

SiteMinder

Apache サーバ用 Web エージェント インス トール ガイド

12.52 SP1



このドキュメント（組み込みヘルプシステムおよび電子的に配布される資料を含む、以下「本ドキュメント」）は、お客様への情報提供のみを目的としたもので、日本 CA 株式会社（以下「CA」）により随時、変更または撤回されることがあります。本ドキュメントは、CA が知的財産権を有する機密情報であり、CA の事前の書面による承諾を受けずに本書の全部または一部を複写、譲渡、変更、開示、修正、複製することはできません。

本ドキュメントで言及されている CA ソフトウェア製品のライセンスを受けたユーザは、社内でユーザおよび従業員が使用する場合に限り、当該ソフトウェアに関連する本ドキュメントのコピーを妥当な部数だけ作成できます。ただし、CA のすべての著作権表示およびその説明を当該複製に添付することを条件とします。

本ドキュメントを印刷するまたはコピーを作成する上記の権利は、当該ソフトウェアのライセンスが完全に有効となっている期間内に限定されます。いかなる理由であれ、上記のライセンスが終了した場合には、お客様は本ドキュメントの全部または一部と、それらを複製したコピーのすべてを破棄したことを、CA に文書で証明する責任を負いません。

準拠法により認められる限り、CA は本ドキュメントを現状有姿のまま提供し、商品性、特定の使用目的に対する適合性、他者の権利に対して侵害のないことについて、黙示の保証も含めいかなる保証もしません。また、本ドキュメントの使用に起因して、逸失利益、投資損失、業務の中断、営業権の喪失、情報の喪失等、いかなる損害（直接損害か間接損害かを問いません）が発生しても、CA はお客様または第三者に対し責任を負いません。CA がかかる損害の発生の可能性について事前に明示に通告されていた場合も同様とします。

本ドキュメントで参照されているすべてのソフトウェア製品の使用には、該当するライセンス契約が適用され、当該ライセンス契約はこの通知の条件によっていかなる変更も行われません。

本書の制作者は CA および CA Inc. です。

「制限された権利」のもとでの提供：アメリカ合衆国政府が使用、複製、開示する場合は、FAR Sections 12.212、52.227-14 及び 52.227-19(c)(1)及び(2)、ならびに DFARS Section 252.227-7014(b)(3) または、これらの後継の条項に規定される該当する制限に従うものとします。

Copyright © 2014 CA. All rights reserved. 本書に記載されたすべての商標、商号、サービス・マークおよびロゴは、それぞれの各社に帰属します。

CA Technologies 製品リファレンス

このマニュアルが参照している CA Technologies の製品は以下のとおりです。

- SiteMinder
- CA IdentityMinder™ (旧 CA Identity Manager)
- eTrust SOA Security Manager (旧 CA SOA Security Manager)

CA への連絡先

テクニカルサポートの詳細については、弊社テクニカルサポートの Web サイト (<http://www.ca.com/jp/support/>) をご覧ください。

マニュアルの変更点

以下のドキュメントの更新は、本書の最新のリリース以降に行われたものです。

- [Apache ベースのサーバが開始しない](#) (P. 101) - Windows の Apache ベース サーバ 2.4.x のサポートに関連するトラブルシューティング情報を追加しました。
- [Windows の設定プログラムが必要とする情報を収集](#) (P. 29) - 設定ウィザード情報を改訂し、Windows の Apache ベース サーバ 2.4.x 以降をサポートするために追加された設定を追加しました。
- [Web エージェントのアンインストール](#) (P. 93) - サポートされている JRE のインストールおよびシステムパスへの包含に関連する古いデータを削除しました。JRE は現在、ソフトウェアに含まれています。
- [UNIX/Linux でエージェントを設定する方法](#) (P. 42) - RedHat Linux に含まれている Apache ベースの埋め込み Web サーバの設定プロシージャを改定しました。CQ175578 および STAR イシュー 21482842-01 を解決します。
- [apachectl コマンドによる Oracle 11g HTTP サーバの起動](#) (P. 71) - Oracle 11g HTTP サーバの起動プロシージャを追加しました。CQ176028 を解決します。

目次

| | |
|---|-----------|
| 第 1 章: 準備 | 9 |
| このガイドでは Apache ベースの Web サーバ手順のみ | 9 |
| SiteMinder エージェントのハードウェア要件 | 10 |
| Apache ベースの Web サーバ用準備ロードマップ | 11 |
| Apache ベースのサーバ上の Web エージェントインストールの準備をする方法 | 12 |
| プラットフォーム サポート マトリックスへのアクセス | 12 |
| Windows での Apache ベース サーバの準備 | 13 |
| UNIX での Apache ベース サーバの準備 | 14 |
| Linux での Apache ベース サーバの準備 | 16 |
| z/OS の準備 | 19 |
| IBM HTTP Server の準備 | 22 |
| ポリシー サーバ要件 | 23 |
| | |
| 第 2 章: Apache ベース エージェントの Windows へのインストールおよび設定 | 27 |
| エージェントインストールとエージェント設定の比較 | 27 |
| Apache ベース エージェントを Windows にインストールする方法 | 28 |
| インストールプログラム用の情報を収集します。 | 28 |
| Windows 上でのインストールプログラムの実行 | 28 |
| Windows で Apache ベース エージェントを設定する方法 | 29 |
| Windows 上の設定プログラムが必要とする情報を収集する | 29 |
| Windows 上での Web エージェント設定プログラムの実行 | 33 |
| Windows 上で後続のエージェントに対して無人またはサイレント インストールおよび設定プログラムを実行する | 35 |
| | |
| 第 3 章: Apache ベースのエージェントの UNIX/Linux へのインストールおよび設定 | 37 |
| エージェントインストールとエージェント設定の比較 | 37 |
| Apache ベース エージェントを UNIX または Linux にインストールする方法 | 38 |
| インストール用の情報の収集 | 38 |
| UNIX/Linux 上でのインストールプログラムの実行 | 39 |
| z/OS システム上のエージェントをインストールする方法 | 40 |
| インストール用の情報の収集 | 40 |

| | |
|---|----|
| z/OS で SiteMinder エージェント インストールプログラムの実行 | 40 |
| UNIX または Linux で Apache ベース エージェントを設定する方法 | 42 |
| 設定プログラムが UNIX/Linux 上で必要とする情報の収集 | 42 |
| RedHat Linux 上の埋め込まれた Apache Web サーバ用の設定ファイルの編集 | 46 |
| UNIX または Linux でのエージェント環境スクリプトの取得 | 46 |
| UNIX または Linux でのライブラリ パス変数の設定 | 47 |
| UNIX/Linux 上での Web エージェント設定プログラムの実行 | 48 |
| SiteMinder デフォルト ヘッダの参照 | 49 |
| UNIX/Linux 上でエージェントに対して無人またはサイレント インストールおよび設定プログラムを実行する | 50 |
| LD_PRELOAD 変数の設定 | 51 |
| SuSE Linux 9 for zSeries 上の Apache エージェントに LD_ASSUME_KERNEL を設定する | 52 |
| Red Hat Linux システム用の CAPKIHOM 変数の設定 | 52 |
| z/OS システム上のエージェントを設定する方法 | 53 |
| 設定プログラムが z/OS 上で必要とする情報を収集する | 53 |
| z/OS 上でライブラリ パス変数を設定する | 56 |
| z/OS 上で SiteMinder エージェント設定プログラムを実行する | 57 |
| (オプション) z/OS 上で SiteMinder エージェント用の無人またはサイレント インストールおよび設定プログラムを実行する | 58 |
| UNIX/Linux でのオプションのエージェント設定 | 60 |
| apachectl スクリプトを使用する場合の Web エージェント変数の設定 | 61 |
| オプションの httpd.conf ファイル変更でサーバのパフォーマンスを改善する | 61 |

第 4 章: ダイナミック ポリシー サーバクラスタ 63

| | |
|---|----|
| ダイナミック ポリシー サーバクラスタへの Web エージェントの接続 | 65 |
|---|----|

第 5 章: Web エージェントの起動と停止 67

| | |
|--|----|
| Web エージェントの有効化 | 67 |
| Web エージェントの無効化 | 68 |
| apachectl コマンドによるほとんどの Apache ベース エージェントの起動または停止 | 69 |
| apachectl コマンドによる IBM HTTP Server の起動 | 70 |
| opmnctl コマンドによる Oracle 11.g.x HTTP サーバの起動 | 71 |

第 6 章: Web エージェントを 12.52 SP1 にアップグレードする 73

| | |
|---|----|
| エージェントアップグレードロードマップ | 74 |
| SiteMinder エージェントのアップグレードの準備をする方法 | 75 |
| SiteMinder r6.x からのアップグレード処理 | 76 |
| SiteMinder r12.0 からのアップグレード処理 | 77 |

| | |
|--|------------|
| SiteMinder r12.0 からのアップグレード処理..... | 78 |
| LD_PRELOAD 変数が既存のエージェントと矛盾しないことを確認する | 79 |
| UNIX および Linux オペレーティング環境での環境スクリプトの取得..... | 79 |
| Windows 上でエージェントをアップグレードするためにインストール ウィザードを実行する..... | 80 |
| UNIX/Linux 上でエージェントをアップグレードするためにインストール ウィザードを実行する | 81 |
| UNIX/Linux 上でアップグレードしたエージェントを設定する前にライブラリ パス変数を設定する | 82 |
| Windows 上でアップグレードしたエージェントを設定する | 83 |
| UNIX/Linux 上でアップグレードしたエージェントを設定する | 84 |
| 第 7 章: エージェントに対するオペレーティング システムの調整 | 87 |
| 共有メモリ セグメントの調整..... | 88 |
| Solaris 10 リソース管理を調整する方法..... | 91 |
| 第 8 章: Web エージェントのアンインストール | 93 |
| Web エージェントのアンインストールに関する注意事項..... | 93 |
| Windows オペレーティング環境からの Web エージェントのアンインストール | 94 |
| UNIX システムからの Apache ベースのエージェントのアンインストール | 97 |
| z/OS システムからの Apache ベース エージェントのアンインストール | 98 |
| 第 9 章: トラブルシューティング | 101 |
| 開始しない Apache ベースのサーバ..... | 101 |
| 起動時に Apache サーバに shmget エラーが表示される..... | 102 |
| Apache エージェントは有効になっているがデフォルトのサーバ ページまたは保護されたリソースにアクセスできない..... | 102 |
| Apache Web エージェントが動作しない..... | 102 |
| 英語以外の入力文字にジャンク文字が含まれる | 103 |
| 付録 A: ワークシート | 105 |
| エージェントインストールワークシート | 105 |
| エージェント設定ワークシート | 105 |

第 1 章: 準備

このセクションには、以下のトピックが含まれています。

[このガイドでは Apache ベースの Web サーバ手順のみ \(P. 9\)](#)

[SiteMinder エージェントのハードウェア要件 \(P. 10\)](#)

[Apache ベースの Web サーバ用準備ロードマップ \(P. 11\)](#)

[Apache ベースのサーバ上の Web エージェントインストールの準備をする方法 \(P. 12\)](#)

[ポリシーサーバ要件 \(P. 23\)](#)

このガイドでは Apache ベースの Web サーバ手順のみ

このガイドでは、SiteMinder エージェントを Apache ベースの Web サーバにインストールまたは設定するための手順についてのみ説明します。

SiteMinder エージェントを他のタイプの Web サーバまたはオペレーティング環境にインストールまたは設定するには、以下のいずれかのガイドを参照してください。

- Domino 用 Web エージェントインストールガイド。
- IIS 用 Web エージェントインストールガイド。
- Oracle iPlanet 用 Web エージェントインストールガイド。

SiteMinder エージェントのハードウェア要件

SiteMinder エージェントをホストするコンピュータには以下のハードウェアが必要です。

Windows オペレーティング環境要件

Windows オペレーティング環境で稼働するエージェントは以下のハードウェアを必要とします。

- CPU : x86 または x64
- メモリ : 2 GB のシステム RAM。
- 使用可能なディスク領域 :
 - インストール場所に 2 GB の空きディスク領域。
 - 一時的な場所に .5 GB の空きディスク領域。

