>設定する必要がある LDAP プロパティ。LDAP プロバイダに合うように既存のプロパティを編集します。</p> <p>注: LDAP パスワードは画面に表示されません。\$(pe.ldap.password)の値は環境変数 PE_LDAP_PASSWORD から取得されます。</p>
ssl-enabled-protocols	HTTPS	使用する TLS プロトコル。デフォルトのプロトコルは最も安全なプロトコルである TLSv1.2 です。互換性の問題が発生した場合のみ、この値を TLSv1.0 または TLSv1.1 などの古いバージョンに変更します。
ephemeral-DH-key-size	HTTPS	安全なアルゴリズムのキーの長さ。デフォルト値は 2048 です。互換性の問題が発生した場合のみ、この値を変更します。
use-secure-ciphers-only	HTTPS	エンドポイントの呼び出し時に使用する暗号セットを安全な暗号のみに制限します。デフォルト値は [True] です。互換性の問題が発生した場合のみ、この値を [False] に変更します。

次の Secure Agent グループ (UI では「クラスタ」) のプロパティを設定できます。

名前	通信方式	説明
name	HTTP/HTTPS	Secure Agent グループの名前。
primary-node	HTTP/HTTPS	Secure Agent をマスタエージェントにする場合は、このプロパティを [True] に設定します。マスタエージェントを選択する場合は、Secure Agent クラスタを作成します。クラスタでは、すべての Secure Agent がマスタ Secure Agent の PostgreSQL データベースを共有します。
load-balance-url	HTTP/HTTPS	<p>Secure Agent にデプロイされたプロセスの呼び出しに使用できるロードバランサ URL。</p> <p>ロードバランサを使用する場合に適用されます。</p>

次の Java 仮想マシンのプロパティを設定できます。

名前	通信方式	説明
min-heap	Secure Agent チャネル	プロセスサーバーが Tomcat JVM に割り当てる最小ヒープメモリ。
max-heap	Secure Agent チャネル	プロセスサーバーが Tomcat JVM に割り当てる最大ヒープメモリ。
additional-properties	Secure Agent チャネル	Tomcat JVM セットに追加できるカスタムシステムプロパティ。例えば、カスタムプロパティ-Dsun.net.inetaddr.ttl=60 を設定できます。

以下のコネクタプロパティを設定できます。

名前	通信方式	説明
http-port	HTTP	Secure Agent がデータを送信する HTTP ポート。デフォルトのポートは 7080 です。 REST および SOAP エンドポイントの URL の構造について詳しくは、アプリケーションの統合ヘルプを参照してください。
http-maxThreads	HTTP	プロセスサーバーが HTTP を介してアプリケーションの統合で作成する接続の最大数。
http-connectionTimeout	HTTP	プロセスサーバーが HTTP 接続の応答を待機する最大時間（ミリ秒単位）。
https-port	HTTPS	Secure Agent がデータを送信する HTTPS ポート。デフォルトのポートは 7443 です。 REST および SOAP エンドポイントの URL の構造について詳しくは、アプリケーションの統合ヘルプを参照してください。
https-maxThreads	HTTPS	プロセスサーバーが HTTPS を介してアプリケーションの統合で作成する接続の最大数。
https-connectionTimeout	HTTPS	プロセスサーバーが HTTPS 接続の応答を待機する最大時間（ミリ秒単位）。
secure-channel-maxThreads	Secure Agent チャネル	プロセスサーバーがアプリケーションの統合で作成する接続の最大数。
secure-channel-connectionTimeout	Secure Agent チャネル	プロセスサーバーが接続の応答を待機する最大時間（ミリ秒単位）。

以下のデータベースプロパティを設定できます。

名前	通信方式	説明
type	Secure Agent チャンネル	プロセスサーバーが実行されるデータベースタイプ。 重要: この設定は変更しないでください。アプリケーションの統合 Secure Agent は他のデータベースをサポートしていません。
driver	Secure Agent チャンネル	プロセスサーバーが実行されるデータベースドライバ。 重要: この設定は変更しないでください。Informatica Cloud Secure Agent は他のデータベースをサポートしていません。
URL	Secure Agent チャンネル	プロセスサーバーがデータベースにアクセスするときの URL。 重要: この設定は変更しないでください。Informatica Cloud Secure Agent は他のデータベースをサポートしていません。
maxActive	Secure Agent チャンネル	プロセスサーバーデータベースに同時に割り当てられるアクティブな接続の最大数。
maxIdle	Secure Agent チャンネル	データベースで一度にアイドル状態のままにできる接続の最大数。アイドル状態の接続数がこの数を越えると、プロセスサーバーは接続を解放します。
maxWait	Secure Agent チャンネル	接続が存在しない場合にデータベースが待機する最大時間。
connection-properties	Secure Agent チャンネル	データベース接続プロパティのキーと値のペア。デフォルトでは、一部のキーが使用できます。 デフォルトのキーは削除しないでください。ただし、次のキーの値は変更できます。 他のキーと値のペアを追加できます。例えば、次のキーと値のペアを追加できます。 キー: autoReconnect 値: true

以下のログプロパティを設定できます。

名前	通信方式	説明
1catalina_org_apache_juli_FileHandler_level	Secure Agent チャンネル	ファイル内のログのレベル: <Secure Agent インストールディレクトリ>\apps \process-engine\logs \catalina.log. デフォルト: FINE
2localhost_org_apache_juli_FileHandler_level	Secure Agent チャンネル	ファイル内のログのレベル: <Secure Agent インストールディレクトリ>\apps \process-engine\logs \localhost.log. デフォルト: FINE

名前	通信方式	説明
3manager_org_apache_juli_FileHandler_level	Secure Agent チャンネル	ファイル内のログのレベル: <Secure Agent インストールディレクトリ>\apps\process-engine\logs\manager.log. デフォルト: FINE
4host_manager_org_apache_juli_FileHandler_level	Secure Agent チャンネル	ファイル内のログのレベル: <Secure Agent インストールディレクトリ>\apps\process-engine\logs\host-manager.log. デフォルト: FINE
java_util_logging_ConsoleHandler_level	Secure Agent チャンネル	Tomcat の起動時に表示される CMD ウィンドウでのログのレベル。 デフォルト: FINE
org_apache_catalina_core_ContainerBase_Catalina_localhost_level	Secure Agent チャンネル	仮想マシンで Tomcat をホストするときの localhost.log ファイルでのログのレベル。 デフォルト: INFO
org_apache_catalina_core_ContainerBase_Catalina_localhost_manager_level	Secure Agent チャンネル	仮想マシンで Tomcat をホストするときの manager.log ファイルでのログのレベル。 デフォルト: INFO
org_apache_catalina_core_ContainerBase_Catalina_localhost_host-manager_level	Secure Agent チャンネル	仮想マシンで Tomcat をホストするときの host-manager.log ファイルでのログのレベル。 デフォルト: INFO

デフォルト接続データベースのプロパティ

次の表では、connection-properties データベースプロパティで利用可能なデフォルトキーについて説明します。

キー	説明
timeBetweenEvictionRuns	アイドル状態のオブジェクト evictor スレッドの実行と実行の間にプロセスサーバーが待機する時間（ミリ秒）。
testOnBorrow value	プロセスサーバーはプールからオブジェクトを借用する前にオブジェクトを検証します。プロセスサーバーがオブジェクトを検証できない場合、プールからこのオブジェクトは削除されます。次に、プロセスサーバーは別のオブジェクトを借用しようとします。
testWhileIdle	プロセスサーバーは、アイドル状態のオブジェクト evictor（存在する場合）によってオブジェクトを検証します。プロセスサーバーがオブジェクトを検証できない場合、プールからこのオブジェクトは削除されます。
validationQuery value	呼び出し元に返す前にこのプールの接続を検証する SQL クエリ。このプロパティを指定する場合は、クエリが 1 つ以上の行を返す SQL SELECT ステートメントである必要があります。

