Server Automation

ソフトウェアバージョン: 10.50

主要コンセプトガイド

ドキュメントリリース日: 2016年7月（英語版）
ソフトウェアリリース日: 2016年7月

Hewlett Packard Enterprise

ご注意

保証

Hewlett Packard Enterprise製品、またはサービスの保証は、当該製品、およびサービスに付随する明示的な保証文によってのみ規定されるものとします。ここでの記載は、追加保証を提供するものではありません。ここに含まれる技術的、編集上の誤り、または欠如について、Hewlett Packard Enterpriseはいかなる責任も負いません。

ここに記載する情報は、予告なしに変更されることがあります。

権利の制限

機密性のあるコンピューターソフトウェアです。これらを所有、使用、または複製するには、Hewlett Packard Enterpriseからの有効な使用許諾が必要です。商用コンピューターソフトウェア、コンピューターソフトウェアに関する文書類、および商用アイテムの技術データは、FAR12.211および12.212の規定に従い、ベンダーの標準商用ライセンスに基づいて米国政府に使用許諾が付与されます。

著作権について

商標について

Adobe®は、Adobe Systems Incorporated (アドビシステムズ社) の登録商標です。

Microsoft®およびWindows®は、Microsoft Corporationの米国における登録商標です。

UNIX®は、The Open Groupの登録商標です。

ドキュメントの更新情報

このマニュアルの表紙には、以下の識別情報が記載されています。

- ソフトウェアバージョン番号: ソフトウェアバージョンを示します。
- ドキュメントリリース日: ドキュメントが更新されるたびに変更されます。
- ソフトウェアリリース日: このソフトウェアバージョンのリリース日を示します。

更新状況、およびご使用のドキュメントが最新版かどうかは、次のサイトで確認できます。https://softwaresupport.hpe.com/

このサイトを利用するには、HPE Passportへの登録とサインインが必要です。HPE Passport IDの登録は、HPEソフトウェアサポートサイトで **[Register]** をクリックするか、HPE Passportのログインページで **[Create an Account]** をクリックしてください。

適切な製品サポートサービスをお申し込みいただいたお客様は、更新版または最新版をご入手いただけます。詳細は、HPEの営業担当にお問い合わせください。

サポート

HPEソフトウェアサポートサイトを参照してください。https://softwaresupport.hpe.com

このサイトでは、HPEのお客様窓口のほか、HPEソフトウェアが提供する製品、サービス、およびサポートに関する詳細情報をご覧いただけます。

HPEソフトウェアオンラインではセルフソルブ機能を提供しています。お客様のビジネスを管理するのに必要な対話型の技術サポートツールに、素早く効率的にアクセスできます。HPEソフトウェアサポートのWebサイトでは、次のようなことができます。

- 関心のあるナレッジドキュメントの検索
- サポートケースの登録とエンハンスメント要求のトラッキング
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HPEサポート窓口の検索
- 利用可能なサービスに関する情報の閲覧
- 他のソフトウェアカスタマーとの意見交換
- ソフトウェアトレーニングの検索と登録

一部のサポートを除き、サポートのご利用には、HPE Passportユーザーとしてご登録の上、サインインしていただく必要があります。また、多くのサポートのご利用には、サポート契約が必要です。HPE Passport IDを登録するには、HPEサポートサイトで **[Register]** をクリックするか、HPE Passportのログインページで **[Create an Account]** をクリックします。

アクセスレベルの詳細については、次のWebサイトをご覧ください。https://softwaresupport.hpe.com/web/softwaresupport/access-levels

HPE Software Solutions Nowでは、HPESWソリューションおよび統合ポータルWebサイトにアクセスできます。このサイトでは、お客様のビジネスニーズに合ったHPE製品ソリューションをご覧いただけます。また、HPE製品間の統合リストとITILプロセスのリストも用意しています。このWebサイトのURLはhttps://softwaresupport.hpe.com/です。

目次

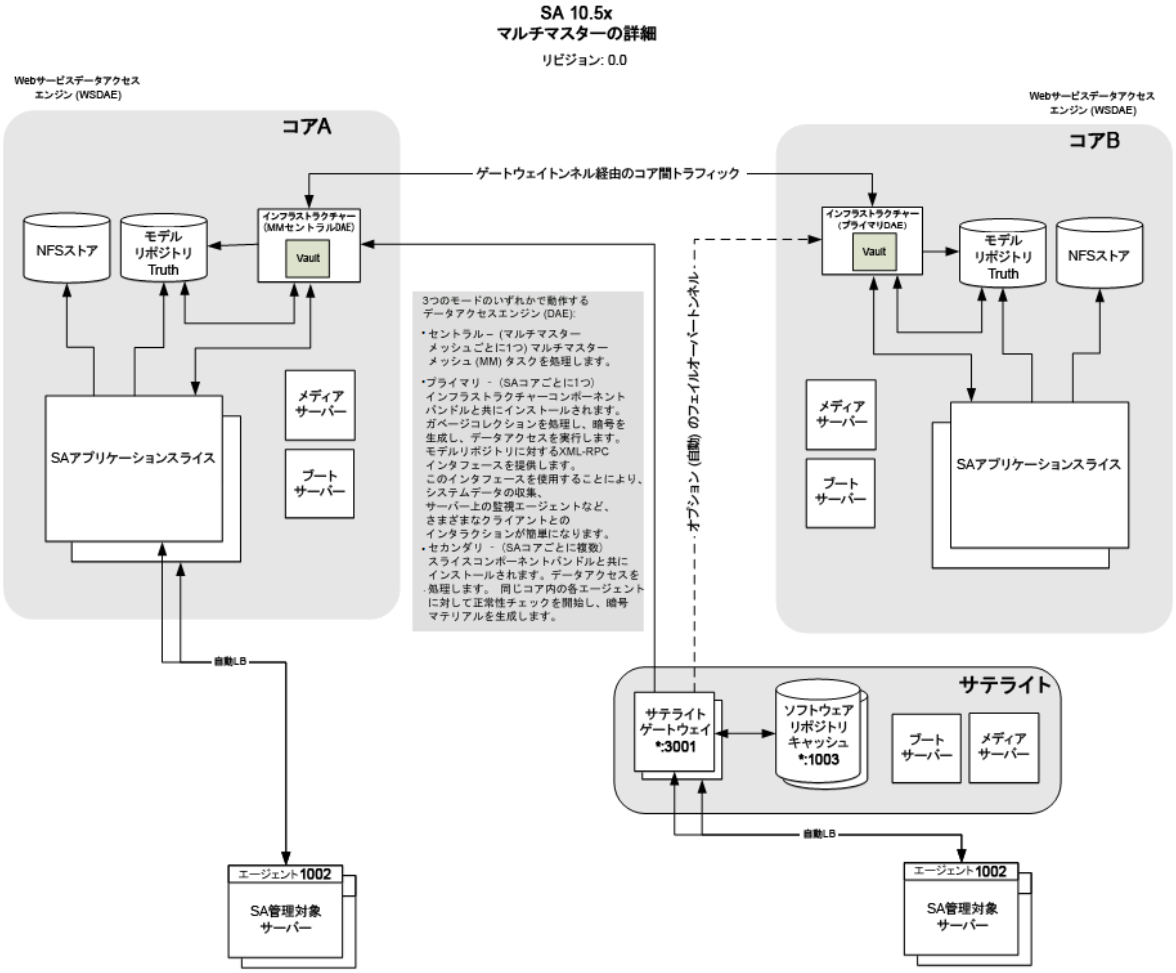
主要コンセプト

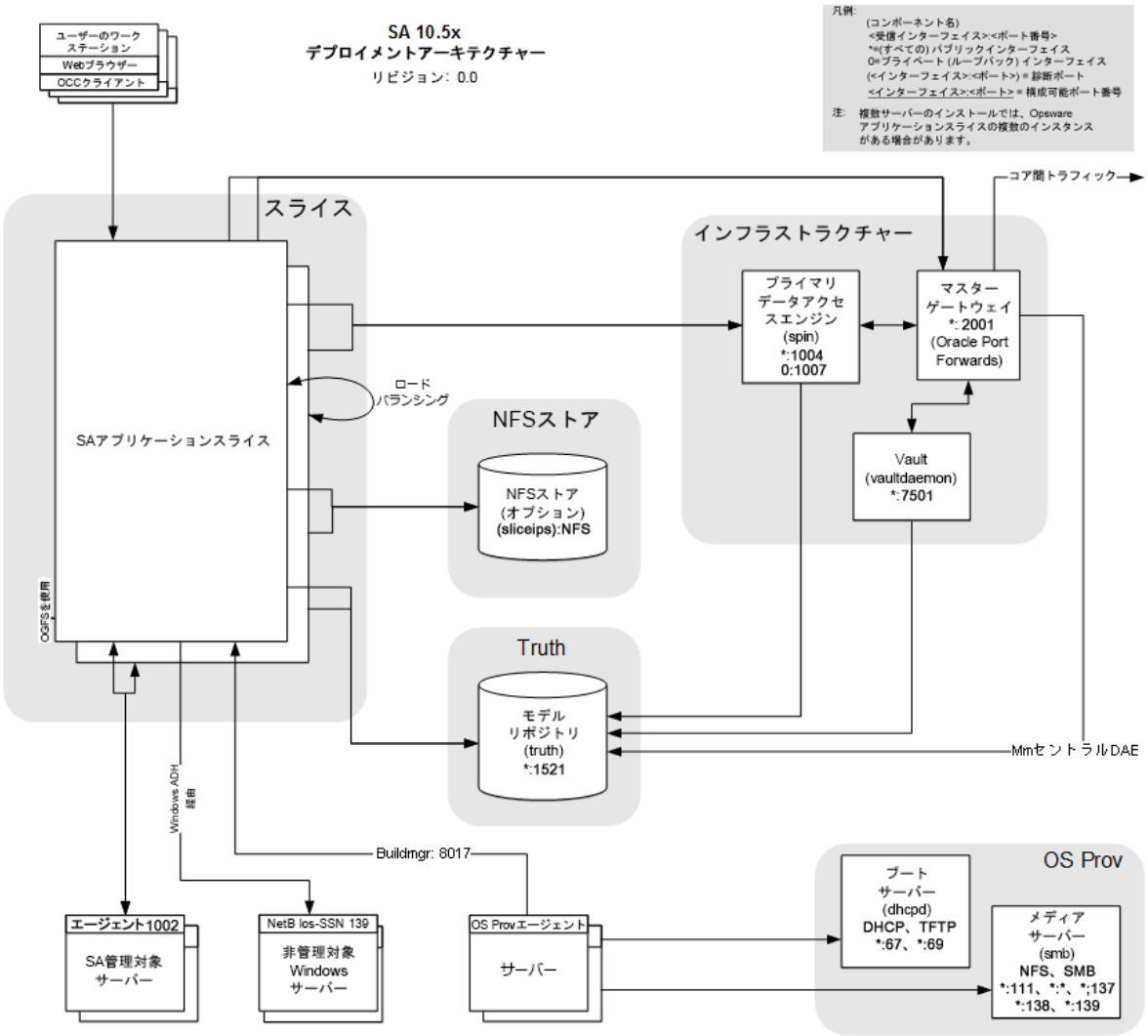
Server Automation (SA) は、データセンター自動化ソフトウェアであり、さまざまなデータセンター機能の一元管理と合理化、データセンターで実施されるサーバー管理の中で重要な分野の自動化を行います。

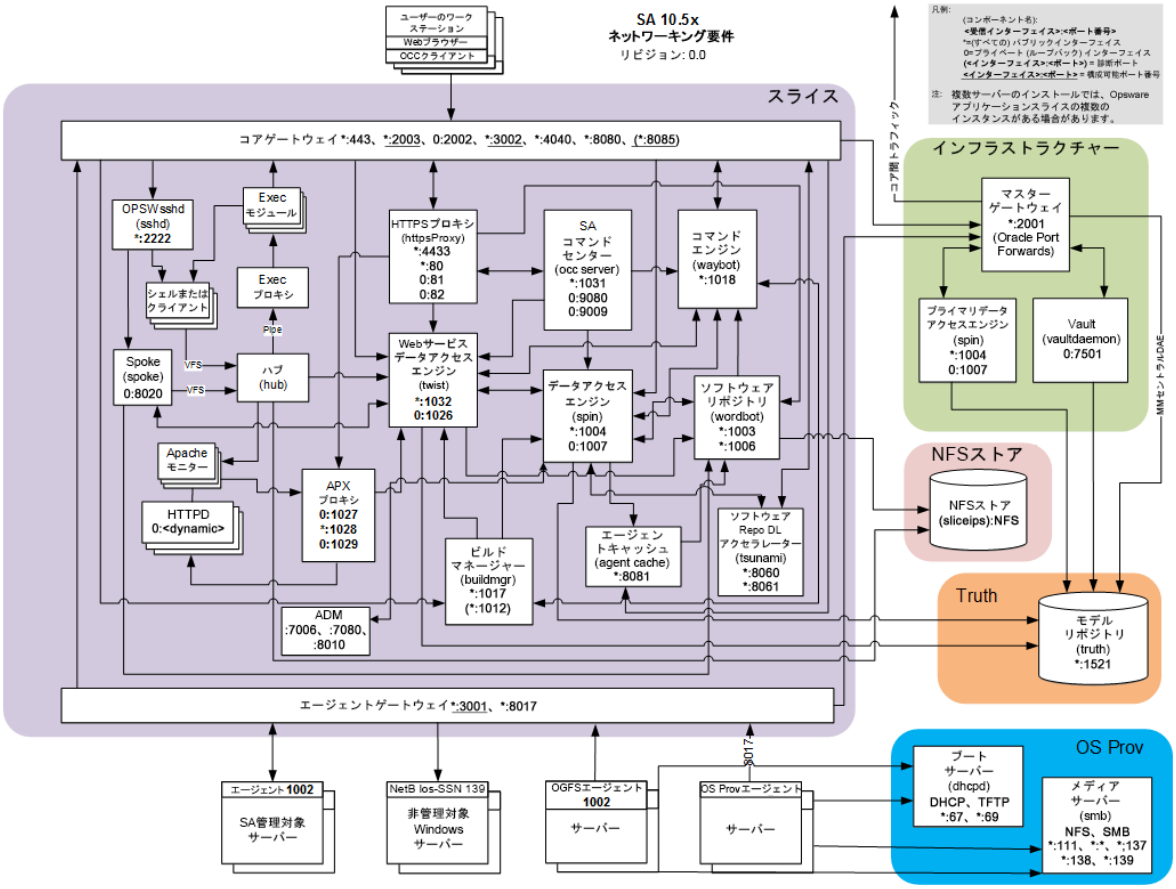
以下のトピックをもとにSAの詳細について説明します。

アーキテクチャー

ここでは、SAコアのレイアウトのカスタマイズ、マルチマスターメッシュの作成、データベースのリモートインストールなどを予定しているユーザーを対象に、SAアーキテクチャーについてさらに詳しく説明します。SAコアとコアコンポーネントの詳細と、コア、サーバーエージェント、サテライト間の関係についても説明します。




SA 10.5x
デプロイメントアーキテクチャー
リビジョン: 0.0
凡例:
(コンポーネント名)
<受信インターフェイス>:<ポート番号>
*=(すべての) パブリックインターフェイス
0=プライベート (ループバック) インターフェイス
(<インターフェイス>:<ポート>) = 診断ポート
<インターフェイス>:<ポート> = 構成可能ポート番号
注: 複数サーバーのインストールでは、Opsware アプリケーションスライスの複数のインスタンスがある場合があります。
ユーザーのワークステーション
Webブラウザー
OCCクライアント
スライス
SAアプリケーションスライス
OGFSを使用
ロードバランシング
インフラストラクチャー
プライマリデータアクセスエンジン (spin) *:1004 0:1007
マスターゲートウェイ *:2001 (Oracle Port Forwards)
Vault (vaultdaemon) *:7501
コア間トラフィック
NFSストア
NFSストア (オプション) (sliceips):NFS
Truth
モデルリポジトリ (truth) *:1521
Mmセントラル DAE
Windows ADH 経由
Buildmgr: 8017
エージェント1002
SA管理対象サーバー
NetB los-SSN 139
非管理対象 Windows サーバー
OS Provエージェント
サーバー
OS Prov
ブートサーバー (dhcpd) DHCP、TFTP *:67、*:69
メディアサーバー (smb) NFS、SMB *:111、*:*、*:137 *:138、*:139