UNIX オペレーティング環境要件

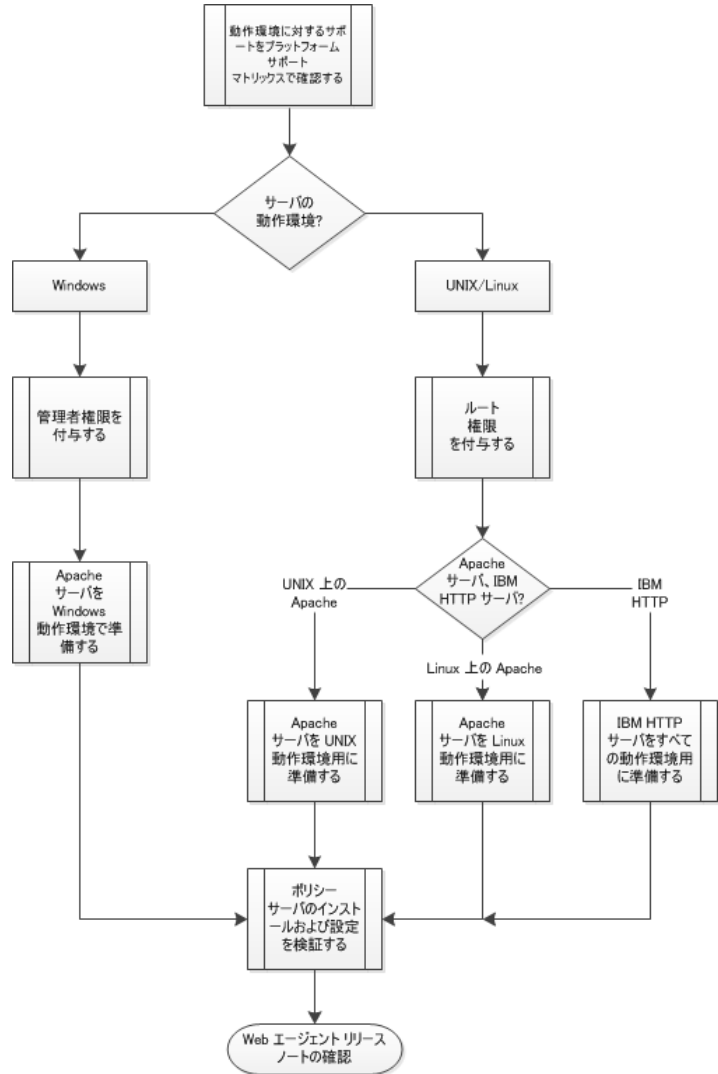
UNIX オペレーティング環境で稼働するエージェントは以下のハードウェアを必要とします。

- CPU :
 - Solaris オペレーティング環境 : SPARC
 - Red Hat オペレーティング環境 : x86 または x64
- メモリ : 2 GB のシステム RAM。
- 使用可能なディスク領域 :
 - インストール場所に 2 GB の空きディスク領域。
 - /tmp 内に .5 GB の空きディスク領域。

注: エージェントの日常運用には、/tmp 内に 10 MB の空きディスク領域が必要です。エージェントは、/tmp の下にファイルおよび名前付きパイプを作成します。これらのファイルおよびパイプが作成されたパスは変更できません。

Apache ベースの Web サーバ用準備ロードマップ

以下の図は、SiteMinder エージェントをインストールする前に Web サーバを準備する方法を示しています。



Apache ベースのサーバ上の Web エージェント インストールの準備をする方法

Apache ベース サーバ上の SiteMinder エージェント インストールの準備をするには、以下の手順を使用します。

1. [プラットフォーム サポート マトリックスを見つけます](#) (P. 12)。インストールする SiteMinder エージェントのバージョンを Web サーバがサポートしていることを確認します。
2. Web サーバに対して以下のいずれかのタイプの権限を持つアカウントがあることを確認します。
 - 管理者権限 (Windows オペレーティング環境)
 - ルート権限 (UNIX または Linux オペレーティング環境)
3. 以下のリストのいずれかを使用して、エージェントが必要とする適切な追加の SiteMinder を設定します。
 - Windows オペレーティング環境での Apache ベース サーバ準備。
 - [UNIX オペレーティング環境での Apache ベース サーバ準備](#) (P. 14)。
 - [Linux オペレーティング環境での Apache ベース サーバ準備](#) (P. 16)。
 - [すべてのオペレーティング環境での IBM HTTP サーバ準備](#) (P. 22)。
4. ポリシーサーバがインストールおよび設定されていることを確認します。
5. 「Web エージェント リリース ノート」の既知の問題セクションを確認します。

プラットフォーム サポート マトリックスへのアクセス

プラットフォーム サポート マトリックスを使用して、オペレーティング環境および他の必要なサードパーティ コンポーネントがサポートされていることを確認します。

次の手順に従ってください:

1. CA サポート サイトに移動します。
2. [Product Pages] をクリックします。
3. 製品名を入力し、Enter キーを押します。

4. ポピュラーなリンクを開き、[Informational Documentation Index] をクリックします。
5. [Platform Support Matrices] をクリックします。

注: 最新の JDK および JRE バージョンは、[Oracle Developer Network](#) でダウンロードできます。

技術パートナーおよび CA によって検証済みの製品

パートナーおよびパートナーによって検証済み製品の最新[リスト](#)

Windows での Apache ベース サーバの準備

Windows オペレーティング環境上で実行される Apache ベースのサーバは、SiteMinder エージェントをインストールする前に以下の準備を必要とします。

1. [すべてのユーザのためのサービスとして Apache ベース Web サーバをインストールする](#) (P. 13)。
2. [適切な許可を持つログサブディレクトリが存在することを確認する](#) (P. 14)。

すべてのユーザのためのサービスとして Windows に Apache Web サーバをインストールする

単一ユーザアカウントを使用して Apache ベースの Web サーバがインストールされた場合、エージェント設定は Apache ベースの Web サーバインストールを検出することができません。

この問題を修正するには、Apache ベースの Web サーバを Windows オペレーティング環境にインストールする場合に以下のオプションを選択します。

サービスとしてインストール、すべてのユーザが使用可能。

Apache ベースの Web エージェントに対する許可でログ サブディレクトリの存在を検証する

Apache ベースの Web サーバ (IBM HTTP サーバを含む) 用の SiteMinder エージェントでは、ログ サブディレクトリが Apache ベースの Web サーバのルートディレクトリの下に存在する必要があります。このサブディレクトリにはユーザ ID の読み取りおよび書き込み許可が必要で、この下で Apache の子プロセスが実行されます。

ログ サブディレクトリが存在しない場合は、必要とされる許可でそれを作成します。

注: この設定要件は、Apache ルートディレクトリの外側にログファイルを書き込む任意の Apache ベースの Web サーバに適用されます。

UNIX での Apache ベース サーバの準備

UNIX オペレーティング環境上で実行される Apache ベースのサーバは、SiteMinder エージェントをインストールする前に以下の準備を必要とします。

1. [display 変数を設定する](#) (P. 14)。
2. [ログ サブディレクトリの存在を確認する](#) (P. 15)。
3. 適切なパッチがオペレーティング環境にインストールされていることを確認します。
 - [Solaris パッチ](#) (P. 15)。
 - [AIX の要件](#) (P. 15)。

UNIX 上で SiteMinder エージェント インストールに DISPLAY を設定する

Telnet や Exceed 端末などのリモート端末から UNIX システムに SiteMinder エージェントをインストールする場合は、DISPLAY 変数がローカルシステムに設定されていることを確認してください。たとえば、マシンが 111.11.1.12 である場合は、以下のように変数を設定します。

```
DISPLAY=111.11.1.12: 0.0
```

```
export DISPLAY
```

注: また、コンソールモードインストールを使用して、エージェントをインストールすることもできます。これには X ウィンドウ表示モードは必要ありません。

ログ サブディレクトリと Apache 用 SiteMinder エージェントのアクセス許可の確認

Apache ベースの Web サーバ (IBM HTTP Server を含む) 用のエージェントでは、ログ サブディレクトリが Apache ベースの Web サーバのルートディレクトリの下に存在する必要があります。このサブディレクトリにはユーザ ID の読み取りおよび書き込み許可が必要で、この下で Apache の子プロセスが実行されます。

ログ サブディレクトリが存在しない場合は、必要とされる許可でそれを作成します。

注: この設定要件は、Apache ルート ディレクトリの外側にログ ファイルを書き込む任意の Apache ベースの Web サーバに適用されます。

必要とされる Solaris パッチ

Solaris コンピュータに SiteMinder エージェントをインストールする前に、以下のパッチをインストールします。

Solaris 9

パッチ 111711-16 が必要です。

Solaris 10

パッチ 119963-08 が必要です。

インストールされたパッチバージョンは、ルート ユーザとしてログインして以下のコマンドを実行することにより確認できます。

```
showrev -p | grep patch_id
```

Solaris パッチを検索するには、「Oracle Solution Center」に移動します。

AIX 要件

AIX システム上で実行する SiteMinder エージェントには以下のコンポーネントが必要です。

- AIX システム上で再設計された (フレームワーク) SiteMinder Apache ベースのエージェントを実行するには、C/C++ ランタイム環境がバージョン 8.0.0.0 である必要があります。

Linux での Apache ベース サーバの準備

Linux オペレーティング環境上で実行される Apache ベースのサーバは、SiteMinder エージェントをインストールする前に以下の準備を必要とします。

1. 必要なパッチがインストールされていることを確認します。
2. 必要なライブラリがインストールされていることを確認します。
3. [必要なツールがインストールされていることを確認する](#) (P. 18)。
4. [Apache ベースの Web サーバをコンパイルする](#) (P. 18)。
5. [ログ サブディレクトリの存在を確認する](#) (P. 19)。

必要とされる Linux パッチ

以下の Linux パッチが必要です。

64 ビット Linux システムで実行する Web エージェント用

- Binutils 2.17
- GCC 4.1.0

必要とされる Linux ライブラリ

Linux オペレーティング環境上で動作するコンポーネントには、特定のライブラリ ファイルが必要です。正しいライブラリをインストールしないと、以下のエラーを引き起こす場合があります。

```
java.lang.UnsatisfiedLinkError
```

このコンポーネントの Linux バージョンをインストール、設定、またはアップグレードする場合は、ホスト システム上で以下のパッケージが必要になります。

Red Hat 5.x

- `compat-gcc-34-c++-3.4.6-patch_version.l386`
- `libstdc++-4.x.x-x.el5.i686.rpm`
- `libidn.so.11.rpm`
- `ncurses`

Red Hat 6.x

- libstdc++-4.x.x-x.el6.i686.rpm
- libidn-1.18-2.el6.i686
- libXext.i686.rpm
- libXrender.i686.rpm
- libXtst.i686.rpm
- libidn.so.11.rpm
- ncurses

Red Hat 6.x (64 ビット) の場合はさらに以下 :

注: 64 ビット Red Hat 6.x に必要な RPM パッケージはすべて、32 ビットの
パッケージです。

- libXau-1.0.5-1.el6.i686.rpm
- libxcb-1.5-1.el6.i686.rpm
- compat-db42-4.2.52-15.el6.i686.rpm
- compat-db43-4.3.29-15.el6.i686.rpm
- libX11-1.3-2.el6.i686.rpm
- libXrender-0.9.5-1.el6.i686.rpm
- libexpat.so.1 (expat-2.0.1-11.el6_2.i686.rpm により提供)
- libfreetype.so.6 (freetype-2.3.11-6.el6_2.9.i686.rpm により提供)
- libfontconfig.so.1 (fontconfig-2.8.0-3.el6.i686.rpm により提供)
- libICE-1.0.6-1.el6.i686.rpm
- libuuid-2.17.2-12.7.el6.i686.rpm
- libSM-1.1.0-7.1.el6.i686.rpm
- libXext-1.1-3.el6.i686.rpm
- compat-libstdc++-33-3.2.3-69.el6.i686.rpm
- compat-db-4.6.21-15.el6.i686.rpm
- libXi-1.3-3.el6.i686.rpm
- libXtst-1.0.99.2-3.el6.i686.rpm
- libXft-2.1.13-4.1.el6.i686.rpm
- libXt-1.0.7-1.el6.i686.rpm

- libXp-1.0.0-15.1.el6.i686.rpm
- libstdc++.i686.rpm
- compat-libtermcap.rpm
- libidn.i686.rpm
- ncurses

必要とされる Linux ツール

Red Hat Enterprise Linux オペレーティング環境で実行中の Red Hat Apache 2.2 Web サーバに SiteMinder エージェントをインストールする前に、Red Hat Legacy Software Development ツール パッケージに含まれるすべてのアイテムをインストールします。

Linux システムでの Apache Web サーバのコンパイル

SiteMinder エージェントが Linux を実行する Apache Web サーバで動作するには、サーバをコンパイルする必要があります。エージェント コードは pthreads (POSIX 準拠のスレッドルーチンのライブラリ) を使用しますが、Linux プラットフォームの Apache サーバは、デフォルトでは使用しないため、コンパイルが必要です。

lpthread オプションでコンパイルしない場合、Apache サーバは開始しますが、その後ハングアップし、リクエストを処理しません。Linux の Apache サーバは、Linux の動的なローダに問題があるために pthreads を使用するモジュールを初期化できません。

次の手順に従ってください:

1. 以下のコマンドを入力します。

```
LIBS=-lpthread
export LIBS
```

2. 以下を入力して、Apache を通常の方法で設定します。

```
configure --enable-module=so --prefix=your_install_target_directory
make
make install
```