ログレベル

次の表に、プロセスサーバーの【ログ】プロパティで設定できるレベルを示しています。

レベル	説明
SEVERE	エラーを記録します。
WARNING	潜在的に有害な状況を記録します。
INFO	アプリケーションの進行状況の概要を表す情報イベントを記録します。
CONFIG	INFO レベルよりも詳細な情報イベントを記録します。
FINE	アプリケーションのデバッグに使用できる詳細な情報イベントを記録します。
FINER	FINE レベルよりも詳細な情報イベントを記録します。
FINEST	すべてのイベントを記録します。

プロセスサーバーのサイズ決定に関する推奨事項

作業負荷に応じて Secure Agent のプロセスサーバーサービスを設定します。

リソースを最適化するには、次のサイズ決定に関する推奨事項を参照してください。

推奨事項	小	中	大
プロセス数	75	175	350
リソースキャッシュ (MB)	75	175	350
作業マネージャの最小スレッドプール	50	100	150
作業マネージャの最大スレッドプール	250	500	750
JVM 最小ヒープ (MB)	デフォルト	768	1024
JVM 最大ヒープ (MB)	デフォルト	デフォルト	4096

デフォルトの [JVM 最小ヒープ] は 512 MB で、デフォルトの [JVM 最大ヒープ] は 1536 MB です。

[プロセス数]、[リソースキャッシュ]、[作業マネージャの最小スレッドプール]、および [作業マネージャの最大スレッドプール] を設定するには、次の手順を実行します。

1. [Informatica Cloud] > [モニタ] > [サービスおよびプロセスコンソール] に移動します。
2. [コンソール] リストから、Secure Agent を選択します。
3. [管理] > [サーバーの設定] > [サーバープロパティ] に移動します。
4. [プロセス数]、[リソースキャッシュ]、[作業マネージャの最小スレッドプール]、および [作業マネージャの最大スレッドプール] プロパティの値を変更します。
5. [保存] をクリックします。

[JVM 最小ヒープ] および [JVM 最大ヒープ] を設定するには、次の手順を実行します。

1. [Informatica Cloud] > [設定] > [ランタイム環境] に移動します。
2. Secure Agent を選択し、ページ上部で [編集] をクリックします。
3. [システム構成の詳細] までスクロールします。
4. [JVM 最小ヒープ] および [JVM 最大ヒープ] の値を変更します。
5. ページ上部の [OK] をクリックします。

[プロセス数]、[リソースキャッシュ]、[作業マネージャの最小スレッドプール]、[作業マネージャの最大スレッドプール] を設定するには、アプリケーション統合コンソールサービスの [サーバー設定] セクションに移動します。

[JVM 最小ヒープ] および [JVM 最大ヒープ] を設定するには、管理者サービスの [ランタイム環境] セクションに移動します。

UNIX オペレーティングシステムでプロセスサーバーを起動すると、次のエラーが表示されることがあります。

Cannot write to temp location [/tmp]

このエラーが発生するのは、UNIX が 1 つのプロセスで作成できるファイルの数を制限しているためです。1 つのプロセスで作成できるファイルの最大数は 1024 です。

このエラーを回避するには、開くファイルの制限を少なくともデフォルト値である 1024 の 10 倍に増やすことをお勧めします。システム管理者に、最大ユーザープロセスなどのその他の関連パラメータの値を増やすように依頼します。

Secure Agent のためのプロセスサーバーのサイズ決定の詳細については、以下のドキュメントを参照してください。

<https://knowledge.informatica.com/s/article/DOC-17439>

Secure Agent との通信

Informatica Intelligent Cloud Services は、Secure Agent チャンネルまたは HTTP や HTTPS 直接リンクを介して Secure Agent からプロセスサーバーにデータを送信します。

Secure Agent は、次の 2 つの方法でプロセスサーバーと通信します。

Secure Agent チャンネル

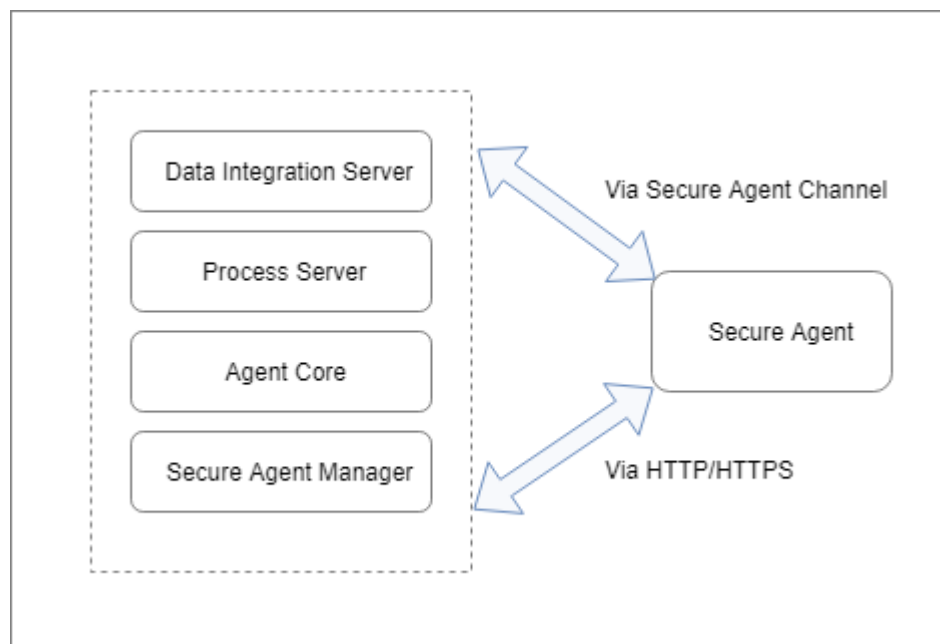
接続された各 Secure Agent とプロセスサーバー間にトンネルを作成するセキュアチャンネル。

HTTP または HTTPS

Secure Agent がプロセスサーバーにデータを直接送信する場合のプロトコル。この通信方法を使用すると、Informatica Intelligent Cloud Services は認証プロバイダに対して資格情報を検証します。

各プロセスサーバープロパティが使用する通信方法の詳細については、[Process Server Properties \(ページ 49\)](#) を参照してください。

次の図に、Secure Agent とプロセスサーバー間の 2 つの通信方法を示します。



プロセスサーバーのための Secure Agent の設定

ビジネスのニーズに応じて、アセットを単一の Secure Agent、Secure Agent グループ、Secure Agent クラスタにデプロイします。

アプリケーションの統合プロセス、接続、またはサービスコネクタを Secure Agent にデプロイする場合、これらのアセットを Secure Agent のプロセスサーバーサービスにデプロイします。そのプロセスサーバーサービスを使用するすべての Secure Agent は、同じ PostgreSQL データベースを使用します。

アセットを次の Secure Agent 構成に割り当てることができます。

単一の Secure Agent

単一の Secure Agent はグループ内の唯一のエージェントであったり、グループの複数のエージェントの 1 つであったりする可能性があります。

詳細については、「[単一の Secure Agent へのデプロイ](#)」(ページ 58)を参照してください。

Secure Agent グループ

Secure Agent グループには複数のエージェントが含まれます。アセットを Secure Agent グループにデプロイすると、Informatica Intelligent Cloud Services によって負荷分散が実行されます。ステートレス要求を処理する場合や OData 要求専用の Secure Agent を使用する場合、要求を分散させるために Secure Agent グループ構成を使用します。