SAコア

SAコアは、コアコンポーネントのグループであり、連携して動作します。コアによって、ネットワーク上でサーバーを検出し、これを管理対象サーバープールに追加した後、プロビジョニング、構成、パッチの適用、監視、監査、管理などの作業をSAクライアントインタフェースから実行できます。SAクライアントでは、1つのインタフェースからSAのすべての情報と管理機能を操作できます。

コアコンポーネントのインストール先となるサーバーは、コアサーバーと呼ばれます。コアコンポーネントは、複数のホストに分散されている場合でも、1つのSAコアの一部として認識されます。

コアコンポーネントは、1つのホストにインストールまたは複数のホストに分散できますが、一般的なSAインストールではコアコンポーネントバンドルが使用されます。バンドルとは、パフォーマンスと管理効率の向上を目的に、いくつかのコンポーネントをまとめて同じサーバーにインストールする方法です。コンポーネントバンドルの詳細については、SAコアコンポーネントのバンドルを参照してください。

SAでは、いくつかのサーバー管理アクティビティとの通信と管理の目的で、各管理対象サーバーにサーバーエージェントがインストールされます。これにより、管理対象サーバーとの通信を、SAコアコンポーネント

の一部であるゲートウェイ経由で行うことができます。またサーバーエージェントは、SAクライアントからユーザーが入力した内容に基づいて、管理対象サーバーでアクションを実行します。

SAサーバーエージェント

SAサーバーエージェントは、SAで管理するすべてのサーバーにインストールされる高機能ソフトウェアです。エージェントレスサーバーにエージェントをインストールすると、エージェントはサーバーをSAに登録し、これによってサーバーは管理対象サーバープールに追加されます。またSAエージェントは、ユーザーが入力したコマンドをコアから受信し、エージェントがインストールされているサーバーで適切なアクションを実行します。このアクションには、ソフトウェアのインストールと削除、ソフトウェアとハードウェアの構成、サーバーステータスのレポート作成、監査などがあります。

SAエージェントは、次の方法でサーバーにインストールできます。

SAエージェントデプロイメントツール(ADT)を使用して、SAサーバーエージェントがインストールされていないサーバー(エージェントレスサーバー)をネットワーク上で検出し、エージェントをインストールします。ADTの詳細については、『SA 10.50ユーザーガイド』を参照してください。

SAプロビジョニングを使用して、ベアボーンサーバーでオペレーティングシステムのプロビジョニングを実行します。これにより、SAサーバーエージェントはオペレーティングシステムと一緒にインストールされます。『SA 10.50管理ガイド』を参照してください。

SAサーバーエージェントのバイナリをサーバーにコピーし、手動でインストールします。『SA 10.50ユーザーガイド』を参照してください。

エージェントの登録では、SAは各サーバーに一意のID(マシンID(MID))を割り当て、このIDをモデルリポジトリに保存します。また、サーバーはMACアドレスでも一意に識別できます。MACアドレスは、ネットワークインタフェースカードに割り当てられている一意の16進数であり、ネットワーク上でのデバイスの物理アドレスとして使用されます。

コアコンポーネント

コアコンポーネントはSAコアの中核であり、サーバーの監視と管理を可能にします。ネットワークサーバーに関する重要な情報の取得、サーバーのプロビジョニング、パッチの適用、サーバーのオンラインまたはオフライン切り替え、サーバーの構成と監査などの操作をユーザーが行う場合、この操作はコアコンポーネントによって制御されます。

次の項では、SAコアコンポーネントとインタフェースについて詳しく説明します。SAコンポーネントが相互に連携してサーバーを管理するしくみについては、『SA 10.50管理ガイド』を参照してください。

モデルリポジトリ

モデルリポジトリには、SAが提供するOracleデータベース、またはSAデータベースの要件を満たす既存のOracleインストールが必要です。詳しい要件については、『SA 10.50インストールガイド』を参照してください。

モデルリポジトリはスタンドアロンのコンポーネントであり、他のコアコンポーネントとはバンドルされません。すべてのSAコンポーネントは、すべてのSA管理対象サーバーについて保持されているデータモデルを使用し、更新します。モデルリポジトリには、次の情報が保存されています。

- SAで管理するすべてのサーバーのインベントリ。
- サーバーに関連付けられているハードウェアのインベントリ(メモリ、CPU、ストレージ容量など)。
- 管理対象サーバーの構成情報。
- 管理対象サーバーにインストールされているオペレーティングシステム、システムソフトウェア、アプリケーションのインベントリ。
- SAプロビジョニングで使用するオペレーティングシステムインストールメディアのインベントリ(メディア本体はSAプロビジョニングメディアサーバーに保存されます)。
- インストール可能なソフトウェアと、ソフトウェアの構成およびインストール方法を制御するソフトウェアポリシーのインベントリ。ソフトウェアインストールメディア本体は、ソフトウェアリポジトリに保存されます。
- 認証とセキュリティの情報。

SAコアコンポーネントのバンドル

一部のSAコアコンポーネントはまとめてバンドル化し、標準インストールで1つの単位としてインストールする必要があります。また、バンドルに含まれる一部のコンポーネント (特に、リポジトリストア、SAプロビジョニングメディアサーバー) は、カスタムインストールを実行して個別に別のホストにインストールすることが可能です。ただし、分散コアコンポーネントなど複雑なインストールにはHPEプロフェッショナルサービスやHPE認定コンサルタントのサポートが必要になるので、カスタマーインストールではサポートされません。

次の表は、SAコンポーネントバンドルと構成コンポーネントの例を示します。スライスコンポーネントバンドルには、インスタンスを複数インストールし、負荷分散を実行できます。

コンポーネントの区分

モデルリポジトリ	インフラストラクチャーコンポーネント	SAプロビジョニングコンポーネント	スライスコンポーネント#1	スライスコンポーネント#x
コアあたり1つ	コアあたり1つ	通常はコアあたり1つ	コアあたり1つ	コアあたり複数
モデルリポジトリ	管理ゲートウェイ	メディアサーバー	コアゲートウェイ/	コアゲートウェイ/

モデルリポジトリ	インフラストラクチャーコンポーネント	SAプロビジョニングコンポーネント	スライスコンポーネント#1	スライスコンポーネント#x
	プライマリデータアクセスエンジン モデルリポジトリマルチマスターコンポーネント ソフトウェアリポジトリストア(別ホストに配置可能)	(OSシーケンスのみ) ブートサーバー(OSシーケンスのみ)	エージェントゲートウェイ コマンドセンター Global File System Webサービスデータアクセスエンジン セカンダリデータアクセスエンジン Build Manager コマンドエンジン ソフトウェアリポジトリ HPE Live Network (HPELN) DCML Exchange Tool (DET) ソフトウェアリポジトリアクセラレーター (`tsunami`) Memcache	エージェントゲートウェイ コマンドセンター Global File System Webサービスデータアクセスエンジン セカンダリデータアクセスエンジン Build Manager コマンドエンジン ソフトウェアリポジトリ HPE Live Network (HPELN) DCML Exchange Tool (DET) ソフトウェアリポジトリアクセラレーター (`tsunami`) Memcache

SAコアコンポーネントのバンドルには、次のような利点があります。

- 複数サーバーのデプロイメントがシンプルになり、堅牢性も向上します。
- スライスコンポーネントバンドルを追加インストールすることで、水平方向の拡張が可能になります。
- 可用性が向上します。
- 複数のインスタンスをインストールすることで、スライス間で負荷を分散できます。

SAコアコンポーネントのアーキテクチャーとインタラクションの詳細については、『SA 10.50主要コンセプトガイド』を参照してください。

ブートエージェントは、サーバーエージェントではなく、SAプロビジョニングの一部として機能します。

コアコンポーネントのバンドル

インフラストラクチャーコンポーネントバンドル

- **プライマリデータアクセスエンジン**
  データアクセスエンジンは、モデルリポジトリに対するXML-RPCインタフェースを提供します。このインタフェースを使用することにより、クライアントとのインタラクション、システムデータの収集、サーバー上の監視エージェントを簡単に実行できます。インフラストラクチャーコンポーネントバンドルでインストールされたデータアクセスエンジンは、プライマリデータアクセスエンジンとなります。スライスコンポーネントバンドルでインストールされたデータアクセスエンジンは、セカンダリデータアクセスエンジンとなります。

  モデルリポジトリとのインタラクションはデータアクセスエンジンを経由するので、モデルリポジトリのスキーマに変更を加えてもクライアントへの影響を小さく抑えることができます。また、データアクセスエンジンには、システム全体を変更しなくてもSAに機能を追加できるという利点もあります。

- **管理ゲートウェイ**
  他のSAコアやサテライトとの通信を管理します。

- **モデルリポジトリ マルチマスターコンポーネント**
  モデルリポジトリのマルチマスターコンポーネントは、インフラストラクチャーコンポーネントバンドルと一緒にインストールされます。マルチマスターメッシュにはコアインストールが複数含まれます。したがって、モデルリポジトリのマルチマスターコンポーネントは、モデルリポジトリ内のデータをメッシュ内のすべてのコアで同期し、1つのリポジトリで行った変更内容を他のリポジトリに反映します。

  モデルリポジトリのマルチマスターコンポーネントには、それぞれ送信コンポーネントと受信コンポーネントが存在します。送信コンポーネント (送信モデルリポジトリマルチマスターコンポーネント) は、モデルリポジトリをポーリングし、未発行のトランザクションがあると他のモデルリポジトリに送信します。受信コンポーネント (受信モデルリポジトリマルチマスターコンポーネント) は、他のモデルリポジトリから送信されたトランザクションを受信します。

- **ソフトウェアリポジトリストア**
  ソフトウェアリポジトリストアコンポーネントは、インフラストラクチャーコンポーネントバンドルをホストする任意のサーバーにインストールできます。SA 10.50で、ソフトウェアリポジトリはスライスコンポーネントバンドルの一部となり、スライスコンポーネントバンドルのホストに対するNFSエクスポートを処理するコンポーネントとして、ソフトウェアリポジトリストアが追加されています。

  ソフトウェアリポジトリストアをインストールしない場合、スライスコンポーネントバンドルサーバーがファイルシステムにアクセスするには、NAS (ファイラー) の構成を手動で行う必要があります。

  スライスコンポーネントバンドル

- **コマンドエンジン**
  スライスコンポーネントバンドルの一部です。コマンドエンジンは、複数のサーバーに分散したプログラムを実行するシステムです(一般的に、SAサーバーエージェントが使用されます)。コマンドエンジンスクリプトはPythonで記述され、コマンドエンジンサーバーで実行されます。コマンドエンジンスクリプトでは、サーバーエージェントにコマンドを発行することができます。これらの要求は安全に配信され、モデルリポジトリに保存されているデータを使用して監査できます。

  スライスコンポーネントバンドルを複数インストールすると、コマンドエンジンも複数使用できるので、水平方向の拡張性が大幅に向上します。コマンドエンジンインスタンスが複数ある場合、スライスコンポーネントバンドルが備えるロードバランシング機能を使ってコマンド送信とスクリプト実行に伴う負荷を分散できます。また、フェイルオーバーと高可用性の機能も向上します。たとえば、コマンドエンジンインスタンスがクラスター内の別ノードにコマンド処理を切り替えようとしたとき、そのノードがダウン状態の場合には次のノードにフェイルオーバーできます。

  SAでは、コマンドエンジンのスクリプトを使用して機能を実装します。

- **ソフトウェアリポジトリ**
  スライスコンポーネントバンドルの一部です。このリポジトリには、ソフトウェア/アプリケーションのプロビジョニングと修復を行うバイナリ/パッケージ/ソースをアップロードして保存しておきます。関連コンポーネントの1つにソフトウェアリポジトリストアがあります。このストアはインフラストラクチャーコンポーネントバンドルと一緒にインストールされ、スライスコンポーネントバンドルのホストに対するNFSエクスポートを処理します。

  SAは、ソフトウェアリポジトリのミラーリングをサポートします。メッシュ内でミラーとして使用するソフトウェアリポジトリと、ジョブをミラーリングする頻度は、SAクライアントの構成パラメーターで制御できます。また、ミラーリングによって、サテライトのソフトウェアリポジトリキャッシュが影響を受けることはありません。

  ソフトウェアリポジトリミラーリングには、大量のディスク容量が必要になります。ミラーリングは、標準インストール時と高度なインストール時に無効にすることができます(デフォルトでは有効)。

  ソフトウェアリポジトリミラーリングの構成については、『SA 10.50管理ガイド』を参照してください。

  ソフトウェアパッケージをSAライブラリにアップロードする方法については、『SA 10.50管理ガイド』を参照してください。

- **コアゲートウェイ/エージェントゲートウェイ**
  コアゲートウェイは、エージェントゲートウェイと直接通信することにより、コアコンポーネントとの間で要求の送信と応答を行います。

- **コマンドセンター**
  コマンドセンター(OCC)は、SAクライアントの基盤となるコアコンポーネントです。OCCには、HTTPSプロキシサーバーとアプリケーションサーバーが含まれます。OCCには、SAクライアントを経由しないとアクセスできません。

- **DCML Exchange Tool**
  DCML Exchange Toolはスライスコンポーネントバンドルと一緒にインストールされるツールであり、SAコンテンツのインポートとエクスポートを実行します。『SA 10.50管理ガイド』を参照してください。

- **Global File System**
  Global File System (OGFS) は、スライスコンポーネントバンドルと一緒にインストールされ、一元的な実行環境をSAに提供します。

  OGFSは、1つまたは複数の物理サーバーで実行できます。したがって、スライスコンポーネントバンドルをコアに追加していくだけで、SAの実行容量を拡張することが可能になります。

  OGFSは、SAビルトインコンポーネントとユーザーが記述したプログラムの両方を仮想ファイルシステム内で実行します。仮想システムは、SAデータモデル、SAアクション、管理対象サーバーを仮想のファイルとディレクトリとして提示します。

  これはSA固有の機能であり、Global Shellと自動化プラットフォーム拡張 (APX) のユーザーは、任意のスクリプト言語やプログラミング言語を使用してSAデータのクエリ実行やサーバー管理を実行できます。OGFSはすべてのデータ、アクション、管理対象サーバーのアクセスをSAセキュリティモデルを使ってフィルター処理するので、OGFSで実行するプログラムのセキュリティはデフォルトで確保されます。

- **Webサービスデータアクセスエンジン**
  Webサービスデータアクセスエンジンは、モデルリポジトリに対するパブリックオブジェクトの抽象化レイヤーを提供し、これによって他のコアコンポーネントのパフォーマンスを向上します。このオブジェクト抽象化には、Simple Object Access Protocol (SOAP) API、サードパーティの統合コンポーネント、コンポーネント (SAクライアントなど) のバイナリプロトコルでアクセスできます。

- **セカンダリデータアクセスエンジン**
  データアクセスエンジンは、モデルリポジトリに対するXML-RPCインタフェースを提供します。このインタフェースを使用することにより、クライアントとのインタラクション、システムデータの収集、サーバー上の監視エージェントを簡単に実行できます。インフラストラクチャーコンポーネントバンドルでインストールされたデータアクセスエンジンは、プライマリデータアクセスエンジンとなります。スライスコンポーネントバンドルでインストールされたデータアクセスエンジンは、セカンダリデータアクセスエンジンとなります。

  モデルリポジトリとのインタラクションはデータアクセスエンジンを経由するので、モデルリポジトリのスキーマに変更を加えてもクライアントへの影響を小さく抑えることができます。また、データアクセスエンジンには、システム全体を変更しなくてもSAに機能を追加できるという利点もあります。

- **Build Manager**
  (OSシーケンスのみ) Build ManagerはSAプロビジョニングの一部ですが、スライスコンポーネントバンドルと一緒にインストールされます。Build ManagerはOSビルドエージェントとコマンドエンジンの間の通信をサポートする機能を持ち、コマンドエンジンが送信したSAプロビジョニングコマンドを受信しま

す。また、SAプロビジョニング手順を実行できるように、プラットフォーム固有のビルドスクリプトの実行時環境を提供します。

- **HPE Live Network (HPELN)**
  HPE Live Networkは、Server Automation (SA)、Network Automation (NA)、Client Automation (CA)、Operations Orchestration (OO)、Service Automation Reporter (SAR) で提供されたコンテンツ更新を配信します。また、セキュリティポリシーとコンプライアンスポリシーを提供することを通じて、SA、NA、CAへの投資効果を最大限に引き出し、拡張可能な自動化プラットフォームで新しい自動化機能を継続的に提供します。