ログ サブディレクトリと Apache 用 SiteMinder エージェントのアクセス許可の確認

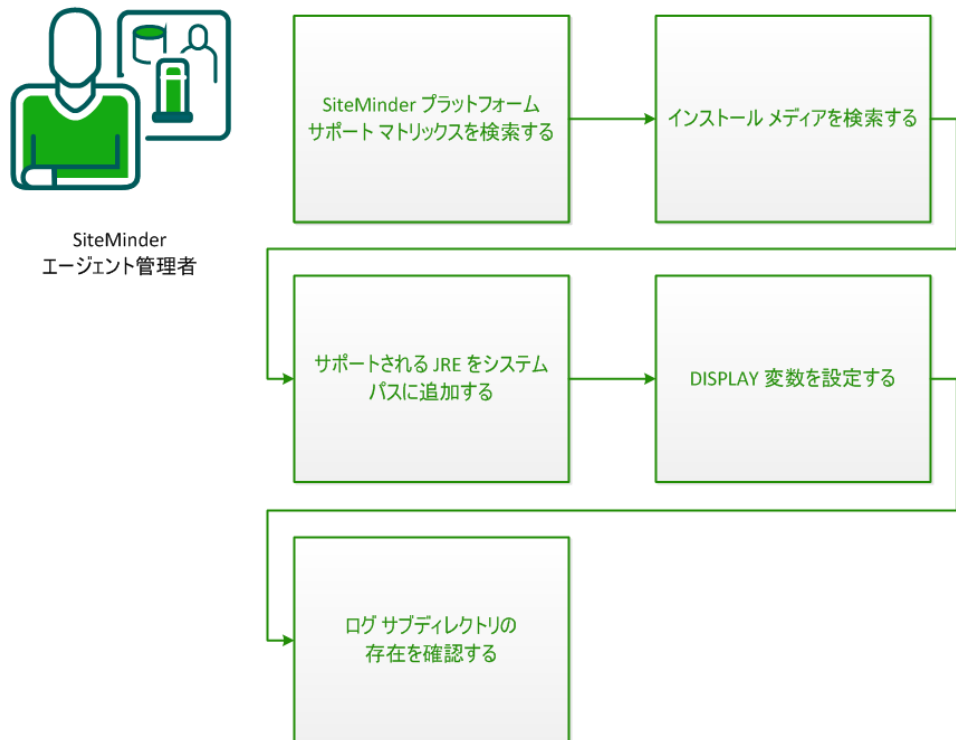
Apache ベースの Web サーバ（IBM HTTP Server を含む）で実行されるエージェントでは、ログ サブディレクトリが Apache ベースの Web サーバのルート ディレクトリの下に存在する必要があります。このサブディレクトリにはユーザ ID の読み取りおよび書き込み許可が必要で、この下で Apache の子プロセスが実行されます。

ログ サブディレクトリが存在しない場合は、必要とされる許可でそれを作成します。

注: この設定要件は、Apache ルート ディレクトリの外側にログ ファイルを書き込む任意の Apache ベースの Web サーバに適用されます。

z/OS の準備

SiteMinder エージェントを z/OS オペレーティング環境にインストールおよび設定する前に、このプロセスに説明される準備手順を実行します。



1. [SiteMinder プラットフォーム サポート マトリックスへアクセスする \(P. 12\)](#)。
2. [インストールメディアを検索する \(P. 20\)](#)。
3. [システムパスへサポートされる JRE を追加する \(P. 22\)](#)。
4. [DISPLAY 変数を設定する \(P. 21\)](#)。
5. [ログ サブディレクトリの存在を確認する \(P. 15\)](#)。

プラットフォーム サポート マトリックスへのアクセス

プラットフォーム サポート マトリックスを使用して、オペレーティング環境および他の必要なサードパーティ コンポーネントがサポートされていることを確認します。

次の手順に従ってください:

1. CA サポート サイトに移動します。
2. [Product Pages] をクリックします。
3. 製品名を入力し、Enter キーを押します。
4. ポピュラーなリンクを開き、[Informational Documentation Index] をクリックします。
5. [Platform Support Matrices] をクリックします。

注: 最新の JDK および JRE バージョンは、[Oracle Developer Network](#) でダウンロードできます。

技術パートナーおよび CA によって検証済みの製品

パートナーおよびパートナーによって検証済み製品の最新[リスト](#)

インストール メディアの検索

ベース リリースが必要な場合は、以下の手順に従います。

1. CA サポート サイトに移動し、[Product Pages] をクリックします。
2. 製品名を入力し、Enter キーを押します。
3. [Quick Access] を開き、[Download Center] をクリックします。
4. ログインします。
5. [Select a Product] リストで製品を探します。

6. リリースと gen レベルを選択します。 [Go] をクリックします。
7. インストール zip をローカルに保存し、一時的な保存場所にキットを解凍します。

累積リリース(cr)が必要な場合は、以下の手順に従います。

1. CA サポート サイトに移動し、 [Product Pages] をクリックします。
2. 製品名を入力し、Enter キーを押します。
3. [Quick Access] を開き、 [Hotfix/Cumulative Release Index] をクリックします。
4. ログインします。
5. 必要なリリースをクリックします。
6. インストール zip をローカルに保存し、一時的な保存場所にキットを解凍します。

z/OS 上で SiteMinder エージェント インストール用の DISPLAY 変数を設定する

リモート端末から SiteMinder エージェントを z/OS システムにインストールしている場合は、DISPLAY 変数がローカルシステムに対して設定されることを確認します。たとえば、サーバの IP アドレスが 111.11.1.12 である場合は、以下のように変数を設定します。

```
export DISPLAY=111.11.1.12:0.0
```

注: また、コンソールモードインストールを使用して、SiteMinder エージェントをインストールすることもできます。これには X ウィンドウ表示モードは必要ありません。

ログ サブディレクトリと Apache 用 SiteMinder エージェントのアクセス許可の確認

Apache ベースの Web サーバ (IBM HTTP Server を含む) 用のエージェントでは、ログ サブディレクトリが Apache ベースの Web サーバのルートディレクトリの下に存在する必要があります。このサブディレクトリにはユーザ ID の読み取りおよび書き込み許可が必要で、この下で Apache の子プロセスが実行されます。

ログ サブディレクトリが存在しない場合は、必要とされる許可でそれを作成します。

注: この設定要件は、Apache ルートディレクトリの外側にログファイルを書き込む任意の Apache ベースの Web サーバに適用されます。

システムパスへサポートされる JRE を追加する

z/OS システム上で SiteMinder エージェントをインストールする前に、サポートされた JRE がシステム上に存在し、PATH と JAVA_HOME システム変数で定義されていることを確認します。

次の手順に従ってください:

コマンドプロンプトで以下のコマンドを入力します。

```
export PATH=JRE/bin:$PATH
export JAVA_HOME=JRE
```

JRE

JRE の場所を指定します。

例 : /sys/java64bt/v6r0m1/usr/lpp/java/*Jversion_number*

IBM HTTP Server の準備

Apache ベースの IBM HTTP サーバは、SiteMinder エージェントをインストールする前に以下の準備を必要とします。

- [サーバログに対する書き込み許可が存在することを確認する](#) (P. 22)。

IBM HTTP サーバログの書き込み許可の有効化

IBM HTTP Server に SiteMinder エージェントをインストールした場合、この Web サーバはルートとしてインストールされ、そのサブディレクトリはすべてのグループのすべてのユーザに書き込み許可を付与しません。

Low Level Agent Worker Process (LLAWP) が Web サーバログにエージェントの初期化メッセージを書き込むには、Web サーバを実行するユーザは、Web サーバのログディレクトリへの書き込み許可が必要です。このユーザに書き込み許可を付与していることを確認します。

ポリシー サーバ要件

以下の条件を確認します。

- ポリシー サーバがインストールされ設定されていることを確認します。
- ポリシー サーバは、エージェントのインストールを計画しているコンピュータと通信できます。

SiteMinder エージェントをインストールおよび設定するには、ポリシー サーバは少なくとも以下のアイテムを必要とします。

- トラステッドホストを登録する権限がある SiteMinder 管理者。

トラステッドホストとは、1つ以上の SiteMinder エージェントがインストールされ、ポリシー サーバに登録されているクライアントコンピュータです。SiteMinder 管理者には、トラステッドホストをポリシー サーバに登録する権限が必要です。トラステッドホストを登録すると、ポリシー サーバ上に一意のトラステッドホスト名オブジェクトが作成されます。

- エージェント識別情報

エージェント識別情報は、ポリシー サーバとエージェントをホストする Web サーバインスタンスの名前または IP アドレスとの間のマッピングを確立します。管理 UI 内にエージェント オブジェクトからのエージェント ID を定義します。そのエージェント ID に名前を割り当てて、Web エージェントとしてエージェントタイプを指定します。

- ホスト設定オブジェクト (HCO)

ポリシー サーバ上のホスト設定オブジェクトによって、初期接続後の、エージェントとポリシー サーバとの間の通信を定義します。初回接続では、SmHost.conf ファイル内のパラメータを使用します。

- エージェント設定オブジェクト (ACO)

このオブジェクトには、エージェント設定を定義するパラメータが含まれています。すべての SiteMinder エージェントで、ACO で定義される以下の設定パラメータのうち少なくとも 1 つが必要です。

AgentName

Web エージェントの ID を定義します。この ID は、エージェントをホストしている各 Web サーバインスタンスの名前と IP アドレスまたは FQDN をリンクします。

以下のイベントのいずれか発生した場合は、DefaultAgentName の値が AgentName パラメータの代わりに使用されます。

- AgentName パラメータが無効。
- AgentName パラメータの値が空。
- AgentName パラメータの値が既存のエージェント オブジェクトに一致しない。

注: このパラメータは複数の値を持つことができます。エージェント設定オブジェクトでこのパラメータを設定する場合は、複数値オプションを使用します。ローカル設定ファイルについては、ファイル内の個別の行に各値を追加します。

デフォルト: デフォルトなし

制限: 複数の値が許可されていますが、各 AgentName パラメータは 4,000 文字に制限されています。文字をパラメータ名に追加することにより、必要に応じて追加の AgentName パラメータを作成します。たとえば、AgentName、AgentName1、AgentName2 などを作成します。

制限: 32-127 の範囲内に 7 ビット ASCII 文字が含まれている必要があり、1 つ以上の印刷可能文字が含まれている必要があります。アンパサンド (&) およびアスタリスク (*) 文字は含めることができません。この値は大文字と小文字が区別されます。たとえば、MyAgent と myagent という名前は、同じように処理されます。

例: myagent1,192.168.0.0 (IPv4)

例: myagent2, 2001:DB8::/32 (IPv6)

例: myagent,www.example.com

例 (複数の AgentName パラメータ): AgentName1、AgentName2、AgentName3。各 AgentName 数値 パラメータの値は、4,000 文字に制限されています。

DefaultAgentName

要求を処理するためにエージェントが使用する名前を定義します。エージェント名値が **AgentName** パラメータに存在しないときは、**DefaultAgentName** の値が IP アドレスまたはインターフェース上の要求に使用されます。

仮想サーバを使用している場合は、**DefaultAgentName** を使用することにより **SiteMinder** 環境を迅速にセットアップできます。**DefaultAgentName** を使用することは、各仮想サーバに対して個別のエージェントを定義する必要がないことを意味します。

重要: **DefaultAgentName** パラメータの値を指定しない場合、**AgentName** パラメータの値にはそのリスト内のすべてのエージェント ID が必要です。そうでない場合、ポリシー サーバはエージェントにポリシーを結び付けることができません。

デフォルト: デフォルトなし

制限: 複数の値を指定できます。

制限: 32-127 の範囲内に 7 ビット ASCII 文字が含まれている必要があり、1 つ以上の印刷可能文字が含まれている必要があります。アンパサンド (&) およびアスタリスク (*) 文字は含めることができません。この値は大文字と小文字が区別されます。たとえば、**MyAgent** と **myagent** という名前は、同じように処理されます。

第 2 章: Apache ベース エージェントの Windows へのインストールおよび設定

このセクションには、以下のトピックが含まれています。

[エージェントインストールとエージェント設定の比較 \(P. 27\)](#)

[Apache ベース エージェントを Windows にインストールする方法 \(P. 28\)](#)

[Windows で Apache ベース エージェントを設定する方法 \(P. 29\)](#)

エージェントインストールとエージェント設定の比較

インストールと設定の概念は、SiteMinder エージェントの説明に使用された場合に特定の意味を持ちます。

インストールは SiteMinder エージェント ソフトウェアをコンピュータ システムにインストールすることを意味します。たとえば、エージェントをインストールするとディレクトリが作成され、SiteMinder エージェント ソフトウェアおよびその他の設定がコンピュータにコピーされます。

設定はインストールの後に発生し、コンピュータ上の特定の Web サーバに対する SiteMinder エージェント ソフトウェア準備のアクションを意味します。この準備には SiteMinder ポリシー サーバにエージェントを登録し、コンピュータにインストールされる Web サーバ用のランタイム サーバインスタンスを作成することが含まれます。

エージェントを最初の Web サーバにインストールおよび設定するには、ウィザードベースのインストールおよび設定プログラムを使用します。ウィザードベースのプログラムは `.properties` ファイルを作成します。