詳細については、「[Secure Agent グループへのデプロイ](#)」(ページ 59)を参照してください。

Secure Agent クラスタ

Secure Agent クラスタは、1 つのマスタ Secure Agent を持つエージェントグループです。すべてのプロセスサーバーでプロセス実行アクティビティに関する情報を受信するようにする場合は、Secure Agent クラスタ構成を使用します。

詳細については、「[Secure Agent クラスタへのデプロイ](#)」(ページ 60)を参照してください。

以下の表は、さまざまなシナリオにおける Secure Agent のプロセス実行の概要を示しています。

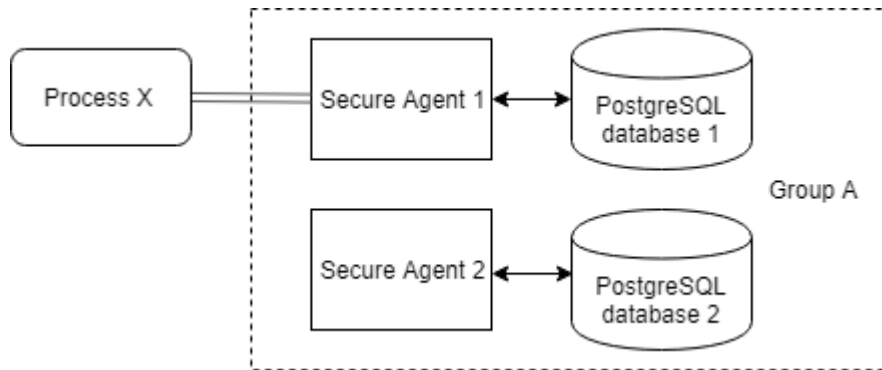
	単一の Secure Agent	Secure Agent グループ	Secure Agent クラスタ
使用可能なエージェント	プロセスが実行される。	プロセスが実行される。	プロセスが実行される。
使用不可能なエージェント	プロセスは実行されない。	プロセスは使用可能な Secure Agent のいずれかで実行される。	プロセスは使用可能な Secure Agent のいずれかで実行される。
エージェントが実行中に停止	プロセスは実行されない。	Secure Agent が停止したときにプロセスが停止される。	別の Secure Agent でプロセスの実行が継続される。

単一の Secure Agent へのデプロイ

アセットをグループ内の単一のエージェントに直接デプロイできます。

アセットを単一の Secure Agent にデプロイする場合、Secure Agent グループ内の他のプロセスサーバーはいずれも、アセット定義を受信しません。

次の図は、プロセス X を Secure Agent 1 に直接デプロイした場合の構成例を示しています。



プロセス X を実行できるのは Secure Agent 1 だけです。Secure Agent 1 が使用不可能になると、プロセスは実行されません。

Secure Agent グループへのデプロイ

Secure Agent グループには複数のエージェントが含まれます。アセットを Secure Agent グループにデプロイすることができます。

ステートレス要求を処理する場合や OData 要求専用の Secure Agent を使用する場合、要求を分散させるために Secure Agent グループ構成を使用します。

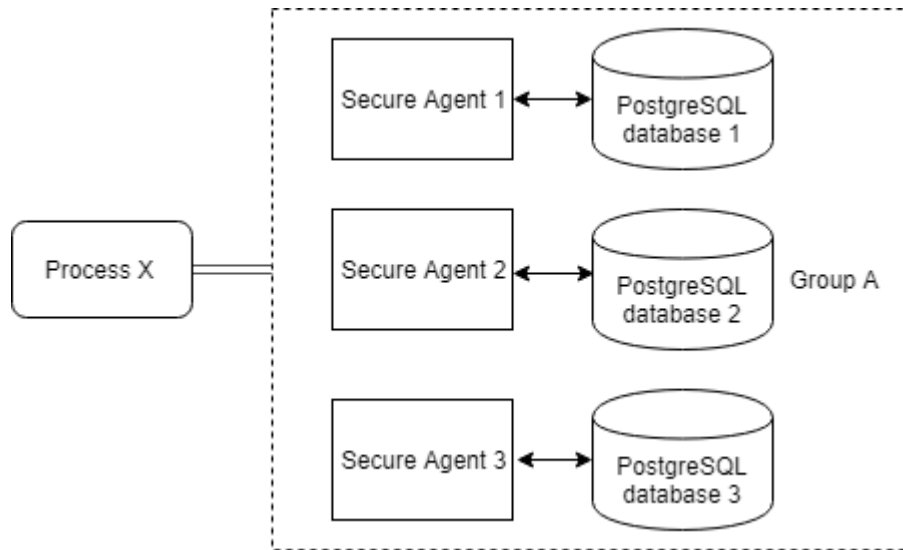
Secure Agent グループでは、受信された要求が Informatica Intelligent Cloud Services によって、使用可能な Secure Agent にラウンドロビン方式でディスパッチされます。

アセットを Secure Agent グループにデプロイすると、負荷分散された構成を使用することになります。Informatica Intelligent Cloud Services によって負荷分散が実行されます。また、load-balance-url プロセスサーバープロパティを設定して、カスタムロードバランサを使用することもできます。詳細については、[Process Server Properties \(ページ 49\)](#)を参照してください。

Secure Agent グループの詳細については、「ランタイム環境」を参照してください。

グループ内のすべての Secure Agent は個別の PostgreSQL データベースを使用します。アセットを Secure Agent グループにデプロイすると、グループ内のすべてのプロセスサーバーが、新規のアセット定義または更新されたアセット定義に関する詳細を受信します。ただし、グループ内の他のプロセスサーバーはアセットの実行アクティビティに関する詳細を受信しません。例えば、プロセス実行中に Secure Agent で障害が発生しても、プロセスはグループ内の別の Secure Agent で継続して実行されません。

次の図は、プロセス X を Secure Agent グループ A にデプロイした場合の構成例を示しています。



プロセス X を変更して再パブリッシュすると、3 つすべての Secure Agent が更新された定義を受信します。すべての Secure Agent がプロセスを実行できます。

例えば、プロセスが開始され、Secure Agent 1 と Secure Agent 2 が使用不可能な場合、負荷分散された構成によって、Secure Agent 3 がプロセス X を実行することが保証されます。ただし、Secure Agent 1 と Secure Agent 2 は、プロセスが失敗したか正常に終了したかどうかについての情報を受信しません。プロセス X の実行中に Secure Agent 3 が停止した場合、プロセスはそれ以降実行されません。