  HPELNは、SAコアのインストール時にスライスコンポーネントバンドルと一緒にインストールされます。

- **ソフトウェアリポジトリアクセラレーター** (tsunami)
  オブジェクトストアのダウンロードアクセラレーターであり、LinuxベースのSAコアと直接通信するエージェントの修復パフォーマンスと拡張性を向上します。

  パフォーマンスと拡張性は、主に次の2つの部分で向上します。

  RPM修復分析: RPM依存関係の分析/プレビューでパッケージヘッダーを取得します。SAの旧リリースよりプレビューが格段に高速になっています。

  修復パッケージのステージング: ソフトウェアリポジトリから管理対象ホストをダウンロードする処理が、SAの旧リリースより格段に高速になり、10GbEネットワーキングにも対応しています。

- **memcache**
  メモリ内のキャッシュレイヤーであり、ソフトウェアリポジトリアクセラレーター (`tsunami`) コンポーネントと連携して、LinuxベースのSAコアと直接通信するエージェントでの修復と拡張性を向上します。

SAプロビジョニングコンポーネントバンドル

- **ブートサーバー**
  ブートサーバーは、プロビジョニングの一部です。inetbootよりSunシステムの、PXEよりx86システムのネットワークブートをサポートします。このサポートを提供するプロセスには、Internet Software ConsortiumのDHCPサーバーが使われています。
- **メディアサーバー**
  メディアサーバーは、プロビジョニングの一部です。SAプロビジョニング時に、ベンダーが提供するメディアへのネットワークアクセスをサポートします。このサポートを提供するプロセスには、Samba SMBサーバーとLinux NFSが含まれます。有効なオペレーティングシステムインストールメディアをメディアサーバーにコピーおよびアップロードします。

OSビルドエージェントは、SAプロビジョニングの一部です。プレプロビジョニング (ネットワークブート) プロセスで実行され、Build Managerを使用してサーバーをSAコアに登録し、OSインストールプロセスを実行します。

- **コアゲートウェイ/エージェントゲートウェイ**

  コアゲートウェイは、エージェントゲートウェイと直接通信することにより、コアコンポーネントとの間で要求の送信と応答を行います。

- **DCML Exchange Tool**

  DCML Exchange Toolはスライスコンポーネントバンドルと一緒にインストールされるツールであり、SAコンテンツのインポートとエクスポートを実行します。『SA 10.50管理ガイド』を参照してください。

- **Global File System**

  Global File System (OGFS) は、スライスコンポーネントバンドルと一緒にインストールされ、一元的な実行環境をSAに提供します。

  OGFSは、1つまたは複数の物理サーバーで実行できます。したがって、スライスコンポーネントバンドルをコアに追加していくだけで、SAの実行容量を拡張することが可能になります。

  OGFSは、SAビルトインコンポーネントとユーザーが記述したプログラムの両方を仮想ファイルシステム内で実行します。仮想システムは、SAデータモデル、SAアクション、管理対象サーバーを仮想のファイルとディレクトリとして提示します。

  これはSA固有の機能であり、Global Shellと自動化プラットフォーム拡張 (APX) のユーザーは、任意のスクリプト言語やプログラミング言語を使用してSAデータのクエリ実行やサーバー管理を実行できます。OGFSはすべてのデータ、アクション、管理対象サーバーのアクセスをSAセキュリティモデルを使ってフィルター処理するので、OGFSで実行するプログラムのセキュリティはデフォルトで確保されます。

- **Webサービスデータアクセスエンジン**

  Webサービスデータアクセスエンジンは、モデルリポジトリに対するパブリックオブジェクトの抽象化レイヤーを提供し、これによって他のコアコンポーネントのパフォーマンスを向上します。このオブジェクト抽象化には、Simple Object Access Protocol (SOAP) API、サードパーティの統合コンポーネントでアクセスできます。

- **セカンダリデータアクセスエンジン**

  データアクセスエンジンは、モデルリポジトリに対するXML-RPCインタフェースを提供します。このインタフェースを使用することにより、クライアントとのインタラクション、システムデータの収集、サーバー上の監視エージェントを簡単に実行できます。インフラストラクチャーコンポーネントバンドルでインストールされたデータアクセスエンジンは、プライマリデータアクセスエンジンとなります。スライスコンポーネントバンドルでインストールされたデータアクセスエンジンは、セカンダリデータアクセスエンジンとなります。

  モデルリポジトリとのインタラクションはデータアクセスエンジンを経由するので、モデルリポジトリのスキーマに変更を加えてもクライアントへの影響を小さく抑えることができます。また、データアクセスエンジンには、システム全体を変更しなくてもSAに機能を追加できるという利点もあります。

- **Build Manager**

(OSシーケンスのみ) Build ManagerはSAプロビジョニングの一部ですが、スライスコンポーネントバンドルと一緒にインストールされます。Build ManagerはOSビルドエージェントとコマンドエンジンの間の通信をサポートする機能を持ち、コマンドエンジンが送信したSAプロビジョニングコマンドを受信します。また、SAプロビジョニング手順を実行できるように、プラットフォーム固有のビルドスクリプトの実行時環境を提供します。

- **HPE Live Network (HPELN)**

  HPE Live Networkは、Server Automation (SA)、Network Automation (NA)、Client Automation (CA)、Operations Orchestration (OO)、Service Automation Reporter (SAR) で提供されたコンテンツ更新を配信します。また、セキュリティポリシーとコンプライアンスポリシーを提供することを通じて、SA、NA、CAへの投資効果を最大限に引き出し、拡張可能な自動化プラットフォームで新しい自動化機能を継続的に提供します。

  HPELNは、SAコアのインストール時にスライスコンポーネントバンドルと一緒にインストールされます。

- **ソフトウェアリポジトリアクセラレーター** (tsunami)

  オブジェクトストアのダウンロードアクセラレーターであり、LinuxベースのSAコアと直接通信するエージェントの修復パフォーマンスと拡張性を向上します。

  パフォーマンスと拡張性は、主に次の2つの部分で向上します。

  RPM修復分析: RPM依存関係の分析/プレビューでパッケージヘッダーを取得します。SAの旧リリースよりプレビューが格段に高速になっています。

  修復パッケージのステージング: ソフトウェアリポジトリから管理対象ホストをダウンロードする処理が、SAの旧リリースより格段に高速になり、10GbEネットワーキングにも対応しています。

- memcache

  メモリ内のキャッシュレイヤーであり、ソフトウェアリポジトリアクセラレーター (`tsunami`) コンポーネントと連携して、LinuxベースのSAコアと直接通信するエージェントでの修復と拡張性を向上します。

SAプロビジョニングコンポーネントバンドル

- **ブートサーバー**

  ブートサーバーは、プロビジョニングの一部です。inetbootよりSunシステムの、PXEよりx86システムのネットワークブートをサポートします。このサポートを提供するプロセスには、Internet Software ConsortiumのDHCPサーバーが使われています。

- **メディアサーバー**

  メディアサーバーは、プロビジョニングの一部です。SAプロビジョニング時に、ベンダーが提供するメディアへのネットワークアクセスをサポートします。このサポートを提供するプロセスには、Samba SMBサーバー

とLinux NFSが含まれます。有効なオペレーティングシステムインストールメディアをメディアサーバーにコピーおよびアップロードします。

OSビルドエージェントは、SAプロビジョニングの一部です。プレプロビジョニング(ネットワークブート)プロセスで実行され、Build Managerを使用してサーバーをSAコアに登録し、OSインストールプロセスを実行します。

サテライトインストール

- **ソフトウェアリポジトリキャッシュ**
  ソフトウェアリポジトリキャッシュには、コアのソフトウェアリポジトリ(または別のサテライト)のコンテンツのローカルコピーが保存されています。ソフトウェアリポジトリのローカルコピーを保存しておくことによって、サテライトの管理対象サーバーにソフトウェアをインストールまたは更新する際、パフォーマンス向上やネットワークトラフィック低減などの効果を期待できます。
- **サテライトエージェントゲートウェイ**
  サテライトエージェントゲートウェイは、サテライトとコア間の通信を、コアの管理ゲートウェイ経由で処理します。

SAゲートウェイ

SAゲートウェイは、管理対象サーバーとSAコアの間の通信、複数コア(マルチマスターメッシュ)間の通信、サテライトインストールとSAコア間の通信を管理します。マルチマスターインストールについては「マルチマスターメッシュ(複数コア)」(21ページ)、サテライトインストールについては「SAサテライト」(25ページ)を参照してください。

ゲートウェイには、次のタイプがあります。

- **管理ゲートウェイ**
  このゲートウェイは、SAコア間の通信と、SAコアとサテライト間の通信を管理します。
- **コアゲートウェイ/エージェントゲートウェイ**
  このゲートウェイは相互に連携して、SAコアと管理対象サーバー上のSAエージェント間の通信を管理します。
- **サテライトゲートウェイ**
  このゲートウェイは、ユーザー構成に応じて、管理ゲートウェイまたはコアゲートウェイを経由してコアと通信します。

マルチマスターのマスターゲートウェイバックアップルート

マルチマスターメッシュでコアを3つ以上インストールすると、第2コアへのバックアップルートが自動作成されます(デフォルト)。これにより、第1コアにはプライマリルート、第2コアにはバックアップルートが確保されます。ゲートウェイバックアップルートは、SAによってインストール時に自動作成されるので、構成情報の指定は必要ありません。ただし、SAがバップアップルートを作成できない場合は、HPEテクニカルサポートへの問い合わせを指示するメッセージが表示され、ゲートウェイバックアップルートの手動作成が必要になることがあります。

ゲートウェイバックアップルートが作成されるのは、SA 10.50の新規インストール時のみで、アップグレードでは作成されません。旧バージョンからSA 10.50へのアップグレードでは、ゲートウェイバックアップルートは作成されません。したがって、手動での作成が必要になります。詳細については、HPEテクニカルサポートに連絡してください。

たとえば、メッシュにコアが3つ以上ある場合、すべてのマルチマスタートラフィックは第1コアのマスターゲートウェイを経由します(デフォルト)。ただし、第1コアのマスターゲートウェイに障害が発生した場合、第2コアのマスターゲートウェイがバックアップのマスターゲートウェイとしてデフォルトで指定されています。これ以降メッシュに追加したコアのマスターゲートウェイは、インストールした順序でバックアップとして指定されます。インストールでは、3番目以降のコアにデフォルトで2つのトンネルが設定されます。最初のトンネルは第1コアのマスターゲートウェイ、2番目のトンネルはメッシュ内の第2コアと通信します(次の図を参照)。

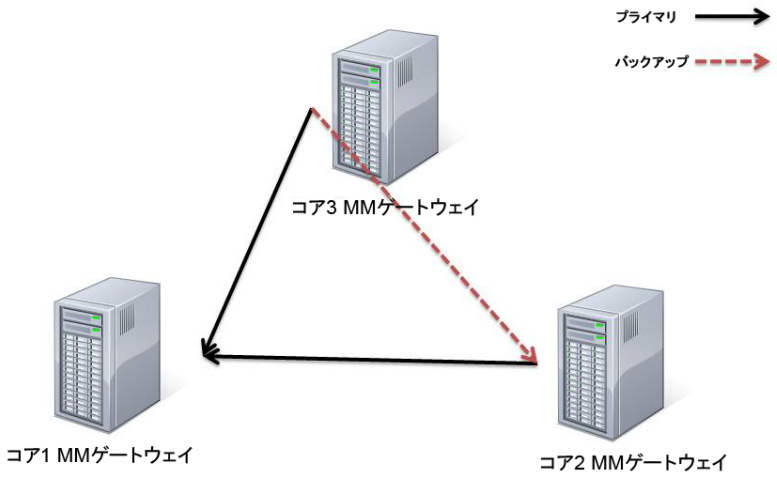


図-2: 推奨セットアップ。ボックスコアが3つ、バックアップルートがある場合

メッシュにマスターゲートウェイが複数あると、バックアップルートも冗長化されます(次の図を参照)。

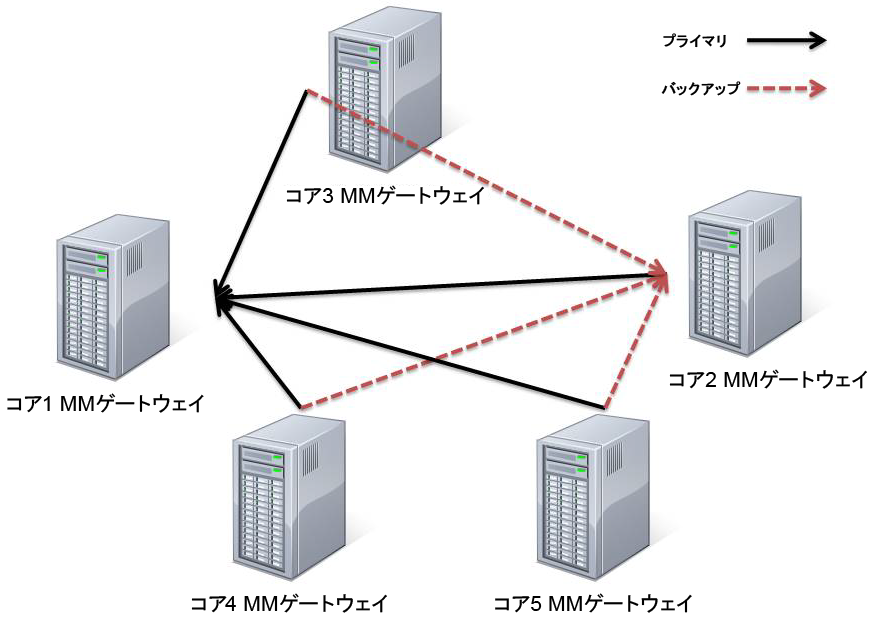


図-3: 推奨セットアップ。ボックスコアが5つ、バックアップルートがある場合

マスターゲートウェイに障害が発生すると、デフォルトで、バックアップルートが自動的にマルチマスターメッシュのトラフィックを処理します。故障したマスターゲートウェイがオンラインに復帰すると、メッシュトラフィックが再びこのゲートウェイを通過するように自動的にルートが戻ります。

SAのトポロジ

SAトポロジは、ファシリティのニーズに最適なものを選ぶ必要があります。ここでは、SAトポロジを選択する際に必要になる内容について説明します。

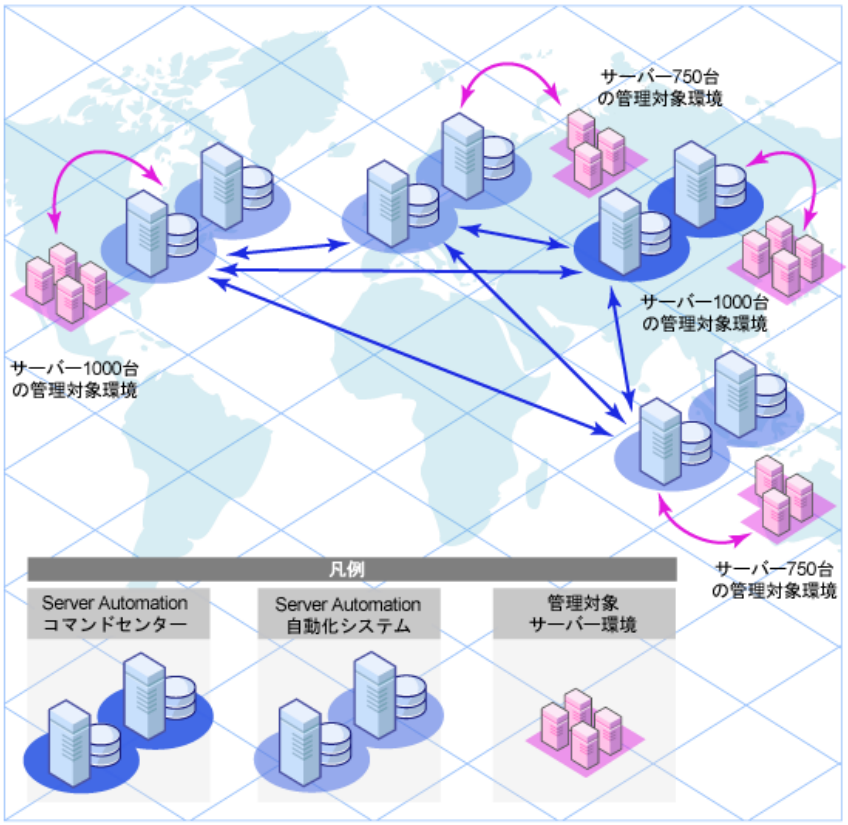
単一ホストコア

単一ホストコア(以前はスタンドアロンコアと呼ばれていました)は最もシンプルなトポロジであり、1つのファシリティでサーバーを管理します。

単一ホストコアは、小規模なサーバーネットワークを1つのファシリティで管理する場合に最適です。単一ホストコアは他のSAコアと通信することはありませんが、必要に応じて、マルチマスターメッシュの一部として構成を変換できます。