`.properties` ファイルおよび対応する実行可能ファイルを使用して、エージェントを追加の Web サーバにサイレントにインストールまたは設定します。

Apache ベース エージェントを Windows にインストールする方法

SiteMinder エージェントを Windows オペレーティング環境にインストールするには、いくつかの個別の手順が必要になります。これらの手順は以下のプロセスを使用して説明されます。

1. [インストールプログラムに必要な情報を収集する](#) (P. 28)。
2. [ウィザードベースのインストールプログラムを実行する](#) (P. 28)。

インストールプログラム用の情報を収集します。

エージェントのインストールプログラムを実行する前に、Web サーバに関する以下の情報を収集します。

インストール ディレクトリ

Web サーバ上のエージェント バイナリ ファイルの場所を指定します。
`web_agent_home` 変数はこの場所に設定されます。

制限： この製品では、パス内の最下部のディレクトリに「webagent」という名前を付ける必要があります。

Windows 上でのインストールプログラムの実行

エージェントのインストールプログラムは Windows オペレーティング環境を使用して、一度に 1 台のコンピュータにエージェントをインストールします。このインストールプログラムはウィザードまたはコンソールモードで実行できます。ウィザードおよびコンソールベースのインストールプログラムは、同じ設定の無人またはサイレント メソッドを使用して、後続のインストールおよび設定のための `.properties` ファイルも作成します。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した `.properties` ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. Web サーバ上の一時ディレクトリに Web エージェントインストール実行可能ファイルをコピーします。
2. 以下のいずれかを実行します。
 - ウィザードベースのインストールでは、インストール実行可能ファイルを右クリックし、次に、[管理者として実行] を選択します。
 - コンソールベースのインストールでは、コマンドライン ウィンドウを開き、以下の例に示すとおり実行可能ファイルを実行します。

```
executable_file_name.exe -i console
```
3. 事前に収集した情報を使用してインストールを完了します。

Windows で Apache ベース エージェントを設定する方法

エージェント設定はインストールの後に実行します。設定には、以下のプロセスを使用して説明されるいくつかの個別の手順が必要になります。

1. [設定用の情報を収集する](#) (P. 29)。
2. [ウィザードベースの設定プログラムを実行する](#) (P. 33)。
3. [\(オプション\) 環境内の後続の Web サーバに対して無人またはサイレントインストールおよび設定プログラムを実行する](#) (P. 35)。

Windows 上の設定プログラムが必要とする情報を収集する

エージェントの設定プログラムを実行する前に、製品の環境に関する以下の情報を収集します。

ホストの登録

トラステッドホストを登録するかどうかを示します。この登録は、Web サーバ上のポリシー サーバおよび `SmHost.conf` ファイルでトラステッドホストオブジェクトを作成します。エージェントは、起動時にポリシー サーバへの初期接続を行うためにこの情報を使用します。トラステッドホストとして各エージェントインスタンスを 1 回のみ登録します。

デフォルト: はい

オプション: はい、いいえ

管理者ユーザ名

ポリシー サーバですでに定義されている管理者権限を持った SiteMinder ユーザの名前を指定します。この SiteMinder ユーザアカウントは、トラステッドホストを登録する権限を必要とします。

Admin Password

ポリシー サーバですでに定義されている管理者ユーザ名用にパスワードを指定します。

管理者パスワードの確認

[管理者パスワード] フィールドに入力したパスワードを繰り返します。この値は、ポリシー サーバですでに定義されている管理者ユーザ名用のパスワードを確認します。

トラステッド ホスト オブジェクト名

登録するトラステッドホスト用の任意の名前を指定します。このトラステッドホストオブジェクトがポリシー サーバ上に格納されます。

ホスト設定オブジェクト

ポリシー サーバですでに定義されているホスト設定オブジェクトの名前を指定します。エージェントがポリシー サーバ (SmHost.conf ファイル設定を使用して) に最初接続した後、後続の接続は [ホスト設定オブジェクト] からの設定を使用します。

ポリシー サーバ IP アドレス

エージェントが起動時に試みるポリシー サーバのインターネットプロトコルアドレスを指定します。ポリシー サーバがファイアウォールの内側にある場合は、ポート番号も指定します。

単一の仮想 IP アドレス (VIP) によってユーザの環境でポリシー サーバを公開するようにハードウェアロードバランサが設定されている場合は、VIP を入力します。

例： (IPV4) 192.168.1.105

例： (ポート番号がある IPV4) 192.168.1.105:44443

例： (IPV6) 2001:DB8::/32

例： (IPV6) [2001:DB8::/32]:44443

FIPS モード設定

以下のいずれかのアルゴリズムを指定します。

FIPS 互換性/AES 互換性

機密データを暗号化するために以前のバージョンの SiteMinder に存在していたアルゴリズムを使用し、以前のバージョンの SiteMinder と互換性があります。組織が FIPS 準拠のアルゴリズムの使用を必要としない場合は、このオプションを使用します。

FIPS 移行/AES 移行

FIPS 互換性モードから FIPS のみのモードへの移行を許可します。FIPS 移行モードでは、SiteMinder 環境は、FIPS 準拠のアルゴリズムを使用して既存の機密データを再暗号化するときに、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。

FIPS のみ/AES のみ

FIPS 準拠アルゴリズムのみを使用して、SiteMinder 環境で機密データを暗号化します。この設定は、SiteMinder の旧バージョンと相互運用されず、また下位互換性もありません。

デフォルト：FIPS 互換/AES 互換

FIPS は、AES（Advanced Encryption Standard、高度暗号化標準）に適合する暗号モジュールを信用する米国政府のコンピュータセキュリティ標準です。

重要： SiteMinder エージェントおよびポリシー サーバの両方に互換性のある FIPS/AES モード（または互換モードの組み合わせ）を使用します。

名前

ポリシー サーバへの初期接続を行うために Web エージェントが使用する設定が含まれている SmHost.conf ファイルの名前を指定します。

デフォルト：SmHost.conf

場所

SmHost.conf ファイルが格納されているディレクトリを指定します。

デフォルト：web_agent_home¥config

共有秘密キーのロールオーバーの有効化

このチェック ボックスをオンにして、ポリシー サーバが Web エージェントへの通信を暗号化するのに使用する共有秘密キーを変更します。

サーバの選択

この手順では複数の画面が表示されます。最初の画面にはサーバタイプ（Apache）が表示され、次の画面には設定プログラムがコンピュータ上で検索する Web サーバインスタンスを表示します。サーバタイプのチェック ボックスをオンにし、設定するインスタンスを選択します。SiteMinder 保護を解除したいインスタンスのチェック ボックスをオフにします。

Apache 2.4.x インストール場所

Apache ベースのサーバ（バージョン 2.4 以上）用のインストールディレクトリの場所を指定します。

エージェント設定オブジェクト名

ポリシーサーバ上ですでに定義されている、エージェント設定オブジェクト（ACO）の名前を指定します。

デフォルト：AgentObj

高度な認証方式のダイアログ

以前に選択した Web サーバインスタンス用の高度な認証方式を指定します。

Windows 上での Web エージェント設定プログラムの実行

エージェント設定用の情報を収集した後、エージェント設定プログラムを実行します。このプログラムは、コンピュータ上で実行される Web サーバ用のエージェント ランタイム インスタンスを作成します。

この設定プログラムは、選択するオプションに応じてウィザードまたはコンソールベースになります。ウィザードまたはコンソールモード内の設定プログラムを 1 回実行すると、プロパティファイルが作成されます。同じオペレーティング環境を持った他のコンピュータ上で無人設定を今後実行するためにプロパティファイルを使用します。

次の手順に従ってください:

1. Web サーバで以下のディレクトリを開きます。

```
web_agent_home¥install_config_info
```

```
web_agent_home
```

SiteMinder エージェントが Web サーバにインストールされるディレクトリを示します。

デフォルト (Windows32 ビット インストールのみ) : <アクセスパス>¥webagent

デフォルト (Windows64 ビット インストールのみ) : <アクセスパス>¥webagent¥win64

デフォルト (64 ビット システム [Wow64] 上で作動する Windows32 ビット アプリケーション) : C:¥Program Files (x86)¥webagent¥win32

2. 以下のいずれかのメソッドを使用します。
 - GUI ベースの設定については、手順 3 に移動します。
 - コンソールベースの設定については、手順 5 に移動します。
3. 以下の実行可能ファイルを右クリックし、次に、[管理者として実行]を選択します。

```
ca-wa-config.exe
```

4. 手順 8 に進みます。
5. 管理者権限を使用してコマンドプロンプト ウィンドウを開きます。
6. 以前にリスト表示された実行可能ファイルに移動し、以下のスイッチで実行します。

```
-i console
```

7. 手順 8 に進みます。
8. 設定プログラムで示されるプロンプトに従います。エージェント設定ワークシートからリクエストされた値を提供します。

エージェントランタイムインスタンスが Web サーバに対して作成されます。

Windows 上で後続のエージェントに対して無人またはサイレント インストールおよび設定プログラムを実行する

無人またはサイレント インストール オプションは、インストールおよび設定プロセスを自動化するのに役立ちます。同一の設定を持つ多くのエージェントを使用する大型の SiteMinder 環境がある場合、このメソッドによって時間を節約できます。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した .properties ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. 最初の Web サーバ上で以下のウィザードを実行します (表示されるおりの順序)。
 - a. SiteMinderWeb エージェント インストール ウィザード。
 - b. SiteMinderWeb エージェント設定ウィザード。
2. 最初の Web サーバで以下のファイルを見つけます。

```
web_agent_home¥install_config_info¥ca-wa-installer.properties
```

注: パスにスペースが含まれる場合は、引用符で囲んでください。

```
web_agent_home
```

SiteMinder エージェントが Web サーバにインストールされるディレクトリを示します。

デフォルト (Windows32 ビット インストールのみ) : <アクセスパス>¥webagent

デフォルト (Windows64 ビット インストールのみ) : <アクセスパス>¥webagent¥win64

デフォルト (64 ビット システム [Wow64] 上で作動する Windows32 ビット アプリケーション) : C:¥Program Files (x86)¥webagent¥win32

3. 環境内の他の Web サーバ上で以下の各手順を実行します。

注: このプロセスを自動化するには、カスタマイズした独自のスクリプトを作成し、システム上でこれらのファイルを実行します。使用するスクリプト言語は任意です。

- a. 後続の Web サーバ上で一時ディレクトリを作成します。
- b. 最初の Web サーバ（手順 1 および 2）から後続の Web サーバ上の一時ディレクトリに以下のファイルをコピーします。

- SiteMinderWeb エージェント インストール実行可能ファイル。
- SiteMinderca-wa-installer properties ファイル。

- c. 一時ディレクトリ内で管理者権限を使用してコマンドプロンプトウィンドウを開きます。

- d. 以下のコマンドを実行します。

```
agent_executable -f properties_file -i silent
```

後続のサーバに SiteMinder エージェントがサイレントにインストールおよび設定されます。

- e. （オプション）後続の Web サーバから一時ディレクトリを削除します。

4. `ca-wa-installer.properties` ファイルの設定で指定される構成を使用する SiteMinder 環境内の各追加の Web サーバに対して手順 3 を繰り返します。

第 3 章: Apache ベースのエージェントの UNIX/Linux へのインストールおよび設定

このセクションには、以下のトピックが含まれています。

[エージェントインストールとエージェント設定の比較 \(P. 37\)](#)

[Apache ベース エージェントを UNIX または Linux にインストールする方法 \(P. 38\)](#)

[z/OS システム上のエージェントをインストールする方法 \(P. 40\)](#)

[UNIX または Linux で Apache ベース エージェントを設定する方法 \(P. 42\)](#)

[z/OS システム上のエージェントを設定する方法 \(P. 53\)](#)

[UNIX/Linux でのオプションのエージェント設定 \(P. 60\)](#)

エージェント インストールとエージェント設定の比較

インストールと設定の概念は、**SiteMinder** エージェントの説明に使用された場合に特定の意味を持ちます。

インストールは **SiteMinder** エージェント ソフトウェアをコンピュータ システムにインストールすることを意味します。たとえば、エージェントをインストールするとディレクトリが作成され、**SiteMinder** エージェント ソフトウェアおよびその他の設定がコンピュータにコピーされます。

設定はインストールの後に発生し、コンピュータ上の特定の **Web** サーバに対する **SiteMinder** エージェント ソフトウェア準備のアクションを意味します。この準備には **SiteMinder** ポリシー サーバにエージェントを登録し、コンピュータにインストールされる **Web** サーバ用のランタイム サーバインスタンスを作成することが含まれます。

エージェントを最初の **Web** サーバにインストールおよび設定するには、ウィザードベースのインストールおよび設定プログラムを使用します。ウィザードベースのプログラムは `.properties` ファイルを作成します。

`.properties` ファイルおよび対応する実行可能ファイルを使用して、エージェントを追加の **Web** サーバにサイレントにインストールまたは設定します。