Secure Agent クラスタへのデプロイ

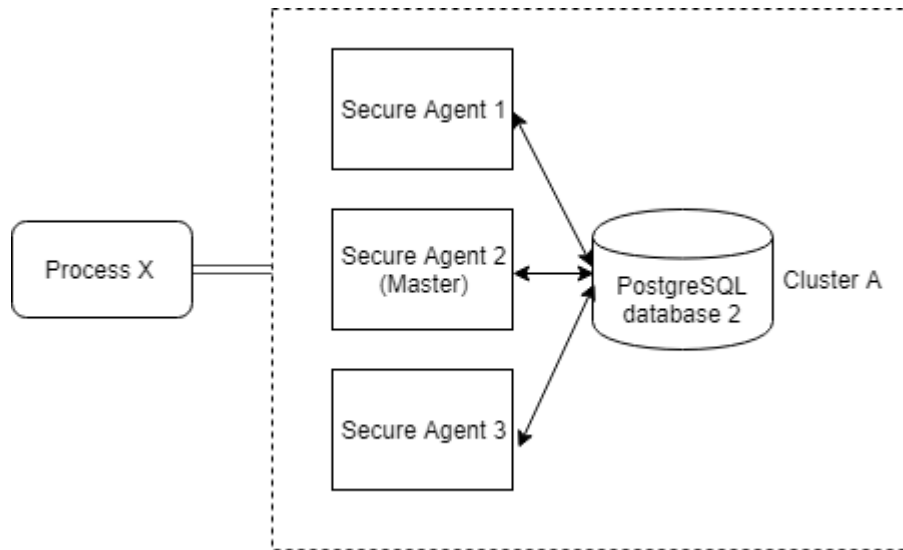
Secure Agent クラスタは、1 つのマスタ Secure Agent を持つエージェントグループです。アセットを Secure Agent クラスタにデプロイすることができます。

アセットを Secure Agent クラスタにデプロイすると、すべてのプロセスサーバーはプロセス実行アクティビティに関する情報を受信します。マスタ Secure Agent は情報を受信し、他のすべての Secure Agent に送信します。プロセス実行中に Secure Agent で障害が発生すると、プロセスはクラスタ内の別の Secure Agent で継続して実行されます。

クラスタ内のすべてのプロセスサーバーはマスタエージェントの PostgreSQL データベースを共有します。

マスタ Secure Agent を定義するには、`primary-node` プロセスサーバープロパティを使用してください。詳細については、[Process Server Properties \(ページ 49\)](#)を参照してください。

次の図は、プロセス X を Secure Agent クラスタ A にデプロイした場合の構成例を示しています。



Secure Agent 3 がプロセス X の実行を開始し、途中でこのエージェントが停止した場合、Secure Agent 1 または Secure Agent 2 がそのプロセスの実行を継続します。

PostgreSQL データベースのインストールとアップグレードの前提条件

PostgreSQL データベースをインストールまたはアップグレードする前に、次の前提条件が満たされていることを確認してください。

- 十分な空きディスク容量がある。Informatica では、最小空きディスク容量を次のディレクトリの Data フォルダの 2 倍のサイズにすることをお勧めします。
<Secure Agent installation directory>\apps\process-engine\data\PostGreSql\Data
- スクリプトを実行して PostgreSQL データベースをインストールまたはアップグレードするユーザーは、次のディレクトリ内のフォルダの読み取り、書き込み、および変更の権限を持っている必要がある。
<Secure Agent installation directory>\apps\process-engine\data

- PostgreSQL データベースとプロセスサーバーを停止する。Secure Agent が詳細クラスタと関連付けられている場合は、すべてのノードでプロセスサーバーを停止する必要があります。

注: このステップは、upgrade コマンドラインオプションを使用して db_upgrade スクリプトを実行する場合に必要です。ただし、check コマンドラインオプションを使用して db_upgrade スクリプトを実行する場合はオプションです。したがって、check コマンドラインオプションを使用すると、db_upgrade スクリプトをダウンタイムなしで実行できます。

- Linux オペレーティングシステムでは、オペレーティングシステムユーザーのロケールは、PostgreSQL データベースのエンコーディングと一致するように UTF-8 エンコーディングを使用する必要がある。使用しない場合、プロセスサーバーの起動に失敗します。
- Linux オペレーティングシステムでは、GNU C (GLIBC) ライブラリファイルがバージョン 2.14 以降であることを確認する。バージョン番号を見つけるには、次のコマンドを実行します: `ldd --version`

Windows での PostgreSQL データベースの管理

PostgreSQL データベースを管理するには、バイナリとユーティリティスクリプトを使用します。

重要: PostgreSQL データベースを管理するには、システム管理者権限を持たないユーザーとしてログインする必要があります。システム管理者は、PostgreSQL バイナリとユーティリティスクリプトを実行できません。

PostgreSQL バイナリに基づいていくつかのユーティリティスクリプトが作成されています。これらのユーティリティスクリプトを使用すると、PostgreSQL データベースを簡単に管理できます。

次のディレクトリには、PostgreSQL データベースのファイルが含まれます。

- PostgreSQL バイナリ: <Secure Agent installation directory>\apps\process-engine\data\db\postgresql-windows-x64-binaries\pgsql\bin
- PostgreSQL ユーティリティスクリプト: <Secure Agent installation directory>\apps\process-engine\data\db\util
- PostgreSQL ログ: <Secure Agent installation directory>\apps\process engine\logs\PostGreSql\postgresql.log
- PostgreSQL データ: <Secure Agent installation directory>\apps\process engine\data\PostGreSql\Data

PostgreSQL スクリプトの詳細については、<https://www.postgresql.org/docs/current/static/index.html> で PostgreSQL のヘルプを参照してください。

以降の一部のセクションには、次のデフォルト値を使用するサンプルコマンドが含まれています。

- デフォルトのデータベース名: activevos
- デフォルトのデータベースユーザー名: bpeluser
- デフォルトのデータベースパスワード: bpel

Windows での PostgreSQL データベースのバックアップ

PostgreSQL データベースをバックアップするには、スクリプト db_backup.bat を使用します。

PostgreSQL データベースをバックアップするには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。
2. 次のコマンドを実行します。
db_backup.bat <dbusername> <dbpassword> <バックアップファイルへのパスと、「.dump」という拡張子を持つバックアップファイルの名前> <dbport>

例えば、次のコマンドを実行すると、Secure Agent はバックアップファイル「BackupFile1.dump」を C:\postgre\backup に作成します。

```
db_backup.bat bpeluser bpel "C:\postgre\backup\BackupFile1.dump" 6432
```

注: dbport 引数はオプションです。デフォルトのポート 5432 とは異なるポートを使用する場合は、dbport 引数を指定します。

Windows での PostgreSQL データベースのリストア

PostgreSQL データベースをバックアップファイルからリストアするには、コマンド db_restore.bat を使用します。

PostgreSQL データベースファイルをバックアップファイルからリストアするには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。

2. 次のコマンドを実行します。

```
db_restore.bat <dbusername> <dbpassword> <ダンプファイルへのパス> <dbport>
```

例えば、次のコマンドを実行すると、ファイル BackupFile1.dump を使用して PostgreSQL データベースがリストアされます。

```
db_restore.bat bpeluser bpel "C:\postgre\backup\BackupFile1.dump" 6432
```

注: dbport 引数はオプションです。デフォルトのポート 5432 とは異なるポートを使用する場合は、dbport 引数を指定します。

Windows での PostgreSQL データベースのリセット

PostgreSQL データベースをシャットダウンしてから、db_reset.bat コマンドを使用してリセットします。

PostgreSQL データベースを元の状態にリセットするには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。
2. サーバーをシャットダウンするには、次のコマンドを実行します。
server_stop.bat
3. PostgreSQL データベースをリセットするには、次のコマンドを実行します。
db_reset.bat

Windows での PostgreSQL サーバーの起動

Windows で PostgreSQL サーバーを起動するには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。
2. 次のスクリプトを実行します。
server_start.bat

注: デフォルトの 5432 ポートを使用しない場合は、次のように引数としてポート番号を渡す必要があります。server_start.bat <port_number>

例: server_start.bat 6789

Windows での PostgreSQL サーバーの停止

Windows で PostgreSQL サーバーを停止するには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。
2. 次のスクリプトを実行します。
server_stop.bat