マルチマスターメッシュ(複数コア)

複数のファシリティでサーバーを管理するには、SAコアのマルチマスターメッシュをインストールします。これは、複数のSAコアとサテライトを組み合わせたものです。



マルチマスターメッシュとは、管理ゲートウェイを介して相互通信する複数のSAコアの集まりであり、モデルリポジトリに保存されている管理対象サーバー関連データを同期することができます。マルチマスターメッシュ内にあるモデルリポジトリのデータが変更されると、変更内容がメッシュ内にある他のモデルリポジトリにブロードキャストされ、同期されます。

各モデルリポジトリの変更内容を他のすべてのモデルリポジトリに反映して同期するSAコアコンポーネントは、モデルリポジトリマルチマスターコンポーネントと呼ばれ、インフラストラクチャーコンポーネントバンドルに含まれます。このレプリケーション機能を使用することによって、ソフトウェアと環境の属性を含む「ブループリント」を各ファシリティに保存および保持できるので、インフラストラクチャーの再構築も簡単に行うことができます。また、追加容量のプロビジョニングや更新の配信に加え、ソフトウェアビルド、テンプレート、依存関係を複数のファシリティで共有する操作も簡単に実行できます。

マルチマスターメッシュには、サテライトインストールも含まれます。

サーバーは、SAコアをインストールした任意のファシリティから、またはSAクライアントを使用して管理できます。

マルチマスターメッシュの利点

マルチマスターメッシュには、次のような利点があります。

- **一元管理**: マルチマスターメッシュ内の管理対象サーバーは、メッシュ内にあるSAコアで管理する任意のファシリティから一元管理できます。管理できる範囲は1箇所に限定されることはなく、地理的な制約もありません。
- **冗長化**: データ管理をファシリティ間で同期 (レプリケーション) することで、冗長化を図ることができます。たとえば、メッシュ内のあるファシリティに含まれるSAコアに障害が発生した場合、マルチマスターメッシュ内の他のコアに管理対象サーバーのコピーデータが保存されているので、これを元に、破損したコアのモデルリポジトリを前回の正常稼働状態に復元することができます。さらに、破損したコアが使用できなくなる間、メッシュ内の他のコアは中断することなく稼働を継続できます。
  また、レプリケーションによって、メッシュ内にあるファシリティの稼働を中断することなく、ファシリティの停止や追加を行うことができます。
- **パフォーマンスの拡張**: マルチマスターメッシュでは、マルチマスターデータベースの同期データのみがネットワーク経由で転送されるので、ネットワーク帯域幅にかかる負荷を軽減できます。
- **地理的な条件に依存しない**: ネットワークが中断しても、場所に関係なくコアはサーバー管理を継続できます。

ファシリティとレルム

SAゲートウェイは、ネットワークトラフィックのルーティングを行うコンストラクトと、IPアドレスの競合を回避するコンストラクトを使用します。

ファシリティ

ファシリティとは、モデルリポジトリに保存されている管理対象環境のデータに基づいて、1つのSAコアが管理するサーバーの集まりを表すコンストラクトです。一般的にファシリティは、Sunnyvale、San Francisco、New Yorkなどの地理的な場所や、特定のデータセンターを表します。

ファシリティはSA内でのアクセス権の境界であり、ユーザーは1つのファシリティ内でのアクセス権を別のファシリティで適用することはできません。管理対象サーバーはそれぞれが1つのファシリティに割り当てられます。デバイスは、最初にSAコアに登録する際、登録で使用するゲートウェイに関連付けられているファシリティに割り当てられます。

たとえば、Admin AというユーザーがSunnyvaleでサーバーパッチの管理を担当しているとします。ファシリティという枠組みの中では、Admin AはSunnyvaleというファシリティにユーザーとして割り当てられます。

Admin Aがサーバーを表示すると、Sunnyvaleに割り当てられているサーバーのみが表示され、他のファシリティのサーバーは表示されません。

ファシリティには、次の2つのタイプがあります。

- **コアファシリティ**: 各コアには、コアファシリティが1つ存在します。
- **サテライトファシリティ**: サテライトをインストールするときに作成されるデフォルトのファシリティです。

レルム

レルムとはSAコンストラクトの1つであり、これによってSAは、1つのファシリティ内にある異なるネットワーク上のサーバーを、IPアドレスが競合している場合でも管理できます。レルムとは、ファシリティのネットワーク内にあるデバイスのIPアドレスに付加される一意のIDです。これに基づいて、SAゲートウェイはマルチマスターメッシュ内の各ネットワーク上にあるデバイスを一意に識別するので、IPアドレスが競合している場合にも対応できます。

レルムはIP名前空間を定義する論理エンティティであり、この名前空間内では管理対象サーバーのIPアドレスはすべて一意である必要があります。ただし、サーバーのレルムが異なるとIPアドレスが重複する可能性がありますが、レルムのメンバーシップによってSA内では一意に識別されます。

レルムは、ゲートウェイによって相互接続されてゲートウェイメッシュを構成し、これは相互接続された単一のSAゲートウェイネットワークとして機能します。

インストール時に新しいファシリティを作成して名前を付けると、ファシリティと同じ名前でレルムが作成されます (デフォルト)。たとえば、Datacenterという名前のファシリティを作成すると、Datacenterという名前のレルムも作成されます。このファシリティでさらに続けてレルムを作成すると、Datacenter001、Datacenter002、などの名前が割り当てられます。各レルム内の管理対象サーバーは、レルムの名前とIPアドレスの組み合わせによって一意に識別されます。

メッシュ内の接続には、接続がメッシュに入る入力 (ソース) レルムと、接続がメッシュから出る出力 (ターゲット) レルムがあります。

すべてのSA管理対象サーバーは、ただ1つのレルムに割り当てられます。割り当て先のレルムは動的に変わることがあります (ただし、通常は変わりません)。コアは、デバイスがコアに登録するたびに、ゲートウェイIDサーバーを使用して登録先の入力レルムを検索し、見つかったレルムにデバイスを割り当てます。

直接接続による登録の場合は、デバイスは暫定レルムに割り当てられます。暫定レルムは、レルムとして捉える必要はなく、デバイスが直接接続でコアに登録していることを示す目的でのみ存在します。

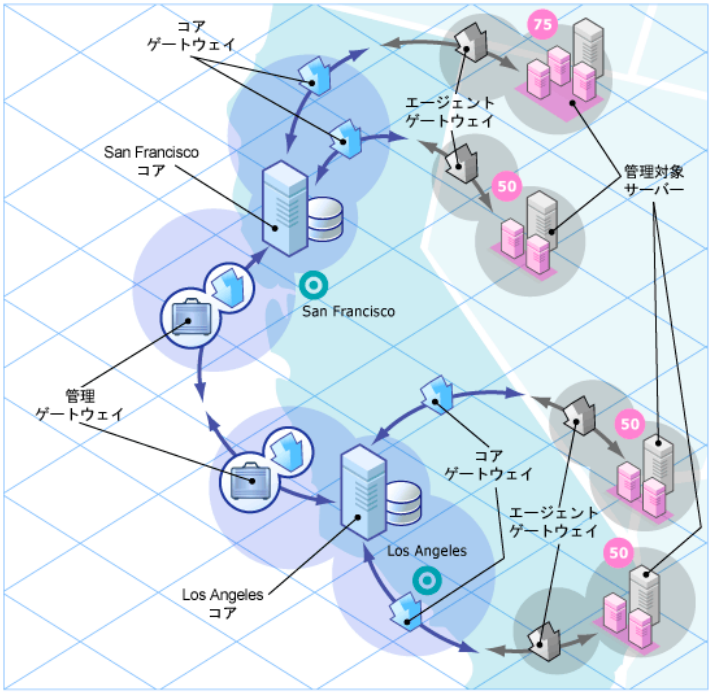
暫定レルムを除き、レルムは複数のファシリティにまたがることはありません (つまり、暫定レルムを除き、個々のレルムはただ1つのファシリティに関係付けられています)。

メッシュ内には、次の2つのカテゴリのレルムが存在します。

- 非ルートレルム
- ルートレルム: メッシュには、メッシュの「中心」にレルムが1つ以上存在します。これらのレルムはルートレルムです。接続のルーティングがゲートウェイで求められると、その接続にターゲットレルムが指定されていない場合は、「直近」のルートレルムに接続がルーティングされます。ここで言う「直近」とは、最低のネットワークコストで到達できるルートレルムを指します。SAでは、ルートレルムとファシリティとの間に特別な関係も持たせています。SAコアがインストールされるたびに、ファシリティが1つ作成されますが、同時に、新たに作成されたファシリティと1対1の関係を持つルートレルムも1つ作成されます。

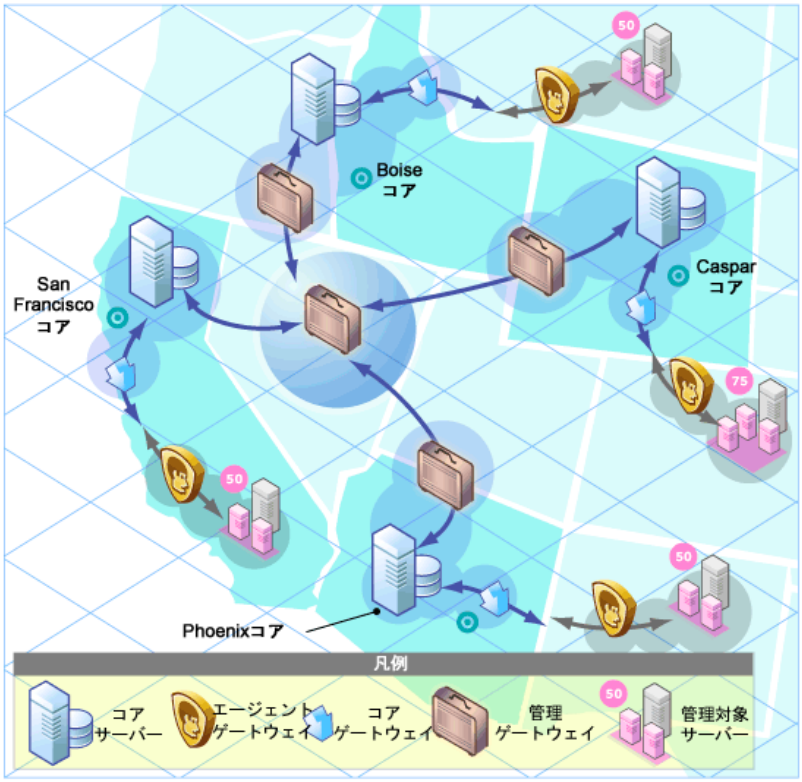
マルチマスターメッシュのトポロジの例

次の図は、San FranciscoとLos Angelesという2つのファシリティにインストールされているコアを含むマルチマスターメッシュを示しています。各ファシリティのコアにはモデルリポジトリが存在し、両方のファシリティ内にある管理対象サーバーに関するデータが保存されています。このデータは、両方のファシリティのモデルリポジトリ間で、定期的に同期 (レプリケーション) されます。コアはそれぞれの管理ゲートウェイ経由で通信します。



Los Angelesのファシリティ内にある管理対象サーバーから、San Franciscoのコアへの通信は、Los Angelesのエージェントゲートウェイを経由してコアゲートウェイ、さらにLos Angelesの管理ゲートウェイへと転送されます。さらにこの管理ゲートウェイは、San Franciscoの管理ゲートウェイとコアゲートウェイを経由して、San Franciscoのコアと通信します。

次の図は、4つのコアを含むマルチマターメッシュを示します。このメッシュトポロジはスター型と呼ばれるものであり、メッシュの中心にSan Franciscoのコアがあります。SAのインストーラーは、スタートポロジでマルチマスターメッシュを構成するときに、デフォルトでバックアップゲートウェイルートを設定します。



SAサテライト

サテライトは、管理対象サーバーの数が少なく完全なSAコアインストールを必要としないリモートサイト向けのソリューションです。サテライトでは、ホストに最小限必要なコアコンポーネントのみをインストールでき、ホストからプライマリコアのデータベースとその他サービスにSAゲートウェイ接続経由でアクセスします。

また、限られたネットワーク接続を使ってプライマリファシリティと接続する場合には、帯域幅の問題を軽減することもできます。サテライトで使用するネットワーク帯域幅の上限となるビットレートを指定することができます。これにより、サテライトのネットワークトラフィックによって、同じパイプ上にある他の重要なシステムのネットワーク帯域幅要件が影響を受けることがなくなります。

一般的に、サテライトの最低構成にはサテライトゲートウェイとソフトウェアリポジトリキャッシュが含まれますが、リモートファシリティでサーバー管理機能をフル装備することも可能です。ソフトウェアリポジトリキャッシュには、サテライト内の管理対象サーバーにインストールされているソフトウェアパッケージのローカルコピーが格納され、サテライトゲートウェイは、プライマリコアとの通信を処理します。

オプションで、SAプロビジョニングブートサーバーとメディアサーバーをサテライトホストにインストールし、リモートSAプロビジョニングをサポートすることが可能です。ただし、サテライトホストには、これ以外のコンポーネントはインストールできません。

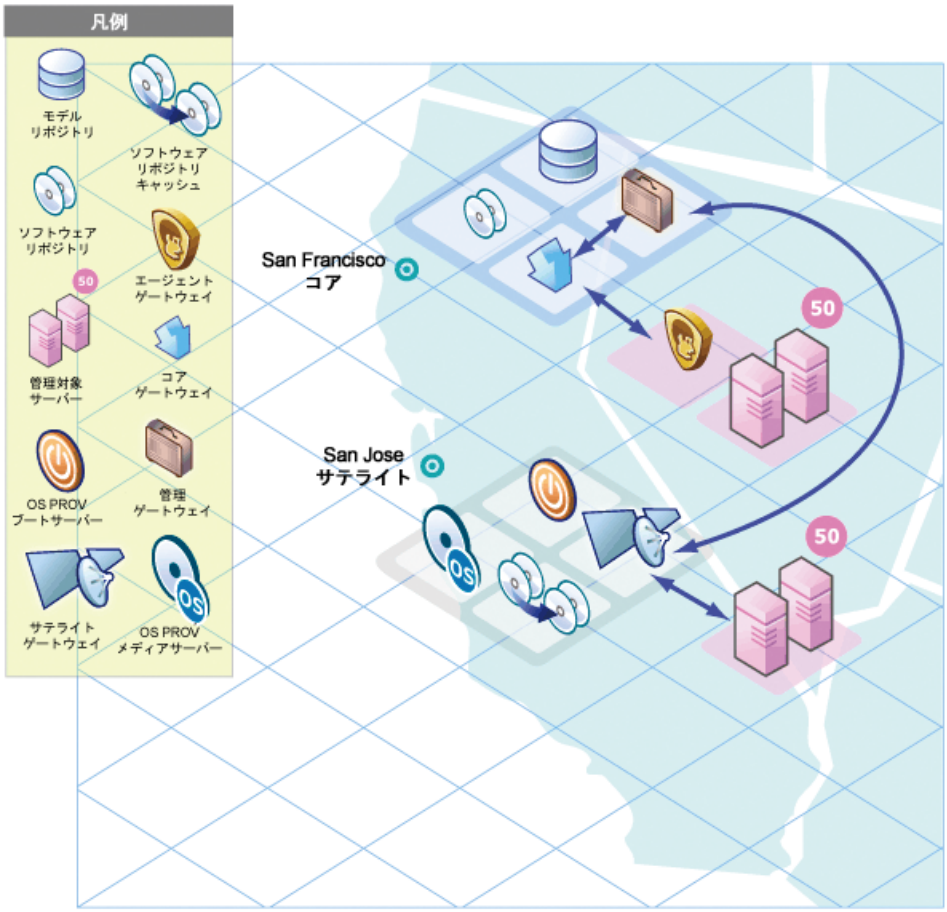
サテライトトポロジの例

シンプルな単一コアとサテライトのリンク

次の図では、単一のサテライトが単一のコアにリンクされています。この例では、San Franciscoにメインファシリティがあり、これよりも小規模なリモートファシリティがSan Joseにあります。