Apache ベース エージェントを UNIX または Linux にインストールする方法

SiteMinder エージェントの UNIX または Linux オペレーティング環境へのインストールには、いくつかの個別の手順が必要になります。これらの手順は以下のプロセスを使用して説明されます。

1. [インストールプログラムが必要とする情報を収集する \(P. 38\)](#)。
2. [ウィザードベースのインストールプログラムを実行する \(P. 39\)](#)。

インストール用の情報の収集

エージェントのインストールプログラムを実行する前に、Web サーバに関する以下の情報を収集します。

インストール ディレクトリ

Web サーバ上のエージェント バイナリ ファイルの場所を指定します。
`web_agent_home` 変数はこの場所に設定されます。

制限： この製品では、パス内の最下部のディレクトリに「webagent」という名前を付ける必要があります。

UNIX/Linux 上でのインストールプログラムの実行

SiteMinder エージェント用のインストールプログラムは、UNIX または Linux オペレーティング環境を使用して、エージェントを一度に 1 台のコンピュータにインストールします。このインストールプログラムはウィザードまたはコンソールモードで実行できます。ウィザードおよびコンソールベースのインストールプログラムは、同じ設定の無人またはサイレントメソッドを使用して、後続のインストールおよび設定に対する `.properties` ファイルも作成します。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した `.properties` ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

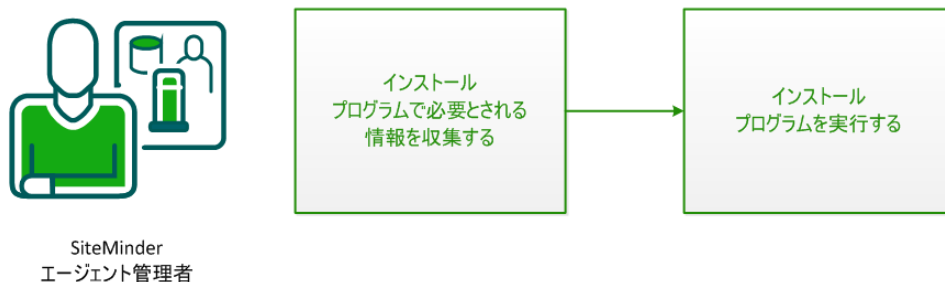
次の手順に従ってください:

1. Web サーバ上の一時ディレクトリに SiteMinder エージェントインストール実行可能ファイルをコピーします。
2. root ユーザとしてログインします。
3. 以下のいずれかを実行します。
 - ウィザードベースのインストールについては、インストール実行可能ファイルを実行します。
 - コンソールベースのインストールでは、コマンドラインウィンドウを開いて実行可能ファイルを以下の例で示されるとおりに実行します。

```
executable_file_name.exe -i console
```
4. インストールプログラムを完了するためにエージェントインストールワークシートからの情報を使用します。

z/OS システム上のエージェントをインストールする方法

z/OS オペレーティング環境に SiteMinder エージェントをインストールするには、以下のプロセスを実行します。



1. [インストールプログラムが必要とする情報を収集する](#) (P. 38)。
2. [インストールプログラムを実行する](#) (P. 40)。

インストール用の情報の収集

エージェントのインストールプログラムを実行する前に、Web サーバに関する以下の情報を収集します。

インストール ディレクトリ

Web サーバ上のエージェント バイナリ ファイルの場所を指定します。
`web_agent_home` 変数はこの場所に設定されます。

制限： この製品では、パス内の最下部のディレクトリに「webagent」という名前を付ける必要があります。

z/OS で SiteMinder エージェント インストール プログラムの実行

SiteMinder エージェント用のインストールプログラムは、z/OS オペレーティング環境を実行する単一のコンピュータにエージェントをインストールします。このインストールプログラムはウィザードまたはコンソールモードで実行できます。ウィザードおよびコンソールベースのインストールプログラムは、同じ設定の無人またはサイレントメソッドを使用して、後続のインストールおよび設定に対する `.properties` ファイルも作成します。

たとえば、環境内のエージェントが同じ **Web** サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した **.properties** ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

テクニカルサポートサイトで、インストールメディアを使用して **SiteMinder** エージェントをインストールします。

注: 実行権限があることを確認してください。インストールメディアへ実行権限を追加するには、以下のコマンドを実行します。

```
chmod +x installation_media
```

```
installation_media
```

SiteMinder エージェント インストーラ実行可能ファイルを指定します。

次の手順に従ってください:

1. root ユーザとしてログインします。
2. 実行中のすべてのアプリケーションを終了します。
3. シェルを開き、インストールメディアに移動します。
4. 以下のいずれかのコマンドを入力して、GUI またはコンソールモード内の設定プログラムを実行します。

GUI モード:

```
java -jar installation_media
```

コンソールモード:

```
java -jar installation_media -i console
```

5. 事前に収集した情報を使用してインストールプログラムを完了します。

UNIX または Linux で Apache ベース エージェントを設定する方法

エージェント設定はインストールの後に実行します。設定には、以下のプロセスを使用して説明されるいくつかの個別の手順が必要になります。

1. 設定プログラムが必要とする情報を収集する。
2. (RedHat Linux のみ) [RedHat Linux 上に埋め込まれた Apache Web サーバ用の設定ファイルを編集します](#) (P. 46)。
3. [エージェント環境スクリプトを用意する](#) (P. 46)。
4. [UNIX または Linux システム上でライブラリ パス変数を設定する](#) (P. 47)。
5. [エージェント設定プログラムを実行する](#) (P. 48)。
6. (オプション) [他のエージェントに対して無人またはサイレントインストールおよび設定プログラムを実行する](#) (P. 50)。
7. [LD PRELOAD 変数を設定する](#) (P. 51)。
8. (SuSE 9 zSeries のみ) [LD ASSUME KERNEL 変数を設定する](#) (P. 52)。
9. (Red Hat Linux のみ) [CAPKIHOME 変数を設定する](#) (P. 52)。

設定プログラムが UNIX/Linux 上で必要とする情報の収集

エージェントの設定プログラムを実行する前に、製品の環境に関する以下の情報を収集します。

ホストの登録

トラステッドホストを登録するかどうかを示します。この登録は、Web サーバ上のポリシー サーバおよび SmHost.conf ファイルでトラステッドホストオブジェクトを作成します。エージェントは、起動時にポリシー サーバへの初期接続を行うためにこの情報を使用します。トラステッドホストとして各エージェントインスタンスを 1 回のみ登録します。

デフォルト： はい

オプション： Yes、No

管理者ユーザ名

ポリシー サーバですでに定義されている管理者権限を持った SiteMinder ユーザの名前を指定します。この SiteMinder ユーザアカウントは、トラステッドホストを登録する権限を必要とします。

管理者パスワード

ポリシー サーバですでに定義されている管理者ユーザ名用にパスワードを指定します。

管理者パスワードの確認

[管理者パスワード] フィールドに入力したパスワードを繰り返します。この値は、ポリシー サーバですでに定義されている管理者ユーザ名用のパスワードを確認します。

トラステッド ホスト オブジェクト名

登録するトラステッドホスト用の任意の名前を指定します。このトラステッドホストオブジェクトがポリシーサーバ上に格納されます。

ホスト設定オブジェクト

ポリシーサーバですでに定義されているホスト設定オブジェクトの名前を指定します。エージェントがポリシーサーバ (SmHost.conf ファイル設定を使用して) に最初接続した後、後続の接続は [ホスト設定オブジェクト] からの設定を使用します。

ポリシー サーバ IP アドレス

エージェントが起動時に試みるポリシーサーバのインターネットプロトコルアドレスを指定します。ポリシーサーバがファイアウォールの内側にある場合は、ポート番号も指定します。

単一の仮想 IP アドレス (VIP) によってユーザの環境でポリシーサーバを公開するようにハードウェアロードバランサが設定されている場合は、VIP を入力します。

例 : (IPV4) 192.168.1.105

例 : (ポート番号がある IPV4) 192.168.1.105:44443

例 : (IPV6) 2001:DB8::/32

例 : (IPV6) [2001:DB8::/32]:44443

FIPS モード設定

以下のいずれかのアルゴリズムを指定します。

FIPS 互換性/AES 互換性

機密データを暗号化するために以前のバージョンの SiteMinder に存在していたアルゴリズムを使用し、以前のバージョンの SiteMinder と互換性があります。組織が FIPS 準拠のアルゴリズムの使用を必要としない場合は、このオプションを使用します。

FIPS 移行/AES 移行

FIPS 互換性モードから FIPS のみのモードへの移行を許可します。FIPS 移行モードでは、SiteMinder 環境は、FIPS 準拠のアルゴリズムを使用して既存の機密データを再暗号化するときに、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。

FIPS のみ/AES のみ

FIPS 準拠アルゴリズムのみを使用して、SiteMinder 環境で機密データを暗号化します。この設定は、SiteMinder の旧バージョンと相互運用されず、また下位互換性もありません。

デフォルト: FIPS 互換/AES 互換

FIPS は、AES (Advanced Encryption Standard、高度暗号化標準) に適合する暗号モジュールを信用する米国政府のコンピュータ セキュリティ標準です。

重要: SiteMinder エージェントおよびポリシー サーバの両方に互換性のある FIPS/AES モード (または互換モードの組み合わせ) を使用します。

名前

ポリシー サーバへの初期接続を行うために Web エージェントが使用する設定が含まれている SmHost.conf ファイルの名前を指定します。

デフォルト : SmHost.conf

場所

SmHost.conf ファイルが格納されているディレクトリを指定します。

デフォルト : `web_agent_home¥config`

共有秘密キーのロールオーバーの有効化

このチェック ボックスをオンにして、ポリシー サーバが Web エージェントへの通信を暗号化するために使用する共有秘密キーを変更します。

サーバの選択

設定プログラムがコンピュータ上で検索する Web サーバインスタンスを示します。設定したいインスタンスのチェック ボックスをオンにします。SiteMinder 保護を解除したいインスタンスのチェック ボックスをオフにします。

エージェント設定オブジェクト名

ポリシー サーバ上ですでに定義されている、エージェント設定オブジェクト (ACO) の名前を指定します。

デフォルト : AgentObj

高度な認証方式のダイアログ

以前に選択した Web サーバ インスタンス用の高度な認証方式を指定します。

RedHat Linux 上の埋め込まれた Apache Web サーバ用の設定ファイルの編集

RedHat Linux システムに（デフォルトで含まれる）埋め込まれた Apache Web サーバの場合、まず特定の設定ファイルを変更して製品に対応します。

次の手順に従ってください：

1. ターゲット RedHat Linux システムにログオンします。
2. テキスト エディタで以下のファイルを開きます。

```
web_agent_home/ca_wa_env.sh
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト（UNIX/Linux インストール）： /opt/ca/webagent

3. LD_PRELOAD 変数を設定する行がコメントで始まっていることを確認します（LD_PRELOAD 変数の設定は無効です）。
4. 変更を保存して、ca_wa_env.sh ファイルを閉じます。
5. テキスト エディタで以下のファイルを開きます。

```
/etc/sysconfig/httpd
```

6. 以下の行をファイルの終わりに追加します。

```
PATH=$PATH:web_agent_home/bin
```

7. 変更を保存し、テキスト エディタを閉じます。

UNIX または Linux でのエージェント環境スクリプトの取得

エージェントのインストールプログラムは以下のディレクトリで環境スクリプトを作成します。

```
web_agent_home/ca_wa_env.sh
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト（UNIX/Linux インストール）： /opt/ca/webagent

ほとんどの Apache ベースの Web サーバでは、以下のいずれのタスクを行った後にこのスクリプトを取得します。

- エージェント設定プログラムの実行。
- Web サーバの起動。

注: 同じシェル内で以前のタスクをすべて実行する場合は、スクリプトを **1 回**のみ取得します。

RedHat Linux に含まれる埋め込み Apache Web サーバの場合は、以下のいずれかのタスクを実行します。

- httpd サービスを開始する前にスクリプトを取得します。
- 以下のファイル内のスクリプトを取得します（毎回手動で開始する場合を除きます）。

/etc/init.d/httpd

UNIX または Linux でのライブラリパス変数の設定

エージェント設定プログラムを実行する前に UNIX または Linux システム上でライブラリパス変数を設定します。

以下の表は、さまざまな UNIX および Linux オペレーティング環境用のライブラリパス変数のリストを示しています。

| オペレーティング システム | ライブラリパス変数の名前 |
|---------------|-----------------|
| AIX | LIBPATH |
| Linux | LD_LIBRARY_PATH |
| Solaris | LD_LIBRARY_PATH |

ライブラリパス変数の値を `web_agent_home/bin` ディレクトリに設定します。

`web_agent_home`

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : `/opt/ca/webagent`

apachectl スクリプトを使用する場合の Web エージェント変数の設定

(POSIX 上で Apache Web サーバを実行する場合などに) apachectl スクリプトを使用して、Apache サーバを実行します。apachectl スクリプトに行を追加すると、エージェントの環境変数が設定されます。

次の手順に従ってください:

1. 以下の例のような 1 行を探します。

```
# Source /etc/sysconfig/httpd for $HTTPD setting, etc
```

2. この行の後に、以下の行を追加します。

```
sh /web_agent_home/ca_wa_env.sh
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : /opt/ca/webagent

UNIX/Linux 上での Web エージェント設定プログラムの実行

エージェント設定用の情報を収集した後、エージェント設定プログラムを実行します。このプログラムは、コンピュータ上で実行される Web サーバ用のエージェント ランタイム インスタンスを作成します。

この設定プログラムは、選択するオプションに応じてウィザードまたはコンソールベースになります。ウィザードまたはコンソールモード内の設定プログラムを 1 回実行すると、プロパティファイルが作成されます。同じオペレーティング環境を持った他のコンピュータ上で無人設定を今後実行するためにプロパティファイルを使用します。

次の手順に従ってください:

1. Web サーバで以下のディレクトリを開きます。

```
web_agent_home/install_config_info
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : /opt/ca/webagent

2. 以下のいずれかのメソッドを使用します。
 - GUI ベースの設定については、手順 3 に移動します。
 - コンソールベースの設定については、手順 5 に移動します。
3. 以下の実行可能ファイルを実行します。
`ca-wa-config.bin`
4. 手順 8 に進みます。
5. root 権限を使用してコマンドプロンプト ウィンドウを開きます。
6. 以前にリスト表示された実行可能ファイルに移動し、以下のスイッチで実行します。
`-i console`
7. 手順 8 に進みます。
8. 設定プログラムで示されるプロンプトに従います。エージェント設定ワークシートからリクエストされた値を提供します。

エージェントランタイム インスタンスが Web サーバに対して作成されます。

SiteMinder デフォルト ヘッダの参照

SMUSER や SMUSERDN などの SiteMinder デフォルト ヘッダを参照するには、Apache 2.4.x Web サーバで LegacyVariables パラメータを No に設定します。

UNIX/Linux 上でエージェントに対して無人またはサイレント インストールおよび設定プログラムを実行する

無人またはサイレント インストール オプションは、インストールおよび設定プロセスを自動化するのに役立ちます。同一の設定を持つ多くのエージェントを使用する大型の SiteMinder 環境がある場合、このメソッドによって時間を節約できます。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した `.properties` ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. 最初の Web サーバ上で以下のウィザードを実行します (表示されるおりの順序)。
 - a. SiteMinderWeb エージェント インストール ウィザード。
 - b. SiteMinderWeb エージェント設定ウィザード。
2. 最初の Web サーバで以下のファイルを見つけます。

`web_agent_home/install_config_info/ca-wa-installer.properties`

注: パスにスペースが含まれる場合は、引用符で囲んでください。

`web_agent_home`

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : `/opt/ca/webagent`

3. 後続の Web サーバ上で以下の手順をそれぞれ実行します。

注: このプロセスを自動化するには、カスタマイズした独自のスクリプトを作成し、システム上でこれらのファイルを実行します。使用するスクリプト言語は任意です。

- a. 後続の Web サーバ上で一時ディレクトリを作成します。
- b. ウィザードを実行した (手順 1 および 2) Web サーバから後続の Web サーバ上の一時ディレクトリに以下のファイルをコピーします。

- SiteMinderWeb エージェント インストール実行可能ファイル。
 - SiteMinderca-wa-installer properties ファイル。
- c. 一時ディレクトリ内でルート権限を使用してコマンドプロンプトウィンドウを開きます。
 - d. 以下のコマンドを実行します。

```
agent_executable.bin -f properties_file -i silent
```

Web サーバに SiteMinder エージェントがサイレントにインストールおよび設定されます。
 - e. (オプション) Web サーバから一時ディレクトリを削除します。
4. ca-wa-installer.properties ファイルの設定で指定される構成を使用する SiteMinder 環境内の各追加の Web サーバに対して手順 3 を繰り返します。

LD_PRELOAD 変数の設定

ほとんどの Apache ベースの SiteMinder エージェントは、LD_PRELOAD 変数が以下の値に設定されることを必要とします。

```
LD_PRELOAD=web_agent_home/bin/libbtunicode.so
```

注: RedHat Linux に含まれる埋め込みの Apache Web サーバは、[別の設定手順](#) (P. 46)を必要とします。

SuSE Linux 9 for zSeries 上の Apache エージェントに LD_ASSUME_KERNEL を設定する

SuSE Linux 9 for zSeries 上で実行する Apache Web サーバに Web エージェントをインストールした後で、LD_ASSUME_KERNEL 環境変数を以下のように設定します。

```
LD_ASSUME_KERNEL=2.4.21
export LD_ASSUME_KERNEL
```

重要: この変数は、Web エージェントライブラリが構築されるカーネルリリースを表すため、**2.4.21** に設定する必要があります。

この設定を行わないと、以下の問題が発生します。

- Apache Web サーバが適切に起動しません。
- ホスト登録はコアをダンプします。

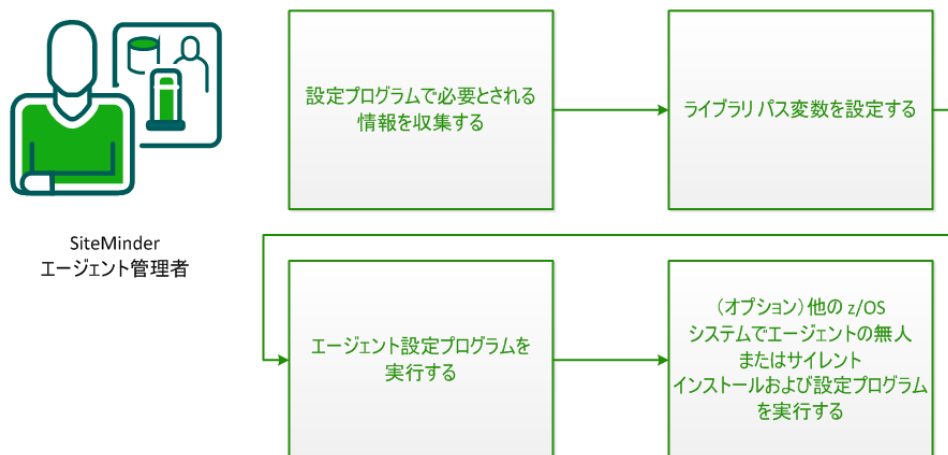
Red Hat Linux システム用の CAPKIHOME 変数の設定

Red Hat Linux システム上で Apache ベースの Web エージェントを実行する場合は、以下のコマンドの入力により CAPKIHOME 環境変数を設定します。

```
CAPKIHOME="/usr/local/CA/webagent/CAPKI"
export CAPKIHOME
```

z/OS システム上のエージェントを設定する方法

インストールの後に以下のプロセスで説明する手順を実行して SiteMinder エージェントを設定します。



1. [設定プログラムが必要とする情報を収集する](#) (P. 53)。
2. [ライブラリパス変数を設定する](#) (P. 56)。
3. [エージェント設定プログラムを実行する](#) (P. 57)。
4. [\(オプション\) 無人またはサイレントインストールを実行し、他の z/OS システムでのエージェント用の設定プログラムを実行する](#) (P. 58)。

設定プログラムが z/OS 上で必要とする情報を収集する

エージェントの設定プログラムを実行する前に、製品の環境に関する以下の情報を収集します。

ホストの登録

トラステッドホストを登録するかどうかを示します。この登録は、Web サーバ上のポリシーサーバおよび SmHost.conf ファイルでトラステッドホストオブジェクトを作成します。エージェントは、開始時の SiteMinder ポリシーサーバへの初期接続をするためにこの情報を使用します。トラステッドホストとして各エージェントインスタンスを 1 回のみ登録します。

デフォルト：はい

オプション：Yes、No

管理者ユーザ名

ポリシー サーバですでに定義されている管理者権限を持った SiteMinder ユーザの名前を指定します。この SiteMinder ユーザアカウントは、トラステッドホストを登録する権限を必要とします。

管理者パスワード

ポリシー サーバですでに定義されている管理者ユーザ名用にパスワードを指定します。

管理者パスワードの確認

[管理者パスワード] フィールドに入力したパスワードを繰り返します。この値は、ポリシー サーバですでに定義されている管理者ユーザ名用のパスワードを確認します。

トラステッド ホスト オブジェクト名

登録するトラステッドホスト用の任意の名前を指定します。このトラステッドホストオブジェクトがポリシー サーバ上に格納されます。

ホスト設定オブジェクト

ポリシー サーバですでに定義されているホスト設定オブジェクトの名前を指定します。エージェントがポリシー サーバ (SmHost.conf ファイル設定を使用して) に最初接続した後、後続の接続は [ホスト設定オブジェクト] からの設定を使用します。

ポリシー サーバ IP アドレス

エージェントが起動時に試みるポリシー サーバのインターネットプロトコルアドレスを指定します。ポリシー サーバがファイアウォールの内側にある場合は、ポート番号も指定します。

単一の仮想 IP アドレス (VIP) によってユーザの環境でポリシー サーバを公開するようにハードウェア ロードバランサが設定されている場合は、VIP を入力します。

例： (IPV4) 192.168.1.105

例： (ポート番号がある IPV4) 192.168.1.105:44443

例： (IPV6) 2001:DB8::/32

例： (IPV6) [2001:DB8::/32]:44443

FIPS モード設定

以下のいずれかのアルゴリズムを指定します。

FIPS 互換性/AES 互換性

機密データを暗号化するために以前のバージョンの SiteMinder に存在していたアルゴリズムを使用し、以前のバージョンの SiteMinder と互換性があります。組織が FIPS 準拠のアルゴリズムの使用を必要としない場合は、このオプションを使用します。

FIPS 移行/AES 移行

FIPS 互換性モードから FIPS のみのモードへの移行を許可します。FIPS 移行モードでは、SiteMinder 環境は、FIPS 準拠のアルゴリズムを使用して既存の機密データを再暗号化するときに、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。

FIPS のみ/AES のみ

FIPS 準拠アルゴリズムのみを使用して、SiteMinder 環境で機密データを暗号化します。この設定は、SiteMinder の旧バージョンと相互運用されず、また下位互換性もありません。

デフォルト: FIPS 互換/AES 互換

注: FIPS は、AES (Advanced Encryption Standard : 高度暗号化標準) に適合する暗号モジュールを信用する米国政府のコンピュータ セキュリティ標準です。

重要: SiteMinder エージェントおよびポリシー サーバの両方に互換性のある FIPS/AES モード (または互換モードの組み合わせ) を使用します。

名前

ポリシー サーバへの初期接続を行うために Web エージェントが使用する設定が含まれている SmHost.conf ファイルの名前を指定します。

デフォルト : SmHost.conf

場所

SmHost.conf ファイルが格納されているディレクトリを指定します。

デフォルト : web_agent_home¥config

共有秘密キーのロールオーバーの有効化

このチェック ボックスをオンにして、SiteMinder ポリシー サーバが Web エージェントへの通信を暗号化するのに使用する共有秘密キーを変更します。

サーバの選択

設定プログラムがコンピュータ上で検索する Web サーバインスタンスを示します。設定したいインスタンスのチェック ボックスをオンにします。SiteMinder 保護を解除したいインスタンスのチェック ボックスをオフにします。

z/OS Path 用 IBM HTTP サーバ

IBM HTTP サーバ httpd.conf 設定ファイルのパスを指定します。

デフォルト： 設定ウィザードを実行するために使用されるアカウントのホーム ディレクトリ。

エージェント設定オブジェクト名

ポリシー サーバ上ですでに定義されている、エージェント設定オブジェクト (ACO) の名前を指定します。

デフォルト： AgentObj

高度な認証方式のダイアログ

以前に選択した Web サーバ インスタンス用の高度な認証方式を指定します。

z/OS 上でライブラリ パス変数を設定する

エージェント設定プログラムを実行する前に z/OS システム上でライブラリ パス変数を設定します。

```
export LIBPATH=web_agent_home/bin
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト： /opt/ca/webagent

z/OS 上で SiteMinder エージェント設定プログラムを実行する

エージェント設定用の情報を収集した後、エージェント設定プログラムを実行します。このプログラムは、コンピュータ上で実行される Web サーバ用のエージェント ランタイム インスタンスを作成します。

この設定プログラムは、選択するオプションに応じてウィザードまたはコンソールベースになります。ウィザードまたはコンソールモード内の設定プログラムを 1 回実行すると、プロパティファイルが作成されます。今後、他の z/OS システム上で無人設定を実行するにはこのプロパティファイルを使用します。

注: 実行権限があることを確認してください。インストールメディアへ実行権限を追加するには、以下のコマンドを実行します。

```
chmod +x installation_media  
installation_media
```

SiteMinder エージェント インストーラ実行可能ファイルを指定します。

次の手順に従ってください:

1. root ユーザとしてログインします。
2. 実行中のすべてのアプリケーションを終了します。
3. シェルを開き、以下のディレクトリに移動します。

```
web_agent_home/install_config_info
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : /opt/ca/webagent