Windows での PostgreSQL サーバーステータスの取得

Windows で PostgreSQL サーバーのステータスを取得するには、次の手順を実行します。

1. <Secure Agent installation directory>\apps\process-engine\data\db\util に移動します。
2. 次のスクリプトを実行します。
server_status.bat

注: デフォルトの 5432 ポートを使用しない場合は、次のように引数としてポート番号を渡す必要があります。server_status.bat <port_number>

例: server_status.bat 6789

Windows での PostgreSQL データベースのクリーンアップ

PostgreSQL データベースをクリーンアップし、古いタブルを削除して領域を確保します。スクリプト `db_maintenance.bat` を使用して、PostgreSQL データベースをクリーンアップします。

デフォルトで、PostgreSQL データベースの自動クリーンアップが行われます。データベースを手動でクリーンアップする場合は、次の手順を実行します。

1. `<Secure Agent installation directory>\apps\process-engine\data\db\util` に移動します。
2. データベース全体をクリーンアップするには、次のコマンドを実行します。
`db_maintenance.bat <dbusername> <dbpassword> <dbport> vacuum`
3. 単一テーブルをクリーンアップするには、次のコマンドを実行します。
`db_maintenance.bat <dbusername> <dbpassword> <dbport> vacuum <tablename>`

例えば、次のコマンドを実行すると、「`aeoprocesslogdata`」テーブルがクリーンアップされます。

```
db_maintenance.bat bpeluser bpel 5432 vacuum aeoprocesslogdata
```

注: デフォルトのポート 5432 を使用する場合でも、`dbport` 引数は必須です。

また、PostgreSQL データベースの期間クリーンアップをスケジュールすることもできます。詳細を確認するには、*[管理者]* の *[メンテナンス]* セクションの *[PostgreSQL メンテナンスのスケジュール]* トピックを参照してください。

Windows での PostgreSQL データベースの再インデックス化

PostgreSQL でデータをクリーンアップした後、インデックスをクリーンアップして使用領域を解放するには、再インデックス化オプションを使用します。スクリプト `db_maintenance.bat` を使用して、PostgreSQL データベースを再インデックス化します。

PostgreSQL データベースを再インデックス化するには、次の手順を実行します。

1. `<Secure Agent インストールディレクトリ>\apps\process-engine\data\db\util` に移動します。
2. データベース全体を再インデックス化するには、次のコマンドを実行します。
`db_maintenance.bat <dbusername> <dbpassword> <dbport> reindex`
3. 単一テーブルを再インデックス化するには、次のコマンドを実行します。
`db_maintenance.bat <dbusername> <dbpassword> <dbport> reindex <tablename>`

例えば、次のコマンドを実行すると、「`aeoprocesslogdata`」テーブルが再インデックス化されます。

```
db_maintenance.bat bpeluser bpel 5432 reindex aeoprocesslogdata
```

注: デフォルトのポート 5432 を使用する場合でも、`dbport` 引数は必須です。

また、PostgreSQL データベースの期間インデックス化をスケジュールすることもできます。詳細を確認するには、*[管理者]* の *[メンテナンス]* セクションの *[PostgreSQL メンテナンスのスケジュール]* トピックを参照してください。

Windows でのトランザクションログのリセット

制御情報の破損が原因で PostgreSQL サーバーが起動しない場合は、コマンド `pg_resetxlog.exe` を使用して制御情報をリセットします。

PostgreSQL データベースの制御情報をリセットするには、次の手順を実行します。

1. `<Secure Agent installation directory>\apps\process-engine\data\db\postgresql-windows-x64-binaries\pgsql\bin` に移動します。

2. 次のコマンドを実行します。

```
pg_resetxlog.exe -D <path to postgresQL data directory>
```

例えば、次のコマンドを実行すると、「Data」ディレクトリのトランザクションログがリセットされます。

```
pg_resetxlog.exe -D "C:\postgre\apps\process-engine\data\PostGreSql\Data"
```

Linux での PostgreSQL データベースの管理

PostgreSQL データベースを管理するには、バイナリとユーティリティスクリプトを使用します。

重要: PostgreSQL データベースを管理するには、ルートアクセス権を持たないユーザーとしてログインする必要があります。ルートユーザーは、PostgreSQL バイナリとユーティリティスクリプトを実行できません。

PostgreSQL バイナリに基づいていくつかのユーティリティスクリプトが作成されています。これらのユーティリティスクリプトを使用すると、PostgreSQL データベースを簡単に管理できます。

次のディレクトリには、PostgreSQL データベースのファイルが含まれます。

- PostgreSQL バイナリ: <Secure Agent installation directory>/apps/process-engine/data/db/postgresql-linux-x64-binaries/pgsql/bin
- PostgreSQL ユーティリティスクリプト: <Secure Agent installation directory>/apps/process-engine/data/db/util
- PostgreSQL ログ: <Secure Agent installation directory>/apps/process-engine/logs/PostGreSql/postgresql.log
- PostgreSQL データ: <Secure Agent installation directory>/apps/process-engine/data/PostGreSql/Data

PostgreSQL スクリプトの詳細については、<https://www.postgresql.org/docs/current/static/index.html> で PostgreSQL のヘルプを参照してください。

一部のセクションには、次のデフォルト値を使用するサンプルコマンドが含まれています。

- デフォルトのデータベース名: activevos
- デフォルトのデータベースユーザー名: bpeluser
- デフォルトのデータベースパスワード: bpel

Linux での PostgreSQL データベースのバックアップ

PostgreSQL データベースをバックアップするには、スクリプト db_backup を使用します。

PostgreSQL データベースをバックアップするには、次の手順を実行します。

1. <Secure Agent installation directory>/apps/process-engine/data/db/util に移動します。
2. 次のコマンドを実行します。

```
db_backup.sh <dbusername> <dbpassword> <バックアップファイルの名前とバックアップファイルへのパス>.dump <dbport>
```

例えば、次のコマンドを実行すると、Secure Agent はバックアップファイル「backupfile1.dump」を/home/data/myfolder/に作成します。

```
db_backup.sh bpeluser bpel "/home/data/myfolder/backupfile1.dump" 6432
```

注: dbport 引数はオプションです。デフォルトのポート 5432 とは異なるポートを使用する場合は、dbport 引数を指定します。

Linux での PostgreSQL データベースのリストア

PostgreSQL データベースをバックアップファイルからリストアするには、スクリプト `db_restore.sh` を使用します。

PostgreSQL データベースファイルをバックアップファイルからリストアするには、次の手順を実行します。

1. `<Secure Agent installation directory>/apps/process-engine/data/db/util` に移動します。
2. 次のコマンドを実行します。
`db_restore.sh <dbusername> <dbpassword> <ダンプファイルへのパス> <dbport>`

例えば、次のコマンドを実行すると、ファイル `backupfile1.dump` を使用して PostgreSQL データベースがリストアされます。

```
db_restore.sh bpeluser bpel "/home/data/myfolder/backupfile1.dump" 6432
```

注: `dbport` 引数はオプションです。デフォルトのポート 5432 とは異なるポートを使用する場合は、`dbport` 引数を指定します。

Linux での PostgreSQL データベースのリセット

最初に PostgreSQL データベースをシャットダウンしてから、スクリプト `db_reset.sh` を使用してリセットします。

PostgreSQL データベースを元の状態にリセットするには、次の手順を実行します。

1. `<Secure Agent installation directory>/apps/process-engine/data/db/util` に移動します。
2. サーバーをシャットダウンするには、次のスクリプトを実行します。
`server_stop.sh`
3. PostgreSQL データベースをリセットするには、次のスクリプトを実行します。
`db_reset.sh`