San Franciscoの単一コアにはいくつかのコンポーネント (ソフトウェアリポジトリ、モデルリポジトリ、エージェントゲートウェイ、管理ゲートウェイ) が含まれます。この図では、簡略化のために、コマンドエンジンなど一部コアコンポーネントが省略されています。

San Joseのサテライトには、ソフトウェアリポジトリキャッシュ、サテライトゲートウェイ、オプションのSAプロビジョニングブートサーバーとメディアサーバーが含まれます。



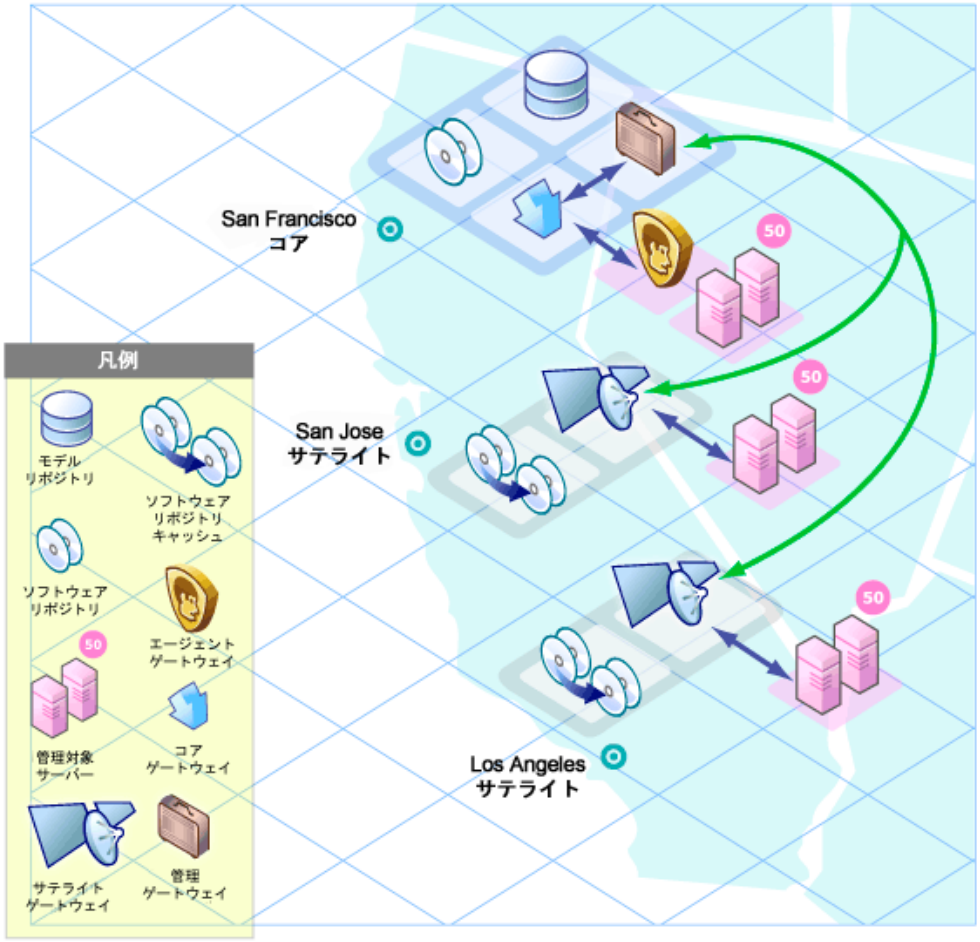
San Joseのサテライトのソフトウェアリポジトリキャッシュには、ファシリティ内の管理対象サーバーにインストールするソフトウェアパッケージのローカルコピーが格納されています。

San Joseのファシリティにある管理対象サーバーにインストールされているサーバーエージェントは、San FranciscoのコアにSan Joseのサテライトゲートウェイ経由で接続します。このゲートウェイは、San Franciscoの管理ゲートウェイと通信し、そこからSan Franciscoのコアゲートウェイを経由して、目的のコアコンポーネントへとつながります。

応答の通信では、このパスが逆方向になります。San Franciscoのファシリティ内にある管理対象サーバーにインストールされているサーバーエージェントは、San Franciscoのファシリティのエージェントゲートウェイとコアゲートウェイを経由してコアコンポーネントと通信します。

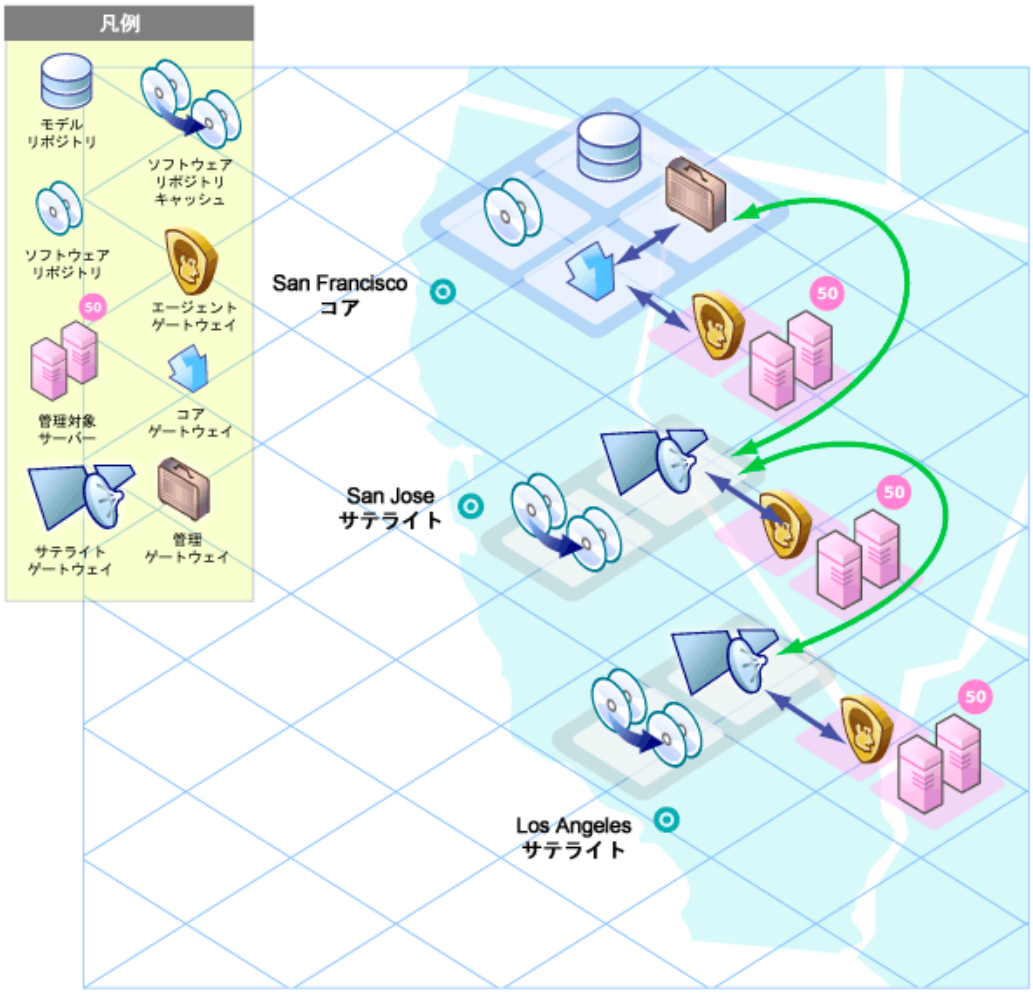
単一コアにリンクする2つのサテライト

次の図では、2つのサテライトが単一のコアにリンクしています。この例では、San Franciscoがメインファシリティであり、SunnyvaleとSan Joseはサテライトファシリティです。



サテライトリンクのカスケーディング

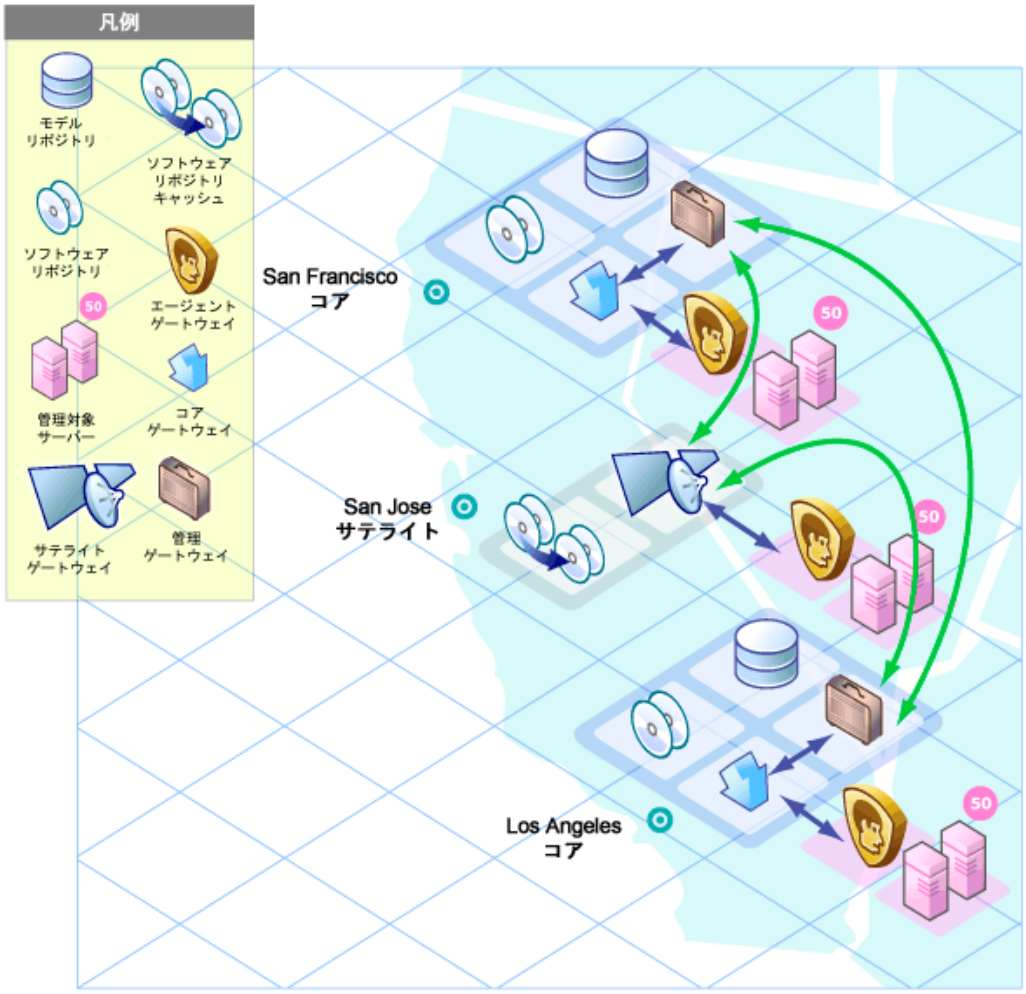
次の図は、サテライトのカスケーディングを示しています。このトポロジでは、サテライトゲートウェイがチェーン状に接続されています。このトポロジでは、ソフトウェアリポジトリキャッシュの階層化が可能です。また、このトポロジではサテライトゲートウェイを異なるSAレルムに割り当てる必要がある点に注意してください。



Los Angelesのファシリティにある管理対象サーバーにパッケージをインストールする場合、SAはまず、Los Angelesのソフトウェアリポジトリキャッシュ内にパッケージが格納されていないかチェックします。パッケージがLos Angelesにない場合、SAはSan Joseのソフトウェアリポジトリキャッシュをチェックします。San Joseにもパッケージがない場合、SAはSan Franciscoのコアのソフトウェアリポジトリを確認します。詳細については、『SA 10.50管理ガイド』を参照してください。

マルチマスターメッシュ内のサテライト

次の図では、マルチマスターメッシュ内にある2つのSAコアにSan Joseのサテライトが接続しています。



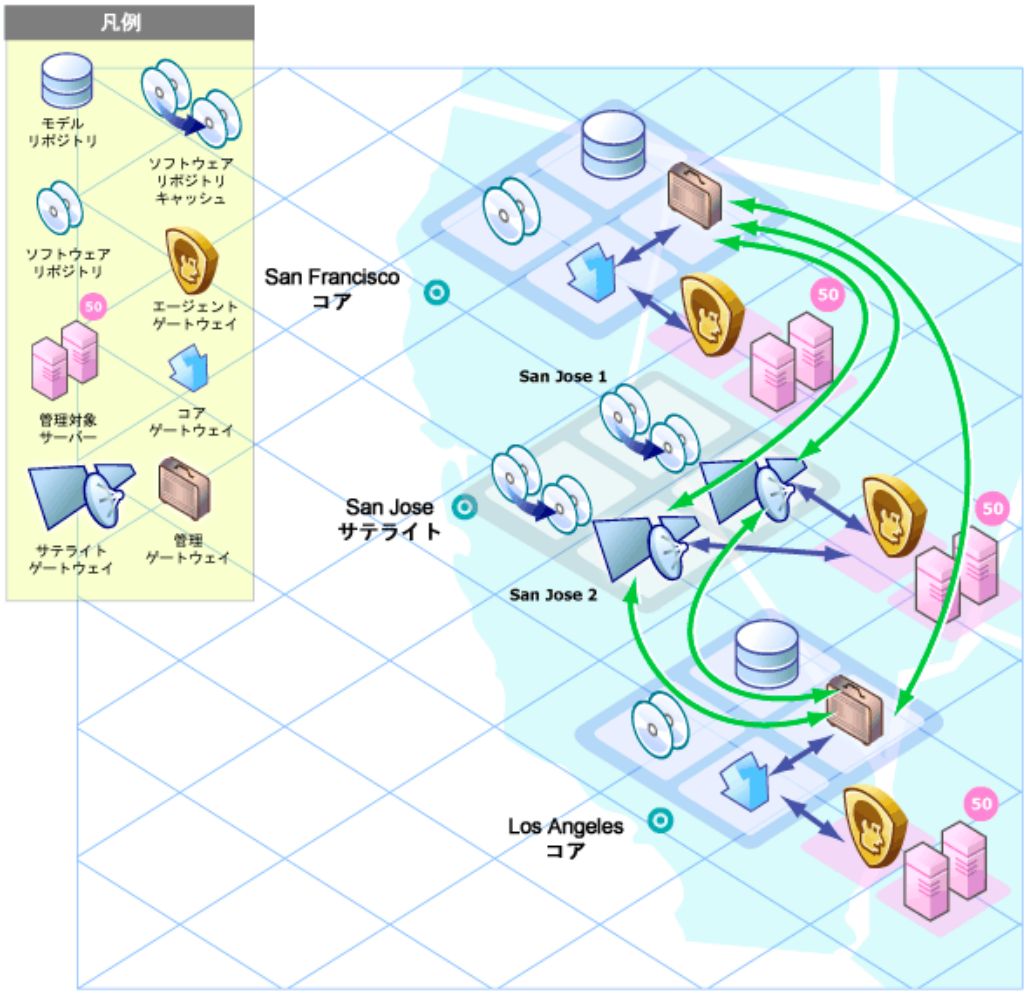
Los AngelesとSan Franciscoの両方に通信可能な場合、管理ゲートウェイはコストが低い方 (この図ではSan Franciscoのルート) を選択します。コスト評価は、ゲートウェイのインストールで指定したパラメーターで制御します。システム設計者は、ネットワーク接続コストが最小になるのはどのSAゲートウェイルートなのかを指定するルールを定義できます。

フェイルオーバーシナリオで使用したサンプル環境を使用する場合、正常稼働時には、San Joseのサテライトにあるサーバーはは San Franciscoのコアによって管理されます。ただし、San FranciscoとLos Angelesのコアは、それぞれの管理ゲートウェイを経由して直接接続しています。

San Joseのサテライトと San Franciscoのコア間の接続が切断されると、San Joseのサテライトゲートウェイはすぐに、San FranciscoからLos Angelesのコアへと通信を移動します。これにより、コアはSan Joseのサーバー管理を継続できます。San Franciscoのコアのモデルリポジトリのデータは、通常のSAオペレーションでLos Angelesのモデルリポジトリに複製されるので、Los AngelesのコアにはSan Joseサイトの最新情報が格納されます。