4. 以下のいずれかのコマンドを入力することで、GUI またはコンソールモード内の設定プログラムを実行します。

GUI モード :

```
ca-wa-config.sh
```

コンソール モード :

```
ca-wa-config.sh -i console
```

5. 設定プログラムで示されるプロンプトに従います。エージェント設定ワークシートからリクエストされた値を提供します。

エージェントランタイムインスタンスが Web サーバに対して作成されます。

(オプション) z/OS 上で SiteMinder エージェント用の無人またはサイレントインストールおよび設定プログラムを実行する

無人またはサイレントインストール オプションは、インストールおよび設定プロセスを自動化するのに役立ちます。同一の設定を持つ多くのエージェントを使用する大型の SiteMinder 環境がある場合、このメソッドによって時間を節約できます。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した `.properties` ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. 最初の Web サーバ上で以下のウィザードを実行します (表示されるとおりの順序)。

- a. SiteMinder エージェント インストール ウィザード。
- b. SiteMinder エージェント設定ウィザード。

2. 最初の Web サーバで以下のファイルを見つけます。

`web_agent_home/install_config_info/ca-wa-installer.properties`

注: パスにスペースが含まれる場合は、引用符で囲んでください。

`web_agent_home`

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : `/opt/ca/webagent`

3. 後続の Web サーバ上で以下の手順をそれぞれ実行します。

注: このプロセスを自動化するには、カスタマイズした独自のスクリプトを作成し、システム上でこれらのファイルを実行します。使用するスクリプト言語は任意です。

- a. 後続の Web サーバ上で一時ディレクトリを作成します。
- b. ウィザードを実行した (手順 1 および 2) Web サーバから後続の Web サーバ上の一時ディレクトリに以下のファイルをコピーします。
 - SiteMinder エージェント インストール実行可能ファイル。
 - `ca-wa-installer` プロパティ ファイル。

c. 一時ディレクトリ内でルート権限を使用してコマンドプロンプトウィンドウを開きます。

d. 以下のコマンドを実行します。

```
java -jar installation_media -f ca-installer.properties -i silent  
installation_media
```

SiteMinder エージェント インストーラ実行可能ファイルを指定します。

Web サーバに SiteMinder エージェントがサイレントにインストールおよび設定されます。

e. (オプション) Web サーバから一時ディレクトリを削除します。

4. `ca-wa-installer.properties` ファイルの設定で指定される構成を使用する SiteMinder 環境内の各追加の Web サーバに対して手順 3 を繰り返します。

UNIX/Linux でのオプションのエージェント設定

Apache ベースのサーバ用の SiteMinder エージェントは以下のオプションの設定を提供します。

- [apachectl スクリプトを使用している場合は、スクリプトでエージェント変数を設定する \(P. 48\)](#)。
- [オプションの httpd.conf ファイル変更でサーバのパフォーマンスを改善する \(P. 61\)](#)。

apachectl スクリプトを使用する場合の Web エージェント変数の設定

(POSIX 上で Apache Web サーバを実行する場合などに) apachectl スクリプトを使用して、Apache サーバを実行します。apachectl スクリプトに行を追加すると、エージェントの環境変数が設定されます。

次の手順に従ってください:

1. 以下の例のような 1 行を探します。

```
# Source /etc/sysconfig/httpd for $HTTPD setting, etc
```

2. この行の後に、以下の行を追加します。

```
sh /web_agent_home/ca_wa_env.sh
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : /opt/ca/webagent

オプションの httpd.conf ファイル変更でサーバのパフォーマンスを改善する

httpd.conf ファイルのデフォルト構成設定を変更することで、サーバのパフォーマンスを改善できますが、これらの変更は必須ではありません。

次の手順に従ってください:

1. Apache ベースのサーバについては、Web サーバ上の他の認証モジュールまたはアクセスモジュールよりも高い優先度レベルを Apache20WebAgent.dll ファイルに割り当てます。
2. トラフィックが少ない Web サイトについては、以下のディレクティブを定義します。
 - Set MaxRequestsPerChild>1000 または Set MaxRequestsPerChild=0
 - MinSpareServers >5
 - MaxSpareServers >10
 - StartServers=MinSpareServers >5

3. トラフィックが多い Web サイトについては、以下のディレクティブを定義します。
 - Set MaxRequestsPerChild>3000 または Set MaxRequestsPerChild=0
 - MinSpareServers >10
 - MaxSpareServers>15
 - StartServers=MinSpareServers>10

注: CA サービスでは、特定の環境でのパフォーマンス チューニングのサポートを提供することができます。

第 4 章: ダイナミックポリシー サーバクラスタ

旧バージョンの SiteMinder エージェントは、ポリシー サーバがクラスタに追加されたり、クラスタから削除されたりした時期を自動的に検出しませんでした。エージェントは、Web サーバが再起動された後にしか変更を認識しませんでした。

SiteMinder12.52 SP1 は動的ポリシー サーバクラスタをサポートします。動的ポリシー サーバクラスタを有効にすると、既存クラスタに対するポリシー サーバの追加または削除がエージェントによって自動検出されません。

たとえば、エージェントが以下のポリシー サーバのクラスタに接続しているとします。

- 192.168.2.100
- 192.168.2.101
- 192.168.2.103
- 192.168.2.104

オペレーティング システムをアップグレードするためにサーバ 192.168.2.103 を削除することを後で決めたとします。この状況では、動的なポリシー サーバクラスタを有効にすると、エージェントはクラスタのメンバシップの変更を再起動なしに認識することができます。

以下のタスクのいずれかを実行する場合は、**Web** サーバを再起動します。

- 既存のポリシー サーバの設定を変更します (設定ウィザードを使用)。
- ポリシー サーバ クラスタの作成
- ポリシー サーバ クラスタの削除
- 以下のいずれかのポリシー サーバ設定値の変更：
 - EnableFailOver
 - MaxSocketsPerPort
 - MinSocketsPerPort
 - NewSocketStep
 - RequestTimeout

ダイナミックポリシー サーバクラスタへの Web エージェントの接続

Web サーバ上の SmHost.conf ファイルを変更することで、1 つ以上のダイナミックポリシーサーバクラスタに Web エージェントを接続することができます。

次の手順に従ってください：

1. テキストエディタで以下のファイルを開きます。

```
web_agent_home¥config¥SmHost.conf
```

```
web_agent_home
```

SiteMinder エージェントが Web サーバにインストールされるディレクトリを示します。

デフォルト (Windows32 ビット インストールのみ) : <アクセスパス>¥webagent

デフォルト (Windows64 ビット インストールのみ) : <アクセスパス>¥webagent¥win64

デフォルト (64 ビット システム [Wow64] 上で作動する Windows32 ビット アプリケーション) : C:¥Program Files (x86)¥webagent¥win32

2. 以下のタスクのいずれかを実行します。

この Web エージェントがポリシーサーバの動的なクラスタに一度も接続されたことがない場合は、以下のテキストを使用して (ファイル内の任意の場所に) 行を作成します。

```
enableDynamicHCO="YES"
```

- この Web エージェントが以前にポリシーサーバのダイナミッククラスタに接続されていた場合は、既存の enableDynamicHCO パラメータの値を「NO」から「YES」に変更します。

1. SmHost.conf ファイルを保存し、テキストエディタを閉じます。
2. Web サーバを再起動します。

Web エージェントはダイナミックポリシーサーバクラスタに接続されました。

第 5 章: Web エージェントの起動と停止

Web エージェントの有効化

エージェントのパラメータを設定して、エージェントが Web サーバ上のリソースを保護できるようにします。

注: SiteMinder ポリシー サーバにポリシーも定義するまでは、リソースは保護されません。

次の手順に従ってください:

1. WebAgent.conf ファイルをテキスト エディタで開きます。

注: 64 ビット オペレーティング環境にインストールされた IIS 用エージェントには、2 つの WebAgent.conf ファイルがあります。1 つのファイルは 32 ビット Windows アプリケーション用です。もう 1 つのファイルは 64 ビット Windows アプリケーション用です。IIS 用エージェントを起動または停止する場合は、*両方*の WebAgent.conf ファイルを変更します。

2. EnableWebAgent パラメータの値を **yes** に変更します。
3. WebAgent.conf ファイルを保存して閉じます。
4. Web サーバ (サーバが実行されるコンピュータではなく Web サーバ自体) を再起動します。

Web エージェントが有効になります。

Web エージェントの無効化

Web エージェントによる Web サーバ上のリソースの保護を停止する場合は、Web エージェントを無効にします。

次の手順に従ってください:

1. WebAgent.conf ファイルをテキスト エディタで開きます。

注: 64 ビット オペレーティング環境にインストールされた IIS 用エージェントには、2 つの WebAgent.conf ファイルがあります。1 つのファイルは 32 ビット Windows アプリケーション用です。もう 1 つのファイルは 64 ビット Windows アプリケーション用です。IIS 用エージェントを起動または停止する場合は、*両方*の WebAgent.conf ファイルを変更します。

2. EnableWebAgent パラメータの値を no に変更します。
3. WebAgent.conf ファイルを保存して閉じます。
4. Web サーバ (サーバが実行されるコンピュータではなく Web サーバ自体) を再起動します。

Web エージェントが無効になります。

apachectl コマンドによるほとんどの Apache ベース エージェントの起動または停止

UNIX または Linux オペレーティング環境で `apachectl` コマンドを使用してほとんどの Apache ベース エージェントを起動または停止するには、まず製品の環境変数を設定する必要があります。

注: Apache ベース エージェントは `apachectl -restart` オプションをサポートしていません。この手順は Apache ベースの IBM HTTP Server には適用されません。代わりに、[この手順](#) (P. 70)を実行します。

次の手順に従ってください:

1. UNIX/Linux オペレーティング環境では、以下のスクリプトを実行することにより環境変数を設定します。

```
./ca_wa_env.sh
```

2. 以下のいずれかのコマンドを使用します。

```
apachectl -stop
```

```
apachectl -start
```

apachectl コマンドによる IBM HTTP Server の起動

apachectl コマンドで IBM HTTP Server 上の Apache ベース エージェントを起動または停止するには、追加の設定手順が必要になります。UNIX または Linux オペレーティング環境で apachectl コマンドを実行する前に、以下のタスクを実行します。

- 環境変数スクリプトを用意します。
- 以下のリストからオペレーティング環境用の適切な変数をエクスポートします。
 - LD_LIBRARY_PATH (32 ビットおよび 64 ビットの動作環境)
 - LD_LIBRARY_PATH_64 (Solaris のみ)

次の手順に従ってください:

1. UNIX/Linux オペレーティング環境では、以下のスクリプトを実行することにより環境変数を設定します。

```
./ca_wa_env.sh
```

2. 以下のタスクのいずれかを実行します。

- 32 ビットまたは 64 ビットのオペレーティング環境では、LD_LIBRARY_PATH 変数を IBM HTTP Web サーバの lib ディレクトリの場所に設定します。以下の例を参考にしてください。

```
export LD_LIBRARY_PATH=/opt/IBM/HTTPServer/lib:$LD_LIBRARY_PATH
```

- (Solaris のみ) 64 ビットのオペレーティング環境では、LD_LIBRARY_PATH_64 変数を IBM HTTP Web サーバの lib ディレクトリの場所に設定します。以下の例を参考にしてください。

```
export LD_LIBRARY_PATH_64=/opt/IBM/HTTPServer/lib:$LD_LIBRARY_PATH_64
```

3. 以下の例のように、apachectl コマンドを実行します。

```
apachectl -start
```

注: Apache ベース エージェントは apachectl -restart オプションをサポートしていません。-stop および -start コマンドを代わりに使用します。

opmnctl コマンドによる Oracle 11.g.x HTTP サーバの起動

RedHat Enterprise Linux で該当

Oracle HTTP サーバの起動または停止には、追加の設定手順が必要です。

次の手順に従ってください:

1. テキストエディタで `opmnctl` スクリプトを開きます。
2. エージェントインストールディレクトリで `LD_LIBRARY_PATH` 変数を更新します。以下の例を参考にしてください。

```
export LD_LIBRARY_PATH=/opt/CA/webagent/bin:$LD_LIBRARY_PATH
```

3. `PATH` 変数へ以下のディレクトリを追加します。

```
CA/webagent/bin
```

4. 変更を保存し、`opmnctl` スクリプトを閉じます。
5. 以下のコマンドを入力して、エージェントを起動します。