Linux での PostgreSQL サーバーの起動

Linux で PostgreSQL サーバーを起動するには、次の手順を実行します。

1. `<Secure Agent installation directory>/apps/process-engine/data/db/util` に移動します。
2. 次のスクリプトを実行します。
`server_start.sh`

注: デフォルトの 5432 ポートを使用しない場合は、次のように引数としてポート番号を渡す必要があります。
`server_start.sh <port_number>`

例: `server_start.sh 6789`

Linux での PostgreSQL サーバーの停止

PostgreSQL サーバーを停止するには、次の手順を実行します。

1. `<Secure Agent installation directory>/apps/process-engine/data/db/util` に移動します。
2. 次のスクリプトを実行します。
`server_stop.sh`

Linux での PostgreSQL サーバーステータスの取得

Linux で PostgreSQL サーバーステータスを取得するには、次の手順を実行します。

1. <Secure Agent installation directory>/apps/process-engine/data/db/util に移動します。

2. 次のスクリプトを実行します。

```
server_status.sh
```

注: デフォルトの 5432 ポートを使用しない場合は、次のように引数としてポート番号を渡す必要があります。 server_status.sh <port_number>

例: server_status.sh 6789

Linux での PostgreSQL データベースのクリーンアップ

PostgreSQL データベースをクリーンアップし、古いタプルを削除して領域を確保します。スクリプト db_maintenance.sh を使用して、PostgreSQL データベースをクリーンアップします。

デフォルトで、PostgreSQL データベースの自動クリーンアップが行われます。データベースを手動でクリーンアップする場合は、次の手順を実行します。

1. <Secure Agent installation directory>/apps/process-engine/data/db/util に移動します。

2. データベース全体をクリーンアップするには、次のコマンドを実行します。

```
db_maintenance <dbusername> <dbpassword> <dbport> vacuum
```

3. 単一テーブルをクリーンアップするには、次のコマンドを実行します。

```
db_maintenance.sh <dbusername> <dbpassword> <dbport> vacuum <tablename>
```

例えば、次のコマンドを実行すると、「aeoprocesslogdata」テーブルがクリーンアップされます。

```
db_maintenance.sh bpeluser bpel 5432 vacuum aeoprocesslogdata
```

注: デフォルトのポート 5432 を使用する場合でも、dbport 引数は必須です。

また、PostgreSQL データベースの期間クリーンアップをスケジュールすることもできます。詳細を確認するには、[管理者] の [メンテナンス] セクションの [PostgreSQL メンテナンスのスケジュール] トピックを参照してください。

Linux での PostgreSQL データベースの再インデックス化

PostgreSQL でデータをクリーンアップした後、インデックスをクリーンアップして使用領域を解放するには、再インデックス化オプションを使用します。スクリプト db_maintenance.sh を使用して、PostgreSQL データベースを再インデックス化します。

PostgreSQL データベースを再インデックス化するには、次の手順を実行します。

1. <Secure Agent installation directory>/apps/process-engine/data/db/util に移動します。

2. データベース全体を再インデックス化するには、次のコマンドを実行します。

```
db_maintenance <dbusername> <dbpassword> <dbport> reindex
```

3. 単一テーブルを再インデックス化するには、次のコマンドを実行します。

```
db_maintenance.sh <dbusername> <dbpassword> <dbport> reindex <tablename>
```

例えば、次のコマンドを実行すると、「aeoprocesslogdata」テーブルが再インデックス化されます。

```
db_maintenance.sh bpeluser bpel 5432 reindex aeoprocesslogdata
```

注: デフォルトのポート 5432 を使用する場合でも、dbport 引数は必須です。

また、PostgreSQL データベースの期間インデックス化をスケジュールすることもできます。詳細を確認するには、[管理者] の [メンテナンス] セクションの [PostgreSQL メンテナンスのスケジュール] トピックを参照してください。

Linux でのトランザクションログのリセット

制御情報の破損が原因で PostgreSQL サーバーが起動しない場合は、コマンド `pg_resetxlog` を使用して制御情報をリセットします。

PostgreSQL データベースの制御情報をリセットするには、次の手順を実行します。

1. `<Secure Agent installation directory>/apps/process-engine/data/db/postgresql-linux-x64-binaries/pgsql/bin` に移動します。
2. 次のコマンドを実行します。
`pg_resetxlog -D <path to postgresql data directory>`

例えば、次のコマンドを実行すると、「Data」ディレクトリのトランザクションログがリセットされます。

```
pg_resetxlog -D "home/apps/process engine/data/PostGreSql/Data"
```

PostgreSQL データベースのアップグレード

PostgreSQL データベースをバージョン 9.5.2、12.4、または 12.6 からバージョン 13.5 にアップグレードできます。バージョン 13.5 ではセキュリティ、パフォーマンス、および拡張性が強化されています。

Informatica が提供するスクリプトを手動で実行することで、都合に合わせてアップグレードを行うことができます。このスクリプトを実行すると既存のデータベースバージョンのバックアップが作成され、アップグレードで問題が発生した場合でも古いデータベースバージョンに復元することが可能です。

PostgreSQL データベースをアップグレードする方法の詳細については、次のコミュニティ記事を参照してください。

<https://knowledge.informatica.com/s/article/DOC-18945>

レプリケーション技術を使用した PostgreSQL データベースのアップグレード

プロセスサーバーの PostgreSQL データベースのバージョンがバージョン 13.5 より前の場合、プロセスサーバーの再起動時にデータベースをアップグレードできます。

プロセスサーバーの PostgreSQL データベースのアップグレードは、デフォルトで無効になっています。Secure Agent の PostgreSQL データベースのレプリケーションアップグレードを有効にするには、Secure Agent のカスタムプロパティを追加する必要があります。

Secure Agent のカスタムプロパティを追加するには、次の手順を実行します。

1. [Administrator] で、[ランタイム環境] を選択します。
2. [ランタイム環境] ページで、Secure Agent の名前をクリックします。
Secure Agent グループ内の Secure Agent を一覧表示するには、Secure Agent グループの展開が必要になる場合があります。
3. [詳細] タブをクリックします。
4. 右上隅の [編集] をクリックします。
5. [カスタム構成の詳細] 領域までスクロールダウンします。

6. 次の画像は、[カスタム構成の詳細] 領域を示しています。

Custom Configuration Details					
Service	Type	Sub-type	Name	Value	Sensitive
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>  

7. すでに設定されているカスタムプロパティがある場合、[追加] アイコンをクリックして、新しいプロパティ行を追加します。
8. サービスとして [プロセスサーバー] を選択します。
9. 設定プロパティの種類として [db] を選択します。
10. プロパティ名として「**replication_upgrade**」、値として「**true**」を入力します。
11. [保存] をクリックします。
- プロセスサーバーサービスのステータスが [再起動が必要です] と表示されます。

プロセスサーバーを初めて再起動すると、データベースは既存のデータベースバージョンからデータのレプリケーションを開始します。プロセスサーバーを再起動するたびに、データベースはレプリケーションステータスを検証します。レプリケーションの完了後、プロセスサーバーを次に再起動するときには、古いバージョンの PostgreSQL データベースはシャットダウンされ、最新バージョンのデータベースが自動的に起動されます。

これにより、Informatica が提供するスクリプトを実行して、プロセスサーバーの PostgreSQL データベースを手動で最新バージョンにアップグレードする必要がなくなります。