マルチマスターメッシュで複数のゲートウェイを持つサテライト

次の図は、2方向でフェイルオーバーを実行できるトポロジの例です。まず、San Joseのサテライト1と2には、San FranciscoとLos Angeles両方の管理ゲートウェイにつながるゲートウェイ接続があります。Los Angelesのコアが利用不可になった場合、San FranciscoのコアがSan Joseのサテライトにあるサーバーを管理します。



次に、San Joseのファシリティ内にある管理対象サーバーにインストールされているエージェントは、サテライトの両方のエージェントゲートウェイをポイントしています。SAエージェントは、利用不可なエージェントゲートウェイに対して自動的に負荷分散するので、San FranciscoまたはLos Angelesのコアと直接通信できます。

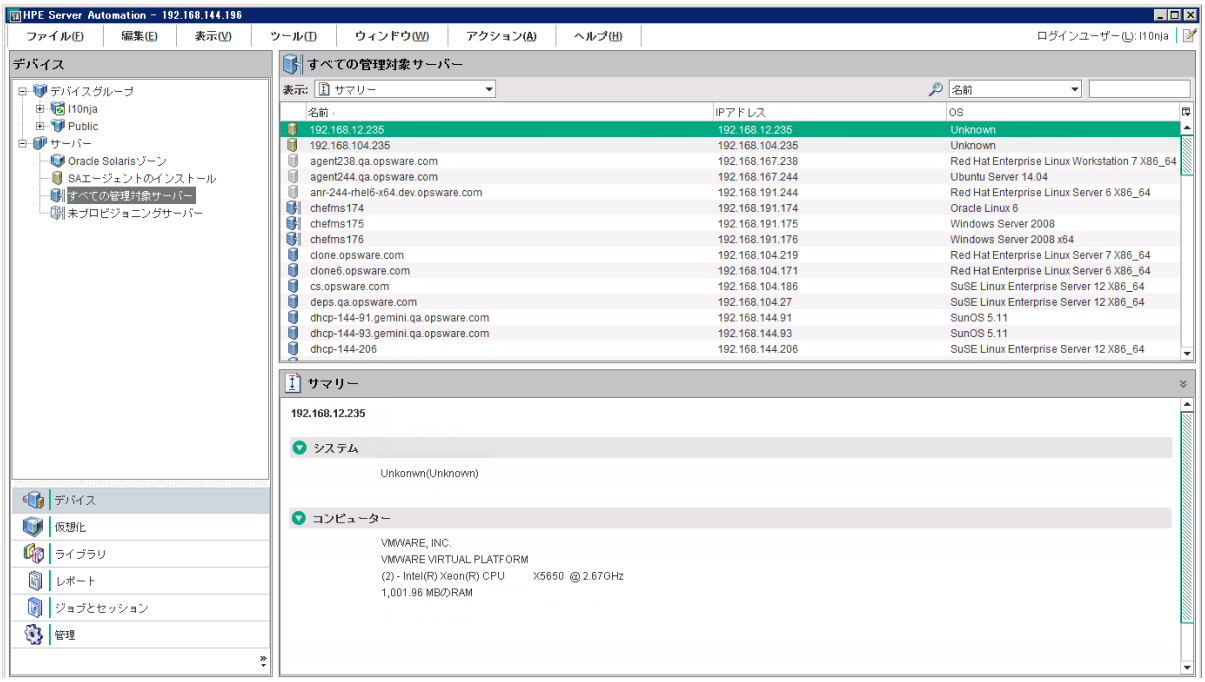
いずれかのゲートウェイが利用不可になった場合、利用不可なゲートウェイをプライマリゲートウェイとするエージェントは、セカンダリゲートウェイへと自動的にフェイルオーバーします。通常のエージェントとコア間の通信では、SAエージェントはサテライトに新しく追加 (または削除) されたゲートウェイを検出します。

SAクライアント

SAクライアントはWindowsアプリケーションであり、SAの後にインストールします。SAへのインタフェースとして機能します。

SAクライアントをインストールするには、コアのホームページを開き、[**Download Server Automation Client**] をクリックして、SAクライアントをダウンロードしてインストールする必要があります。

次の図は、SAクライアントのメイン画面です。SAクライアントの詳細については、『ユーザーガイド』を参照してください。

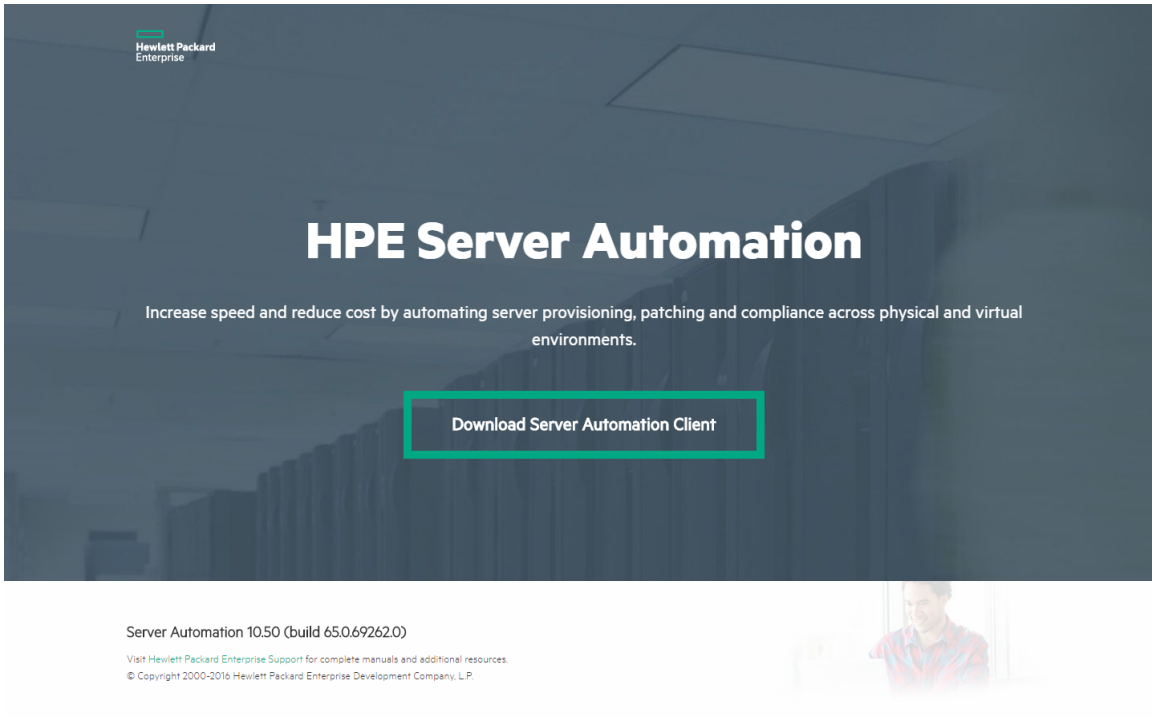


SA Webクライアント

SA Webクライアントは非推奨になりました。いくつかのSA機能は今でもSA Webクライアントによって提供されていますが、可能であればSAクライアントを使用してください。

SA Webクライアントは、WebベースのSAとのユーザーインタフェースであり、SAクライアントランチャーをダウンロードするのに使用します。ブラウザーでSA Webクライアントを起動する手順は、『SA 10.50ユーザーガイド』を参照してください。SA Webクライアントを起動した後に、SAクライアントランチャーの実行可能ファイルをダウンロードしてインポートできます。

次の図は、SA Webクライアントのホームページを示しています。



機能

SAは、ミスが発生しやすい手動のアドホックなデータセンタープロセスを自動化します。たとえばSAプロビジョニングでは、サーバーのタイプごとに標準を設定し、サーバープロビジョニングを自動化します。これによって、時間が節約できるだけでなく、一貫性のあるオペレーティングシステム環境を構築できます。

パッチポリシーでは、IT環境内の管理対象サーバーで稼働するサポート対象オペレーティングシステムにおいて、パッチのインストールと管理を実行できます。

コンプライアンスでは、管理対象サーバーにコンプライアンス違反がないかを把握できます。非コンプライアンスサーバーが検出されたら、作成したポリシーに基づいて修復し、準拠した状態に戻します。

SAでは、次の機能を提供しています。

SAは、クロスプラットフォームをサポートし、新規と既存のデータセンター環境の自動化に対応します。

デバイスエクスプローラー

デバイスエクスプローラーでは、管理対象環境で稼働するサーバーの情報が表示されます。

サーバーエクスプローラーでは、次のタスクを実行できます。

- サーバースナップショットの作成、サーバー監査の実行、アプリケーション構成の監査、パッケージの作成、リモートターミナルセッションをリモートサーバーで開く、などを実行します。
- サーバーのファイルシステム、レジストリ、ハードウェアインベントリ、ソフトウェアとパッチのリスト、サービスを参照します。
- SAの情報 (プロパティ、構成可能なアプリケーション、サーバー履歴など) を参照します。
- グループブラウザーでは、次のタスクを実行できます。
- システム情報の監査、サーバースナップショットの取得、アプリケーションの構成を行います。
- グループメンバー (サーバーグループとその他グループ) の表示とアクセスを行います。
- グループのサマリー情報と履歴情報を表示します。

仮想化管理

HPEでサポートする仮想化ベンダーとクラウドコンピューティングの統合ソリューションは、仮想化サービスと呼ばれます。

仮想化ベンダーは、仮想化環境内の複数のハイパーバイザーとVMを管理します。HPEでは、VMware vCenterサーバーやMicrosoft System Center Virtual Machine Manager (SCVMM) との統合をサポートしています。

OpenStackなどのクラウドコンピューティングソリューションは、サービスとしてのインフラストラクチャー (IaaS) を提供します。HPEでは、OpenStackとの統合を制限付きでサポートしています。

HPE Server Automationで仮想化を管理すると、次のような利点があります。

- データセンター、すべての物理マシンと仮想マシン (VM) を**可視化**します。
- 法規制や社内ポリシーすべてにおいて**コンプライアンス**を確保します。
- 仮想環境全体を**管理**することにより、VMの無秩序な増加を防止し、短時間で問題を解決します。

アプリケーション構成管理

アプリケーション構成管理 (ACM) は、テンプレートを作成することによって、サーバーアプリケーションに関連付けられているアプリケーション構成の変更と管理を行います。このような構成の管理、更新、変更を一元的に行うことで、ファシリティ内のアプリケーションの構成を正確で一貫性のある状態に維持できます。

ACMでは、次のタスクを実行できます。

- ファイルとオブジェクト (Windowsレジストリ、IISメタベース、WebSphere、COM+など) に基づいて構成を管理します。
- 構成変更を適用する前に、変更前にプレビューします。
- 構成変更を編集し、個々のサーバーまたはサーバーグループにプッシュします。
- SAデータモデルの情報を使用して、構成値を設定します。
- 構成テンプレートを作成することにより、任意のアプリケーションの構成を管理します。
- サーバーのアプリケーション構成を監査し、サーバー上にテンプレートで指定した値と一致しない構成ファイルがないか確認します。

詳細については、『SA 10.50開発者ガイド』を参照してください。

監査と修復

監査と修復では、IT環境内でチェック対象のオブジェクト、チェックを行う場所とタイミングを指定できます。

- 監査ポリシーでは、チェック対象 (ファイル、ディレクトリ、構成値など) を定義します。
- 監査では、チェック対象となる場所 (サーバー、サーバーグループなど) を定義します。
- 監査スケジュールでは、チェックを行うタイミング (特定の日時、反復実行など) を定義します。

これらの機能を使用することによって、管理対象サーバー環境でコンプライアンスを確保し、サーバーのコンプライアンス状態を維持する方法を理解できます。SAでは、ファシリティ内のサーバーを標準ポリシーに準拠した状態にする方法として、サーバー構成ポリシーを使用します。非コンプライアンス状態 (想定通りに構成されていない状態) として検出されたサーバーは、修復によって、組織で規定されている標準に準拠した状態にすることが可能です。

SAクライアントでは、動作中のサーバーまたはサーバースナップショット、ユーザー指定の値、事前定義された監査ポリシーをベースに、サーバー構成値の監査を実行します。また、サーバー構成スナップショットを使ってシステムの現在の状態を取得し、他のサーバーと比較することも可能です。

監査ポリシーでは、会社全体または業界共通のコンプライアンスおよびセキュリティ標準を定義し、監査やスナップショット仕様などの監査ポリシーで使用できます。監査やスナップショット仕様で定義した監査ポリシーを参照すれば、組織内の最新のコンプライアンス定義に適応できているかどうかを確認できます。

監査と修復では、次のタスクを実行できます。

- サーバーまたはスナップショットと評価基準となるサーバーまたはスナップショットとの比較
- 反復使用できる監査の作成
- 組織に適用するコンプライアンス標準およびセキュリティ標準を定義する監査ポリシーを作成
- 個々のサーバーまたは動的なサーバーグループに監査を関連付ける
- 複数のレベル(ファイル、ディレクトリ、パッチ、レジストリキー、パッケージなど)で問題を修復

Windowsパッチ管理

Windowsパッチ管理では、Microsoft® Windowsパッチの確認、インストール、削除によって、組織内にある管理対象サーバーのセキュリティを確保します。Windowsオペレーティングシステムについて、セキュリティの脆弱性に対するパッチを確認し、インストールできます。

詳細については、『SA 10.5 Support and Compatibility Matrix』を参照してください。

Windowsではセキュリティの脅威に対するパッチが頻繁にリリースされるので、システムをリスクにさらさないようにするには迅速にパッチを適用する必要があります。ただし、パッチを誤って適用すると、パフォーマンスの低下や重大なエラーなど深刻な問題が発生する原因になるので注意が必要です。

パッチ管理では、新しく検出された脅威に迅速対応できるだけでなく、パッチインストールの厳格なテストと標準化をサポートします。さらに、パッチが原因で問題が発生した場合には、テストと承認の後であっても、Windowsパッチ管理で安全かつ標準化された方法でパッチをアンインストールできます。

詳細については、『SA 10.50ユーザーガイド』を参照してください。

HP-UXパッチ管理

SAでは、次の機能によってHP-UXパッチ管理を自動化します。

- モデルベースのアプローチでHP-UXサーバーを管理するHP-UXソフトウェアポリシーを定義します。Server Automationでは、HP-UXソフトウェアポリシーを使用して、IT環境のモデルを作成できます。これらのソフトウェアポリシーでは、管理対象サーバーにインストール可能なパッチとスクリプトを指定します。
- HP-UXパッチとパッチバンドルを管理対象サーバーにインストールします。
- パッチインストールプロセスを確立します。
- パッチ管理の各ステージ (分析、ダウンロード、インストール) のスケジュールを設定します。また、各ステージごとに電子メール通知を設定し、ジョブにチケットIDを関連付けることもできます。
- ソフトウェアポリシーに基づいて、サーバーのコンプライアンスステータスを検証します。
- コンプライアンスビューで、サーバーがソフトウェアポリシーに基づいて構成されているかを確認し、非コンプライアンスサーバーがあれば修復します。
- ソフトウェアリソースとサーバーを検索します。
- SAライブラリでは、強力かつ柔軟な検索条件 (可用性、アーキテクチャー、オペレーティングシステム、再起動オプション、バージョンなど) を使用して、HP-UXパッケージ、パッチ、ソフトウェアポリシーを検索します。HP-UXソフトウェアポリシーは、名前、フォルダー名、可用性、オペレーティングシステムで検索することも可能です。
- パッチインストールのプレビューでは、パッチの依存関係とパッチの適用性分析が表示されます。

詳細については、『SA 10.50ユーザーガイド』を参照してください。

SolarisおよびSolaris 11のパッチ管理

Server Automationパッチ管理では、Solarisパッチの確認、インストール、削除によって、組織内にある管理対象サーバーのセキュリティを確保します。

Server AutomationのSolarisパッチ管理は、パッチポリシーを適用することによって、Solarisパッチとパッチクラスターのインストールとアンインストールを自動化します。さらに、ポリシー内のパッチ間の依存関係、優先度、適用可能性を分析し、サーバーへのインストール順序を示す最新のパッチリストを表示します。この機能により、サーバーのコンプライアンスステータスが検証できます。非コンプライアンスサーバーが検出された場合は修復し、SolarisパッチをSAに自動ダウンロードしてパッチポリシーに組み込むことができます。