PostgreSQL 構成ファイル

PostgreSQL データベースをインストールすると、postgresql.conf ファイルが次のディレクトリに自動的に作成されます。

<SecureAgent インストールディレクトリ>\apps\process-engine\data\PostGreSql\Data

postgresql.conf ファイルの設定パラメータにより、監査、認証、暗号化、およびその他の動作に関連するサーバープロパティのデフォルト値を定義します。

postgresql.conf ファイルは、新しいバージョンの PostgreSQL データベースをインストールまたはアップグレードするたびに上書きされます。postgresql.conf ファイルが上書きされると変更が失われるため、カスタマイズされた動作に対してこのファイルを更新しないでください。

user.conf ファイルを使用して、postgresql.conf ファイルで定義されているデフォルト値を上書きします。

user.conf ファイルが存在しない場合は、postgresql.conf ファイルと同じディレクトリにファイルを作成し、値を上書きします。PostgreSQL データベースを再起動すると、変更が有効になります。

PostgreSQL ログローテーションの設定

PostgreSQL ログには、Secure Agent でパッケージ化されている PostgreSQL データベースのログ情報が含まれています。

PostgreSQL ログは次のディレクトリに含まれています。

<Secure Agent インストールディレクトリ>\apps\process-engine\logs\PostGreSql\postgresql.log

時間の経過とともに PostgreSQL ログのサイズは非常に大きくなるため、管理が困難になる可能性があります。ログローテーションを設定してファイルサイズを縮小し、ファイルの管理を簡単にすることができます。時間またはファイルサイズに基づいてログローテーションを設定できます。

1. user.conf という名前のファイルが存在しない場合は、次の場所に作成します。

<SecureAgent インストールディレクトリ>\apps\process-engine\data\PostGreSql\Data

user.conf ファイルは、postgresql.conf ファイルで定義されている値を上書きします。

2. 次のいずれかの手順を実行します。

- 時間に基づいてログをローテーションするには、user.conf ファイルに次のプロパティを追加します。

```
log_filename = 'postgresql-%Y-%m-%d_%H%M%S.log'
log_rotation_age=<value in minutes>
```

例えば、log_rotation_age プロパティの値を 1440 に設定すると、ログファイルは毎日ローテーションされます。

- ファイルサイズに基づいてログをローテーションするには、user.conf ファイルに次のプロパティを追加します。

```
log_filename = 'postgresql-%Y-%m-%d_%H%M%S.log'
log_rotation_size=<value in kilobytes>
log_truncate_on_rotation=on
```

例えば、log_rotation_size プロパティの値を 10240 に設定すると、ファイルサイズが 10 MB を超えた場合にログファイルがローテーションされます。

3. user.conf ファイルを保存します。

4. PostgreSQL データベースを再起動して、変更を有効にします。

プロセスサーバーに対するパブリック証明書とプライベートキーの設定

アプリケーション統合プロセスと接続を使用して SSL 対応エンドポイントに接続する場合は、パブリック証明書やプライベートキーが必要です。

プロセスと接続用のパブリック証明書をプライベートキーを Secure Agent にインポートする必要があります。

プロセスと接続用のパブリック証明書とプライベートキーのインポート

Web サービス、キュー、JDBC 接続などの SSL 対応エンドポイントに接続するには、パブリック証明書やプライベートキーが必要です。

プロセスまたは接続がこれらのエンドポイントへの SSL 対応接続を確立するためにパブリッシュされる Secure Agent マシンに証明書をインポートする必要があります。

パブリック証明書やプライベートキーをインポートするには、次の手順を実行します。

- パブリック証明書の場合は、証明書ファイルを以下の場所に配置して、SecureAgent を再起動します。

<Secure Agent installation directory>\apps\process-engine\conf\certs

- プライベートキーの場合は、キーを以下の場所の ae.keystore ファイルにインポートして、Secure Agent を再起動します。

<Secure Agent installation directory>\apps\process-engine\conf

上記の certs フォルダに x509 形式のパブリック証明書ファイルをインポートし、配置する必要があります。使いやすさとアップグレードとの互換性を確保するために、証明書とキーを同じ場所にインポートする必要があります。

さらに、Informatica Keystore 内に秘密プライベートキーをインポートするには、秘密鍵が同じキーストア形式、つまり PKCS12 ".p12"である必要があります。例えば、秘密鍵が".pfx"形式で提供された場合、それを".p12"に変換する必要があります。これは、証明書プロバイダーで確認できます。

localhost ではなく、ドメイン名で Secure Agent に接続するには、証明書に接続し、certs フォルダにコピーするドメイン名に基づいて証明書を生成できます。

プロセスサーバーの相互認証を有効にする

プロセスサーバーの相互認証を有効にするには、次の手順を実行します。

1. Secure Agent マシンにログインします。
2. 次のディレクトリに移動します。
`<Secure Agent installation directory>/downloads/package-process-engine.<latest_version>/package/app/conf/`
3. `server.xml.mustache` ファイルを編集し、`clientAuth` プロパティの値を `want` から `true` に変更します。
4. `server.xml.mustache` ファイルを保存します。
5. 変更を有効にするには、Secure Agent を再起動します。

注: デフォルトのキーストアは `ae.keystore` で、localhost 証明書を使用してインストールされます。

プロセスサーバーのキーストアとトラストストアの設定については、ナレッジベースの記事 [611562](#) の添付書類を参照してください。

スループットを向上させるためのスレッドプールプロファイルの設定

AMQP や Kafka などのイベントベースのコネクタを使用する場合、デフォルトのスレッドプールプロファイルのスレッドプールサイズを増やすことで、スレッド数を増やすことができます。スレッドプールサイズを増やすには、Secure Agent マシンの `aeEngineConfig.xml.mustache` ファイルを更新します。

デフォルトでは、スレッドプールサイズは 10 スレッドに制限されており、最大 10 個のメッセージを接続によって同時に処理できます。Informatica では、スループットを向上させたい場合にのみスレッドプールサイズを増やすことをお勧めします。これは、追加のスレッドが一部のリソースを占有し、プールが大きくなりすぎることが推奨されないためです。スレッドプールサイズは、すべてのイベントベースの接続に適用されます。

スレッドプールプロファイルを設定するには、次の手順を実行します。

1. Secure Agent マシンにログインします。
2. 次のディレクトリに移動します。
`<Secure Agent インストールディレクトリ>/downloads/package-process-engine.<latest_version>/package/app/webapps/process-engine/WEB-INF/classes`
3. テキストエディタで `aeEngineConfig.xml.mustache` ファイルを開き、次のエントリを検索します。
`<entry name="IAeESBManager">...</entry>`
4. `IAeESBManager` エントリ内で、`camelContext` という名前のサブエントリを検索します。

cluster.enabled オプションのステータスに基づき、次のエントリを見つけます。

- cluster.enabled オプションが無効になっている場合、次の camelContext エントリを編集します。

```
<entry name="Class" value="com.activeee.rt.camel.AeCamelIntegrationManager"/>
<entry name="camelContext">
  <entry name="Class" value="com.activeee.rt.camel.core.AeDefaultCamelContext"/>
</entry>
```
- cluster.enabled オプションが有効になっている場合、次の camelContext エントリを編集します。

```
<entry name="Class" value="com.activeee.rt.cluster.AeClusterDistributedCamelManager"/>
<entry name="camelContext">
  <entry name="Class" value="com.activevos.socrates.connectors.camel.AeSocratesCamelContext"/>
</entry>
```

5. 新しい threadpoolProfile サブエントリを camelContext エントリ内に追加します。

cluster.enabled オプションが無効になっている場合、次の行を追加します。

```
<entry name="camelContext">
<entry name="Class" value="com.activeee.rt.camel.core.AeDefaultCamelContext"/>
<!-- New subentry -->
<entry name="threadpoolProfile">
  <entry name="PoolSize" value="<PoolSizeValue>"/>
  <entry name="MaxPoolSize" value="<MaxPoolSizeValue>"/>
  <entry name="MaxQueueSize" value="<MaxQueueSizeValue>"/>
  <entry name="KeepAliveTime" value="<KeepAliveTimeValue>"/>
  <entry name="TimeUnit" value="SECONDS"/>
  <entry name="AllowCoreThreadTimeout" value="true"/>
  <entry name="RejectedPolicy" value="CallerRuns"/>
</entry>
</entry>
```

cluster.enabled オプションが有効になっている場合、次の行を追加します。

```
<entry name="camelContext">
<entry name="Class" value="com.activevos.socrates.connectors.camel.AeSocratesCamelContext"/>
<!-- New subentry -->
<entry name="threadpoolProfile">
  <entry name="PoolSize" value="<PoolSizeValue>"/>
  <entry name="MaxPoolSize" value="<MaxPoolSizeValue>"/>
  <entry name="MaxQueueSize" value="<MaxQueueSizeValue>"/>
  <entry name="KeepAliveTime" value="<KeepAliveTimeValue>"/>
  <entry name="TimeUnit" value="SECONDS"/>
  <entry name="AllowCoreThreadTimeout" value="true"/>
  <entry name="RejectedPolicy" value="CallerRuns"/>
</entry>
</entry>
```

6. 必要に応じてスレッドプールサイズを増やします。

注: プールは、SecureAgent で実行中のすべてのイベントベースの接続によって使用されます。

7. aeEngineConfig.xml.mustache ファイルを保存します。
8. 変更を有効にするには、Secure Agent を再起動します。

第 14 章

Secure Agent サービスプロパティの設定

Secure Agent サービスプロパティを設定するには、**【ランタイム環境】** ページを開いて Secure Agent を編集します。Secure Agent サービスのプロパティ値を変更、マスク、およびリセットできます。サービスのカスタムプロパティを追加および削除できます。また、Secure Agent 名を変更することもできます。

カスタムプロパティはコネクタ固有です。カスタムプロパティの詳細については、該当するコネクタのヘルプを参照してください。

警告: グループレベルのプロパティ設定を使用する Secure Agent グループのエージェントに対して、エージェントレベルの Secure Agent サービスプロパティ設定を構成しないでください。エージェントレベルのプロパティ設定を構成する場合は、エージェントプロパティを構成する前に、グループレベルのプロパティ設定を削除してください。グループレベルのプロパティ設定の詳細については、*REST API リファレンス*のランタイム環境に関する説明を参照してください。

1. **【ランタイム環境】** ページで、Secure Agent の名前をクリックします。
Secure Agent グループ内の Secure Agent を一覧表示するには、Secure Agent グループの展開が必要になる場合があります。
2. **【詳細】** タブをクリックします。
3. 右上隅の **【編集】** をクリックします。
4. Secure Agent の名前を変更するには、**【エージェント名】** フィールドに新しい名前を入力します。
5. サービスプロパティを編集するには、次の手順を実行します。
 - a. **【システム構成の詳細】** 領域で、サービスを選択します。
 - b. 設定プロパティの種類を選択します。
 - c. 編集するプロパティを含む行で、**【エージェント設定の編集】** アイコンをクリックします。
 - d. プロパティ値を変更するには、新しいプロパティ値を入力します。
プロパティが機密プロパティである場合、プロパティを編集すると既存の値がクリアされます。
 - e. プロパティに機密データが含まれており、Secure Agent の詳細ページで値をマスクする場合は、**【機密】** オプションを有効にします。
機密オプションを有効にすると、入力した値がマスクされます。フィールドが複数行のテキストフィールドである場合、変更を保存した後に値がマスクされます。
 - f. プロパティをシステムデフォルト値にリセットするには、**【エージェント設定をシステムデフォルトにリセット】** アイコンをクリックします。

6. サービスのカスタムプロパティを追加するには、次の手順を実行します。

a. **【カスタム構成の詳細】** 領域までスクロールダウンします。

次の画像は、**【カスタム構成の詳細】** 領域を示しています。

Custom Configuration Details					
Service	Type	Sub-type	Name	Value	Sensitive
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>  

b. すでに設定されているカスタムプロパティがある場合、**【追加】** アイコンをクリックして、新しいプロパティ行を追加します。

c. 設定するサービスを選択します。

d. 設定プロパティの種類を選択します。

e. 設定プロパティタイプにサブタイプがある場合は、適切なサブタイプを選択します。

例えば、ログレベルを決定するには、サブタイプとして **【情報】** または **【デバッグ】** を選択します。

f. プロパティの名前と値を入力します。

g. プロパティに機密データが含まれており、Secure Agent の詳細ページで値をマスクする場合は、**【機密】** オプションを有効にします。

7. カスタムプロパティを削除するには、カスタムプロパティの隣にある **【削除】** アイコンをクリックします。

8. すべての設定プロパティをデフォルト設定にリセットするには、**【すべてリセット】** をクリックします。

9. **【保存】** をクリックします。

索引

C

Cloud Application Integration コミュニティ
URL [6](#)
Cloud 開発者コミュニティ
URL [6](#)

G

GitRepoConnectApp
ローカルリポジトリディレクトリ [37](#)
概要 [37](#)

I

Informatica Intelligent Cloud Services
Web サイト [6](#)
Informatica グローバルカスタマサポート
連絡先情報 [7](#)

N

NetworkRetryInterval
データ統合サーバーのプロパティ [31](#)
NetworkTimeoutPeriod
データ統合サーバーのプロパティ [31](#)

S

Secure Agent
サービスプロパティの設定 [73](#)
Discovery Agent アプリケーション [26](#)
Discovery Agent アプリケーションのプロパティ [26](#)
GitRepoConnectApp サービスの概要 [37](#)
GitRepoConnectApp の設定プロパティ [38](#)
エージェント名の変更 [73](#)
エラスティックサーバーサービスの概要 [34](#)
エラスティックサーバーの構成プロパティ [34](#)
カスタム構成のプロパティ [73](#)
サービスの概要 [8](#)
データ統合サーバーサービスの概要 [31](#)
データ統合サーバーの設定プロパティ [32](#)
ネットワーク中断設定 [31](#)
ファイル取り込みの設定プロパティ [40](#)
マスキング設定のプロパティ [73](#)
メタデータ基盤アプリケーション [43](#)
メタデータ基盤アプリケーションのプロパティ [43](#)
共通統合コンポーネントプロパティ [18](#)
Secure Agent サービス
CMI ストリーミングエージェント [15](#)
データベース取り込みエージェントの環境変数 [25](#)
データベース取り込みサービスのプロパティ [21](#)

W

Web サイト [6](#)

あ

アップグレード通知 [7](#)

え

エラスティックサーバー
概要 [34](#)

か

カスタム構成のプロパティ
Secure Agent [73](#)

し

システムステータス [7](#)

す

ステータス
Informatica Intelligent Cloud Services [7](#)
ストリーミング取り込み
Secure Agent [15](#)

そ

ソース管理
ローカルリポジトリディレクトリの設定 [37](#)

て

データ統合サーバー
概要 [31](#)

ふ

ファイル統合サービス [36](#)

め

メンテナンスの停止 [7](#)