SAでは、次の機能によってSolarisでのパッチ適用を自動化します。

- 管理対象サーバーで必要なパッチを特定します。
- Solarisパッチポリシーを作成します。

- Solarisパッチ、パッチクラスター、パッチバンドルをダウンロードし、さらにそれらと関連ベンダー情報をSAライブラリに保存します。
- Solarisパッチが依存するパッチをすべて解決します。
- Solarisパッチとパッチクラスターを管理対象サーバーにインストールします。
- Solarisパッチをシングルユーザーモードでインストールします。
- Oracle Solarisゾーンごとにパッチをインストールします。
- パッチインストールプロセスを確立します。
- パッチポリシーに基づいて、サーバーのコンプライアンスステータスを検証します。
- ソフトウェアリソースとサーバーを検索します。

詳細については、『SA 10.50ユーザーガイド』を参照してください。

Ubuntuパッチ管理

HPE Server AutomationのUbuntuパッチ管理では、Ubuntu Debianパッケージ更新の確認、インストール、削除によって、組織内にある管理対象サーバーのセキュリティを確保します。SAでサポートされる管理対象サーバープラットフォームのセキュリティ脆弱性に対して、対応するUbuntuパッケージを確認してインストールすることができます。

SAではパッチ管理の主要な機能が自動化されていますが、Ubuntuパッケージのインストール方法やインストール条件は、細かく制御することができます。パッチ適用プロセスを自動化することで、パッチ適用に伴うダウンタイムを短縮できます。また、SAでは、パッチアクティビティのスケジュールを設定することで、ピーク以外の時間帯にパッチを適用することができます。

SAのUbuntuパッチ適用を使用して、バイナリパッケージをインポートする前にメタデータをインポートできます。ダウンロードしたメタデータのみを使用して、Ubuntuスキャナーを実行し、サーバーの脆弱性を特定できます。その後、Ubuntuパッケージインポーターを実行して、管理対象サーバーに必要なパッケージのみをインポートできます。この方法により、ストレージ容量と、スキャンおよび修復プロセス時間を節約できます。

Ubuntuパッチ管理のドキュメントには、Ubuntuメタデータおよびパッケージのインポート方法、脆弱性のスキャン、パッチポリシーを使用したUbuntuパッケージ更新のインストールに関する情報が記載されています。

SAでは、次のような機能や特徴を利用して、Ubuntuパッチ適用を自動化しています。

- **セントラルリポジトリ**: パッケージがそれぞれの標準形式で保存され、整理されます
- **データベース**: これまでに適用したすべてのパッケージの情報を保存します
- **動的パッチポリシー**: ベンダーの最新メタデータに基づいて、プラットフォーム脆弱性を分析します。
- **高度な検索機能**: パッケージ更新が必要なサーバーを識別できます
- **監査機能**: 重要なパッケージ更新のデプロイメントをトラッキングします

詳細については、『SA 10.50ユーザーガイド』を参照してください。

UNIXパッチ管理

UNIXパッチ管理では、パッチの確認、インストール、削除によって、組織内にある管理対象サーバーのセキュリティを確保します。SAクライアントは、AIXオペレーティングシステム環境に存在するセキュリティの脆弱性に対するパッチを特定して、インストールできます。

SAは、新しく検出された脅威に迅速対応できるだけでなく、パッチインストールの厳格なテストと標準化をサポートします。さらに、パッチが原因で問題が発生した場合には、テストと承認の後であっても、安全かつ標準化された方法でパッチをアンインストールできます。

SAは、SAライブラリにパッチ情報を保存します。これには、管理対象サーバー、サーバー上にインストールされているパッチとソフトウェア、インストール可能なソフトウェアとパッチに関する詳細な情報が含まれます。このデータを元に新しく検出された脅威の重大度を判定し、パッチをインストールした場合のメリットとダウンタイムコストを比較して、テスト要件を特定します。

パッチの適用手順を自動化することで、パッチ適用に伴うダウンタイムを短縮できます。また、パッチアクティビティのスケジュールを設定することで、ピーク以外の時間帯にパッチを適用することができます。

UNIXパッチ管理では、次の機能が提供されているため、オペレーティングシステムごとのパッチの参照、パッチのダウンロードとインストールのスケジュール設定、電子メール通知の設定、パッチインストールのプレビュー、ソフトウェアポリシーと修復によるパッチのインストールとアンインストール、再利用可能なファイル形式へのパッチ情報のエクスポートなどを実行できます。

- パッチの保存先であり、各形式で編成されているSAライブラリ
- これまでに適用したパッチの情報が格納されたデータベース
- パッチインストールの前後に実行できるカスタマイズスクリプト
- パッチの適用が必要なサーバーを特定できる高度な検索機能
- セキュリティ担当者が重要なパッチのデプロイメントを追跡できる監査機能

詳細については、『SA 10.50ユーザーガイド』を参照してください。

レポート

SAのレポートでは、環境内の管理対象サーバー、ネットワークデバイス、ソフトウェア、パッチ、カスタマー、ファシリティ、オペレーティングシステム、コンプライアンスポリシー、ユーザーとセキュリティに関する包括的な情報がリアルタイムで提供されます。レポートはグラフと表の形式で提示され、ポリシーや監査などのレポート内のオブジェクトに対して適切なアクションを実行できるようになっています。また、組織で使用しやすいファイル形式 (`.html`と`.xls`) でローカルファイルシステムにエクスポートもできます。

SAプロビジョニング

SAプロビジョニングでは、ベアメタルサーバーと仮想サーバーに対して事前構成済みのオペレーティングシステムベースラインのインストール (またはプロビジョニング) を一貫した方法で迅速に実行し、手動による操作を最小限に抑えることができます。ベアメタルサーバーと仮想サーバーでのSAプロビジョニングは、サーバーを運用環境に移行するプロセス全体の中で重要な部分です。

SAプロビジョニングにより、ファシリティ内の各サーバーに、標準として定められたデフォルトのオペレーティングシステム構成を適用することができます。SAプロビジョニングの詳細については、『SA 10.50管理ガイド』を参照してください。

SAプロビジョニングには、次のような利点があります。

- **他のSA機能との統合**
  SAプロビジョニングは、パッチ管理、ソフトウェア管理、分散スクリプト実行といったSAの自動化機能と統合されているので、ITグループ間でシームレスな受け渡しが可能になります。SAを使用することによって、すべてのITグループが環境の現状を理解した上で連携できるので、高品質のオペレーションと信頼性の高い変更管理を実現する上で大きな効果を発揮します。
- **イメージの再適用なしでサーバーベースラインを更新** 他のプロビジョニングソリューションとは異なり、でプロビジョニングを行ったシステムは、プロビジョニング後も簡単に変更でき、新しい要件に適応させることができます。これを可能にしているのが、テンプレートの使用と、インストールベースでプロビジョニングを行うアプローチです。
- **さまざまな環境に対応できる柔軟なアーキテクチャー** プロビジョニングでは、さまざまなタイプのサーバー、ネットワーク、セキュリティアーキテクチャー、オペレーションプロセスがサポートされます。SAは、CD (Linuxプロビジョニング) やネットワークブート環境 (DHCPと非DHCPの両方の環境) で問題なく動作し、スケジュールされたワークフローまたはオンデマンドのワークフロー、幅広いハードウェアモデルをサポートします。このように優れた柔軟性を備えているので、ユーザー組織のニーズに合わせたオペレーティングシステムのプロビジョニングが可能になります。

SAプロビジョニングは、SAクライアントから実行できます。SAは、包括的なサーバーベースラインのプロビジョニングプロセス全体を自動化します。これは次のタスクで構成されます。

- OSインストールプロファイルを使用して、OSをインストールできるようにハードウェアを準備します(OSシーケンスのみで必要)。
- オペレーティングシステムインストールの前後に実行するタスクを定義するOSビルド計画を作成します。OSビルド計画は、OSシーケンスの代わりに使用すると便利です。
- インストール中にサーバーで実行するタスクを定義するOSシーケンスを作成します。OSシーケンスでは、アプリケーション、パッチ、修復の各ポリシーを定義できます。ただし、OSシーケンスよりも柔軟性の高いOSビルド計画の使用をお勧めします。
- OSビルド計画またはOSシーケンスを使用して、基本オペレーティングシステムとデフォルトのOS構成をインストールします。
- 最新のOSパッチセットを適用します。実際の内容は、サーバーで実行するアプリケーションによって異なります。
- rootパスワードなどのシステム構成を行うインストール前スクリプトまたはインストール後スクリプトを実行します。
- システムエージェントとユーティリティ(SSH、PC Anywhere、バックアップエージェント、監視エージェント、ウイルス対策ソフトウェアなど)をインストールします。
- Java仮想マシンなど、広い範囲で共有するシステムソフトウェアをインストールします。

SAプロビジョニングのサポート対象:

- Windows、Solaris、Linux
- ネットワークベースまたはCD/DVDベースのインストール
- データセンターのスタッフとシステム管理者の作業分担
- モデルベースのアプローチにより、多数のシステムにインストール可能な標準ビルドをSAで作成

SAプロビジョニングは、オペレーティングシステムベンダー独自のインストールテクノロジーを統合します。

- Windowsセットアップ作業の応答ファイル: `unattend.xml`、`sysprep.inf`
- Red Hat Kickstart
- SuSE YaST (Yet another Setup Tool)
- Solaris Jumpstart
- WINPE/WIN-BCOM/UNDI

オペレーティングシステムのプロビジョニングは、次のサーバーで実行できます。

- SAのエージェントレスサーバープールに含まれ、オペレーティングシステムがインストールされていないサーバー (ベアメタルサーバー)
- 仮想サーバー
- SAの非管理対象サーバープールに含まれ、オペレーティングシステムがインストール済みのサーバー
- SAの管理対象サーバープールに含まれ、オペレーティングシステムがインストール済みのサーバー (再プロビジョニング)

アプリケーションデプロイメント

アプリケーションデプロイメントでは、データセンター内のターゲットサーバーで、カスタムソフトウェアアプリケーションの作成、テスト、デプロイメントを行います。たとえば、テストを行う目的で、アプリケーションを開発チームから品質保証チームに移動します。テストが完了したアプリケーションは、運用準備、ステージング、さらに運用開始など各フェーズへと移動します。アプリケーションデプロイメントツールでは、単一のアクセスポイントから、すべてのユーザーがそれぞれの役割に応じて関連データを表示または入力できるので、アプリケーションのデプロイメントに伴う複雑なコミュニケーションを簡略化できます。

アプリケーションデプロイメントには次のような機能があります。

- コード、スクリプト、構成ファイル、層 (アプリケーションサーバー、Webサーバー、データベースなど) といったアプリケーションコンポーネントをモデル化します。
- アプリケーションの複数のリリースとバージョンを管理します。
- ターゲットサーバーでアプリケーションのデプロイ、ロールバック、アンデプロイを行います。
- アプリケーションで必要になる層を実行するターゲットサーバーをモデル化します。このターゲットサーバーは、Server Automationで管理します。
- ソフトウェアアプリケーション開発者、品質保証およびテスト担当者、システム管理者、その他オペレーションの担当者が、相互に明確で簡潔な方法でコミュニケーションを図る手段を提供します。
- アプリケーション開発、QA、運用準備、ステージング、運用開始というライフサイクルをモデル化し、実装します。ユーザーのエンタープライズライフサイクルに合わせたカスタマイズが可能です。

詳細については、『SA 10.50開発者ガイド』を参照してください。

スクリプト実行

SAのスクリプト実行では、SAで管理するサーバーファーム全体で、アドホックスクリプトや保存済みスクリプトを共有および実行できます。

手動ではなくSAを使用してスクリプトを実行する方法には、次のような利点があります。

- 多数のUNIXまたはWindowsサーバーでスクリプトを並列実行することにより、時間を節約できるだけでなく、一貫性を維持できます。
- 役割ベースのアクセス制御を適用することで、承認した管理者のみに、アクセス権を持つホストでのスクリプト実行を許可します。
- プライベートライブラリとパブリックライブラリにスクリプトを保存し、スクリプトのアクセス制御を行います。
- スクリプト出力の表示とダウンロードを行います。サーバーごとのレポート、またはすべてのサーバーの出力を1箇所に集約したレポートを作成します。
- スクリプトの一括カスタマイズが可能です。管理者は、SAに保存されているサーバーの環境と状態に関する情報にアクセスできます。これは、適切なサーバーで適切なスクリプトを実行する上で非常に重要な機能です。
- 包括的な監査証跡を作成し、各スクリプトの実行者と実行日時を記録します。

スクリプト実行はSAの統合された機能であり、スタンドアロンのスクリプト実行ツールに比べて、次のような利点があります。

- システムの状態や構成情報をベースに、スクリプト実行をカスタマイズできます。SAには、サーバーを所有するカスタマーやビジネス、サーバーがステージングサーバーか運用サーバーかの区別、サーバーが設置されているファシリティ、カスタマイズした名前と値のペアなどさまざまな情報が保存されているので、これを参照またはアクセスすることでスクリプトをカスタマイズします。
- セキュリティを損なわない方法でスクリプトを共有できます。SAでは、スクリプト実行を許可するユーザーと対象サーバーのアクセス制御が厳格に実施され、スクリプト実行の監査証跡が作成されるので、セキュリティを損なわずにスクリプトを共有できます。

エージェントレスサーバーの検出とSAエージェントのインストール

エージェントレスサーバーの検出とSAエージェントのインストールでは、ファシリティ内にある多数のサーバーにサーバーエージェントをデプロイし、SAで管理することができます。

次のタスクを実行できます。

- ネットワークでエージェントレスサーバーをスキャンします。
- SAエージェントをインストールするサーバーを選択します。
- 通信ツールを選択し、ユーザーとパスワードを提供します。
- エージェントのインストールオプションを選択し、エージェントをデプロイします。

Service Automation Visualizer (SAV)

Service Automation Visualizer (SAV) は、IT環境内に分散したビジネスアプリケーションについて、オペレーションアーキテクチャーと動作に関する情報提供と管理を効率的に行います。分散ビジネスアプリケーションは、多くのサーバーで実行されるサービス、およびネットワークを含む複雑な構成なので、相互の関連性、パフォーマンス低下の原因、トラブルシューティングと問題解決の方法、環境の変更がもたらす結果を把握することはますます難しくなっています。

SAVでは、このような情報を物理的な側面と論理的な側面からグラフィック表示します。

SAクライアントでのコンプライアンス

SAクライアントのコンプライアンスビューでは、ファシリティ内にあるすべてのサーバーとサーバーグループの全体的なコンプライアンスレベルを確認できます。一般的にはコンプライアンスダッシュボードと呼ばれるこのビューから、非コンプライアンス状態のサーバーを修復することができます。コンプライアンスを表示する対象として、個々のサーバー、複数のサーバー、サーバーグループ、すべてのSA管理対象サーバーを選択できます。

コンプライアンスダッシュボードには、サーバーまたはサーバーグループの監査、監査ポリシー、ソフトウェアポリシー、パッチポリシー、アプリケーション構成に対するすべてのコンプライアンスステータスの結果が表示されます。サーバーのコンプライアンスステータスは、コンプライアンスポリシーを基準に判定されます。コンプラ

イアンスポリシーではサーバー構成の設定や値が一意に定義されており、これに基づいてIT環境が想定通りに構成されているかどうかが確認されます。

コンプライアンスポリシーの作成と定義は、一般的にポリシー設定の担当者が行います。環境によっては、システム管理者がアドホックポリシーを作成する場合もあります。ポリシー設定担当者は、作成したコンプライアンスポリシーをサーバーにアタッチします。これによって、サーバーが組織の標準とポリシーに準拠しているかどうかを確認できます。

たとえば、ポリシー設定の担当者は、ソフトウェアポリシーを作成し、サーバー上にインストールしなくてはならないパッチとパッケージの標準セットを定義します。また、サーバーでの特定のアプリケーションファイルの構成方法も定義できます。サーバーまたはサーバーグループの構成が、ポリシー設定担当者がコンプライアンスポリシーで定義したルールと一致した場合、コンプライアンス状態であるとみなされます。

コンプライアンスダッシュボードでは、サーバーにインストールされているソフトウェア、パッケージ、パッチ、構成ファイルの実際の設定が、ソフトウェアポリシーで定義した構成と一致しているかどうかを確認できます。コンプライアンスビューでは、サーバーグループのコンプライアンスステータスを、グループのすべてのメンバーとサブグループのメンバーごとに表示できます。また、非コンプライアンス状態のサーバーとサーバーグループを検出し、問題を修復できます。

ソフトウェア管理

ソフトウェアポリシーを使用してソフトウェアをモデル化し、サーバーでのソフトウェアのデプロイとアプリケーションの構成を1つのステップで自動化する強力な機能があります。さらに、ソフトウェアリソースをフォルダー構造にまとめ、セキュリティに関するアクセス権を定義することもできます。また、サーバーのコンプライアンスステータスを検証し、非コンプライアンスサーバーを修復する機能もあります。

SA ソフトウェア管理では、次の機能を提供しています。

- ソフトウェアの組織構造を作成
- フォルダーでのセキュリティ境界を定義
- 組織内にあるIT環境の管理に適用するモデルベースのアプローチを定義
- ユーザーグループ間でソフトウェアリソースを共有
- アプリケーションのデプロイと構成を同時実行
- 1つのサーバーに複数のアプリケーションインスタンスをデプロイ
- ソフトウェアデプロイメントプロセスを確立
- ソフトウェアポリシーに基づいてサーバーのコンプライアンスステータスを検証

- レポートを作成
- ソフトウェアリソースとサーバーを包括的に検索

詳細については、『SA 10.50ユーザーガイド』を参照してください。

Global Shell

SA Global Shellでは、コマンドラインインタフェースからサーバーを管理します。次のタスクをリモート実行できます。

- 管理対象サーバーで繰り返し行う管理タスクを実行します。
- 管理対象サーバーで発生した問題のトラブルシューティング、特定、修復を行います。

Global Shellでは、SA内のサーバー管理にファイルシステムとコマンドラインインタフェースを使用します。このファイルシステムはSA Global File System (OGFS) と呼ばれます。OGFS内にあるすべてのタイプのオブジェクト (サーバー、カスタマー、ファシリティなど) はファイルシステムのディレクトリ構造で表示されます。

また、管理対象サーバー上にあるファイルシステム、Windowsレジストリ、Windowsサービスオブジェクトに対するユーザーアクセス権を管理することもできます。

FIPS 140-2準拠

HPE Server Automation (SA) は、Federal Information Processing Standards規格140-2に準拠します。これは、政府機関が検証済みの暗号モジュールを使用する機器を調達する際に適用するセキュリティ標準です。

ここでは、SAコア、サテライト、および管理対象サーバーのFIPS 140-2準拠状況とSAのFIPS 140-2準拠を実現するための方法について説明します。

- 「SAコア」(48ページ)
- 「SAエージェント」(48ページ)
- 「SAゲートウェイ」(49ページ)
- 「SAサテライト」(49ページ)
- 「SA管理対象サーバー」(49ページ)

SAコア

SAコアは、コアコンポーネントのグループであり、連携して動作します。コアによって、ネットワーク上でサーバーを検出し、これを管理対象サーバープールに追加した後、プロビジョニング、構成、パッチの適用、監視、監査、管理などの作業をSAクライアントインタフェースから一元的に実行できます。SAクライアントでは、1つのインタフェースからSAのすべての情報と管理機能を操作できます。

コアコンポーネントのインストール先となるサーバーは、コアサーバーと呼ばれます。コアコンポーネントは、複数のホストに分散されている場合でも、1つのSAコアの一部として認識されます。コアコンポーネントは、1つのホストにインストールまたは複数のホストに分散できますが、一般的なSAインストールではコアコンポーネントバンドルが使用されます。バンドルとは、パフォーマンスと管理効率の向上を目的に、いくつかのコンポーネントをまとめて同じサーバーにインストールする方法です。

SAでは、いくつかのサーバー管理アクティビティとの通信と管理の目的で、各管理対象サーバーにサーバーエージェントがインストールされます。これにより、管理対象サーバーとの通信を、SAコアコンポーネントの一部であるゲートウェイ経由で行うことができます。またサーバーエージェントは、SAクライアントからユーザーが入力した内容に基づいて、管理対象サーバーでアクションを実行します。

> **注:** FIPSモードでは、SAコンポーネントが適切に開始し機能するために、キャラクターデバイス /dev/randomから得られる十分なエントロピーがコアサーバー上に存在する必要があります。

SAエージェント

SAエージェントは、SAを使用して管理するサーバーにインストールされるインテリジェントソフトウェアです。非管理対象サーバーにエージェントをインストールすると、エージェントがSAコアに登録され、その後サーバーを管理対象サーバープールに追加できます。またSAエージェントは、コマンドをコアから受信し、ローカルサーバーで適切なアクションを実行します。このアクションには、ソフトウェアのインストールと削除、ソフトウェアとハードウェアの構成、サーバーステータスのレポート作成、監査などがあります。

エージェントの登録では、SAは各サーバーに一意のID (マシンID (MID)) を割り当て、このIDをモデルリポジトリに保存します。また、サーバーはMACアドレスでも一意に識別できます。MACアドレスは、ネットワークインタフェースカードに割り当てられている一意の16進数であり、ネットワーク上でのデバイスの物理アドレスとして使用されます。

SAゲートウェイ

SAゲートウェイは、管理対象サーバーとSAコアの間の通信、複数コア間の通信、サテライトインストールとSAコア間の通信を管理します。

ゲートウェイには、次のタイプがあります。

- 管理ゲートウェイ
  このゲートウェイは、SAコア間の通信と、SAコアとサテライト間の通信を管理します。
- コアゲートウェイ/エージェントゲートウェイ
  このゲートウェイは相互に連携して、SAコアとエージェント間の通信を管理します。
- サテライトゲートウェイ
  このゲートウェイは、ユーザー構成に応じて、管理ゲートウェイまたはコアゲートウェイを経由してSAコアと通信します。

SAサテライト

サテライトは、管理対象サーバーの数が少なく完全なSAコアインストールを必要としないリモートサイト向けのソリューションです。サテライトでは、ホストに最小限必要なコアコンポーネントのみをインストールでき、ホストからプライマリコアのデータベースとその他サービスにSAゲートウェイ接続経由でアクセスします。

また、限られたネットワーク接続を使ってプライマリファシリティと接続する場合には、帯域幅の問題を軽減することもできます。サテライトで使用するネットワーク帯域幅の上限となるビットレートを指定することができます。これにより、サテライトのネットワークトラフィックによって、同じパイプ上にある他の重要なシステムのネットワーク帯域幅要件が影響を受けることがなくなります。

一般的に、サテライトの最低構成にはサテライトゲートウェイとソフトウェアリポジトリキャッシュが含まれますが、リモートファシリティでサーバー管理機能をフル装備することも可能です。ソフトウェアリポジトリキャッシュには、サテライト内の管理対象サーバーにインストールされているソフトウェアパッケージのローカルコピーが格納され、サテライトゲートウェイは、プライマリコアとの通信を処理します。

SA管理対象サーバー

SA管理対象サーバーは、SAエージェントがインストールされ、SAでアクティブに管理されているサーバーです。

関連トピック

- 「FIPS 140-2について」
- 「FIPS 140-2準拠テクノロジー」
- 「サポートされているSAコアとサテライトのオペレーティングシステム」
- 「サポートされている管理対象サーバーのオペレーティングシステム」
- 「サポートされているFIPS 140-2セキュリティレベル」
- 「略語」
- 「関連文書」

FIPS 140-2について

Federal Information Processing Standards Publication (FIPS) 140-2「暗号モジュールに関するセキュリティ要件」は、National Institute of Standards and Technology (NIST) が2001年5月に発行したものです。この規格では、取扱注意ではあるが機密ではない情報を保護するセキュリティシステム内で使用される暗号モジュールに関するセキュリティ要件を規定しています。FIPS 140-2は、検証済み暗号モジュールの使用を促進するために米国およびカナダ政府が採用している標準規格の1つです。FIPS 140-2は、連邦機関が調達する機器が規格に準拠し、検証済み暗号モジュールを内蔵していることを評価するためのセキュリティ基準を提供します。

HPE Server Automation (SA) では、FIPS 140-2をサポートするため、FIPS準拠の暗号モジュールを使用しています。

FIPS 140-2準拠テクノロジー

SAは、NISTの認定プロセスに合格した暗号モジュールを使用して、FIPS 140-2への準拠を実現しています。SAは、次のFIPS 140-2準拠テクノロジーを使用します。

NSS暗号モジュール

SAでは、Mozilla Public Licenseによる許可のもとで、オープンソースの汎用暗号化ライブラリである、FIPS 140-2認定のネットワークセキュリティサービス(NSS)暗号モジュールを採用しています。

NSS暗号モジュールには、RSA (EMC Corporationのセキュリティ部門)で発行される、業界標準の公開鍵暗号標準 (PKCS) #11の暗号トークンインタフェースバージョン2.20に基づいた、APIが含まれています。

TLS/SSLトランスポートプロトコル

SAはまた、次世代のSecure Sockets Layer (SSL)である、Transport Layer Security (TLS)も使用しています。

SAプラットフォームは、セキュリティで保護されていないネットワークを介して機密情報を伝送する、複数の分散コンポーネントから構成されています。SSLは、実効性が証明されている業界標準であり、次の機能を提供します。

- データ(イベント/ユーザーインタラクション)の傍受を確実に阻止する暗号化
- 回線上でデータが意図的または偶発的に改変されることを防止するデータ整合性 (MAC)
- 資格情報が回線経由で漏洩することを防止する認証機能

TLSとSSLの機能は同じであるため、これらのプロトコルはまとめてTLS/SSLと呼ばれますが、セキュリティキーの交換に異なるアルゴリズムを使用しています。

SSL 2.0および3.0のプロトコルは、FIPS 140-2に準拠していません。TLSは、SSLを基に開発されたプロトコルの中で、唯一、インターネットエンジニアリングタスクフォース(IETF)標準に基づくFIPS 140-2認定アルゴリズムが組み込まれています。

SHA-1/SHA-2ファミリー

セキュアハッシュアルゴリズムとは、アメリカ国立標準技術研究所 (NIST)によって米国連邦標準規格(FIPS)として公開された一連の暗号ハッシュ関数です。SAはSHA-256を使用しますが、SHA-1およびSHA-2ファミリーの他のハッシュ関数もサポートされます。

サポートされているSAコアとサテライトのオペレーティングシステム

SAコアのFIPS 140-2モードは、サポートされるすべてのSA管理対象プラットフォームで有効にできます。

サポートされている管理対象サーバーのオペレーティングシステム

管理対象サーバーのFIPS 140-2モードは、サポートされるすべてのSA管理対象プラットフォームで有効にできます。ただし、以下の場合を除きます。

- IA 64上のRed Hat Enterprise Linux 5
- S390Xプラットフォーム(Zシリーズ)上のRed Hat Enterprise Linux 5および6
- S390Xプラットフォーム(Zシリーズ)上のSUSE Linux Enterprise Server 10および11
- HPUX PA-RISC 11.11、11.23、11.31
- HPUS IA64 11.11、11.23、11.31
- IA 64上のWindows Server 2008 R2

サポートされているFIPS 140-2セキュリティレベル

FIPS 140-2セキュリティレベル

SAコンポーネント	サポートされているFIPS 140-2セキュリティレベル	NSSのバージョン	OpenSSLのバージョン
SA 10.10以降	レベル1	3.15.1	1.0.1h (2.0.5 FIPSモジュール)

SA暗号モード

SAには、次の2つの暗号モードが用意されています。

- FIPS 140-2モード (取扱注意ではあるが機密ではない情報)
- ESM標準暗号 (デフォルトモード)

FIPS 140-2モード

FIPS 140-2モードでは、取扱注意ではあるが機密ではない情報 (SBU) のセキュリティが有効になります。FIPS 140-2モードは、SAコアに接続してデータを交換するSA関連のすべてコンポーネントに、NSS暗号モジュールがデプロイされ、有効になっていることを意味します。

FIPS 140-2モードは、RSA公開鍵暗号化技術に基づいており、ESMの標準暗号システムとは別のセキュアな暗号化システムです。FIPS 140-2モードを有効にすると、ESMの標準暗号システムは使用されません。

ESM標準暗号化

FIPS 140-2暗号が要件になっていないデプロイメントをサポートするために、SAでは既存の暗号アルゴリズムとキーストア形式が引き続き使用されます。

略語

略語	説明
ESM	Enterprise Security Management (エンタープライズセキュリティ管理)
FIPS	Federal Information Processing Standards (連邦情報処理標準)
HMAC	Keyed-Hash Message Authentication Codes (鍵付きハッシュメッセージ認証コード)
HTTPS	Secure Hypertext Transfer Protocol (over TLS/SSL)
ECDSA	Elliptical Curve Digital Signature Algorithm (楕円曲線デジタル署名アルゴリズム)。最高機密までの情報に対応したSuite Bセキュリティをサポートするために使用されます。
IDS	Intrusion Detection System (侵入検知システム)
IEC	International Electrotechnical Commission (国際電気標準会議)

IETF	Internet Engineering Task Force (インターネット技術タスクフォース)
ISO	International Organization for Standardization (国際標準化機構)
MD5	Message-Digest Algorithm 5 (メッセージダイジェストアルゴリズム5)
NIST	National Institute of Standards and Technology (米国国立標準技術研究所)
NSA	National Security Agency (国家安全保障局)
NSS	Network Security Services (ネットワークセキュリティサービス)
PKCS	Public Key Cryptography Standards (公開鍵暗号化標準)
RSA	RSA Security, Inc. (EMC Corporationのセキュリティ部門) によって開発された公開鍵暗号化テクノロジー 。RSAは、この方式の発明者であるRivest、 Shamir、Adelmanの頭文字を表しています。
SBU	Sensitive But Unclassified (取扱注意ではあるが機密ではない)。ある種の暗号方式で保護する情報を指します。
SHA	Secure Hash Algorithm (セキュアハッシュアルゴリズム)
SSL	Secure Sockets Layer (TLSに関連)
TSL	Transport Security Layer (次世代のSSL)
W3C	World Wide Web Consortium (ワールドワイドウェブコンソーシアム)

関連文書

FIPS 140-2規格、OpenSSL暗号モジュール、および関連するテクノロジーの詳細については、以下のリソースを参照してください。

関連文書

トピック	リソース
FIPS PUB 140-2	米国国立標準技術研究所 (NIST) の情報技術ラボラトリが公開している情報処理規格 (FIPS) に関する文書。2001年5月25日発行。 http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
OpenSSL Cryptographic Module	OpenSSL.orgが公開しているOpenSSL Cryptographic Moduleバージョン0.9.8jのFIPS 140-2 Non-Proprietary Security Policyレベル1およびレベル2検証。 http://www.openssl.org/docs/fips/fipsvalidation.html http://www.openssl.org/docs/fips/UserGuide-2.0.pdf

関連文書 (続き)

トピック	リソース
Approved Cryptographic Modules	NISTによって承認されたすべての暗号モジュールのリスト。 http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140valall.htm
PKCS #11	公開鍵暗号化標準 (PKCS) #11に関する文書。デバイスの独立性および複数のデバイスにアクセスする複数のアプリケーション間でのリソース共有を可能にする暗号トークンインタフェースAPIについて説明。 http://www.rsa.com/rsalabs/node.asp?id=2133
Transport Layer Protocol (TLS)	次世代Secure Sockets Layer (SSL) であるTransport Layer Protocol (TLS) に関する概要説明。 http://en.wikipedia.org/wiki/Transport_Layer_Security TLSの実装の方法および理由に関する注意: Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementation (2005年、NISTにより公開): http://csrc.nist.gov/publications/nistpubs/800-52/SP800-52.pdf
Internet Engineering Task Force (IETF)	TLSプロトコルを策定した団体で、ワールドワイドウェブコンソーシアム (W3C)、国際標準化機構 (ISO)、国際電気標準会議 (IEC) により承認されたインターネット規格の利用促進を行っているIETFの概要。 http://en.wikipedia.org/wiki/IETF IETFのWebサイト: http://www.ietf.org/

ドキュメントのフィードバックを送信

本ドキュメントについてのご意見、ご感想については、電子メールでドキュメント制作チームまでご連絡ください。このシステムに電子メールクライアントが設定されている場合は、上記のリンクをクリックすると、次の情報が件名行に記載された電子メールウィンドウが開きます。

フィードバック: 主要コンセプトガイド (Server Automation 10.50)

フィードバックを追加して [送信] をクリックしてください。

電子メールクライアントが使用できない場合は、Webメールクライアントのメッセージに上記の情報をコピーし、hpe_sa_docs@hpe.com までフィードバックをお送りください。

ご協力をお願いいたします。