```
opmnctl startall
```


第 6 章: Web エージェントを 12.52 SP1 にアップグレードする

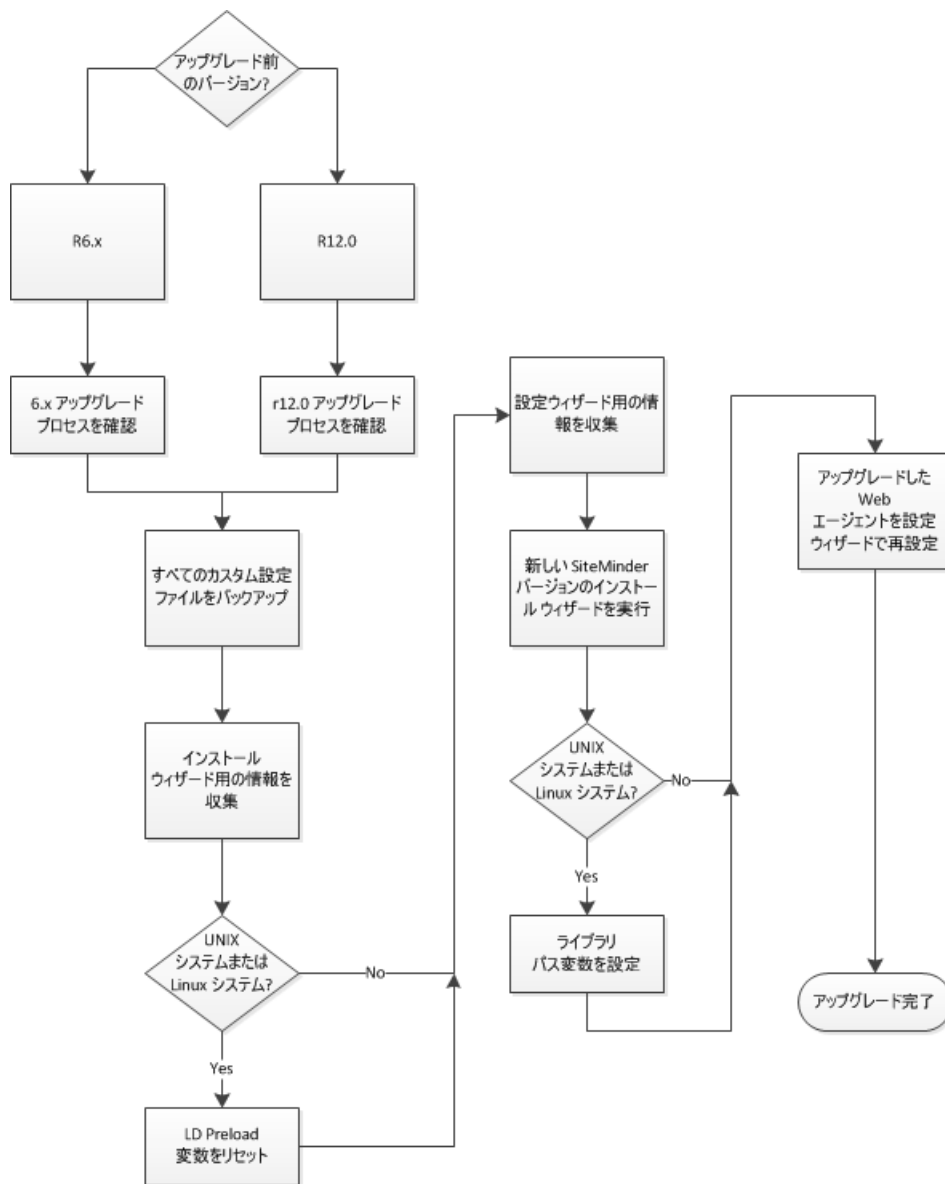
このセクションには、以下のトピックが含まれています。

[エージェントアップグレードロードマップ \(P. 74\)](#)

[SiteMinder エージェントのアップグレードの準備をする方法 \(P. 75\)](#)

エージェントアップグレードロードマップ

以下の図は、SiteMinder エージェントを r12.5 にアップグレードする方法を示しています。



SiteMinder エージェントのアップグレードの準備をする方法

SiteMinder エージェントのアップグレードはいくつかの別個の手順を必要とします。エージェントのアップグレードの準備をするには、次の手順に従ってください：

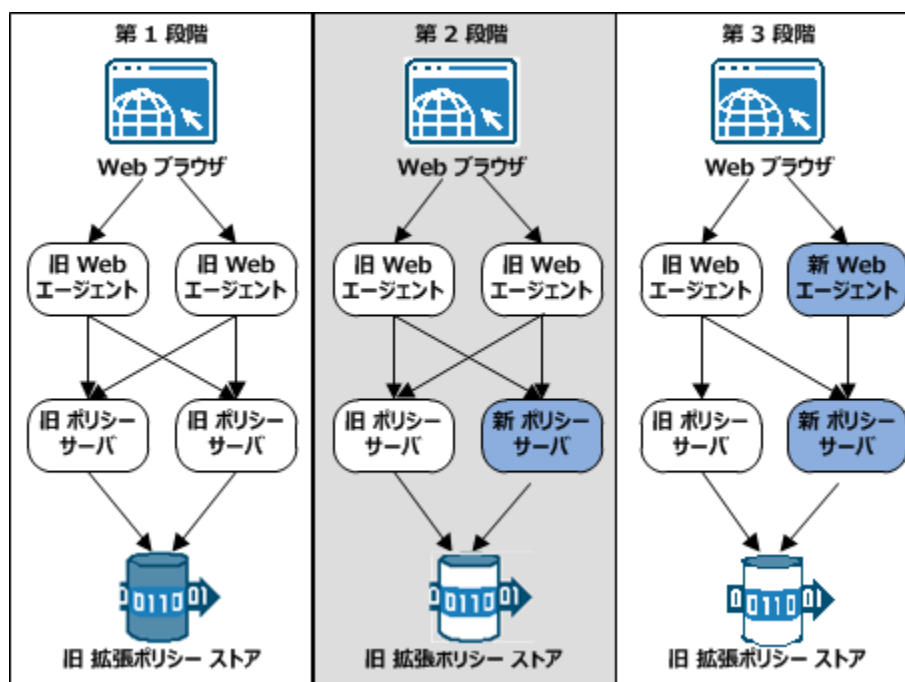
1. 以下の図のいずれかを使用して、エージェントのアップグレードで適切な SiteMinder アップグレード手順を使用していることを確認します。
 - [r6.x からのアップグレード処理](#)
 - [r12.0 からのアップグレード処理 \(P. 78\)](#)
2. Web サーバ上で任意のカスタマイズされた SiteMinder ファイルのバックアップコピーを作成します。エージェントのインストールまたは設定後にカスタマイズできるファイルには以下のファイルが含まれます。
 - LocalConfig.conf
 - WebAgent.conf
3. 以下の SiteMinder プログラムの情報を収集します。
 - エージェントインストール ウィザード。
 - エージェント設定ウィザード。
4. UNIX/Linux オペレーティング環境でエージェントをアップグレードする場合は、[LD PRELOAD 変数をクリア \(P. 79\)](#) します。その他の場合は、次の手順へ進んでください。
5. エージェントをアップグレードするには以下のいずれかの手順を使用します。
 - [Windows オペレーティング環境用のインストール ウィザードを実行する \(P. 80\)](#)。
 - [UNIX/Linux オペレーティング環境用のインストール ウィザードを実行する \(P. 81\)](#)。
6. UNIX/Linux オペレーティング環境でエージェントをアップグレードする場合は、[LD LIBRARYPATH 変数を設定 \(P. 82\)](#) します。その他の場合は、次の手順へ進んでください。
7. アップグレードされたエージェントを設定するために以下の手順のいずれかを実行します。
 - [Windows オペレーティング環境上のアップグレードされたエージェントを設定する \(P. 83\)](#)。

- [UNIX/Linux オペレーティング環境上のアップグレードされたエージェントを設定する \(P. 84\)](#)。

SiteMinder r6.x からのアップグレード処理

以下の例に示されるように、r6.x から 12.52 SP1 へのエージェント アップグレードが SiteMinder アップグレードプロセスの手順 3 で発生します。

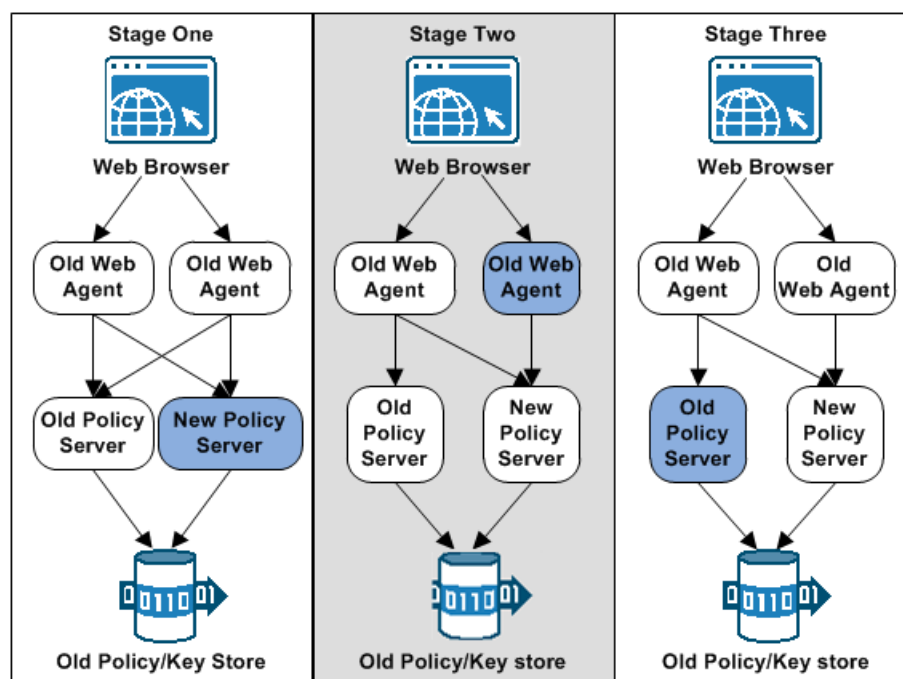
図 1: r6 SP5 移行。第 1 段階～第 3 段階。



SiteMinder r12.0 からのアップグレード処理

以下の例に示されるように、r12.x から 12.52 SP1 へのエージェントアップグレードが SiteMinder アップグレードプロセスの手順 2 で発生します。

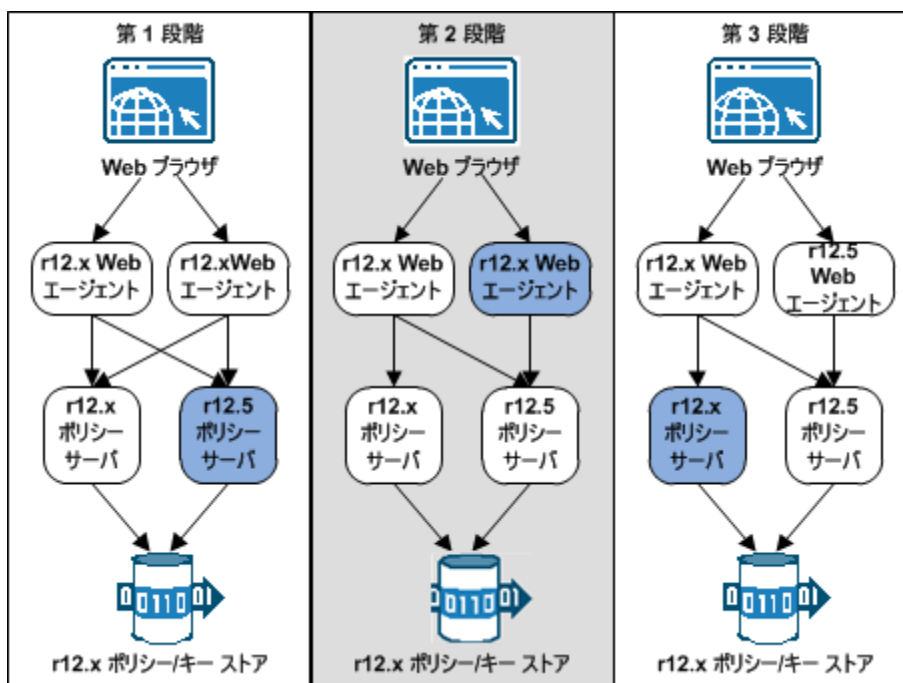
図 2: r12.x 移行の概要。



SiteMinder r12.0 からのアップグレード処理

以下の例に示されるように、r12.0 から 12.52 SP1 へのエージェントアップグレードが SiteMinder アップグレードプロセスの手順 2 で発生します。

図 3: r12.x 移行の概要。



注: SiteMinder アップグレード処理の詳細については、「アップグレードガイド」を参照してください。

LD_PRELOAD 変数が既存のエージェントと矛盾しないことを確認する

シェルから Linux システム上の Web エージェントをアップグレードまたは再インストールする場合は、既存の Web エージェントインストールディレクトリとは別の場所を指すように、LD_PRELOAD 変数を設定します。たとえば、既存の LD_PRELOAD エントリが以下のように設定されているとします。

```
LD_PRELOAD=web_agent_home/bin/libbtunicode.so
```

再インストールまたはアップグレードする前に、以下のように変数を設定します。

```
export LD_PRELOAD=
```

このエントリは変数を空の値に設定します。

UNIX および Linux オペレーティング環境での環境スクリプトの取得

UNIX または Linux で Web エージェントをアップグレードする場合は、以下のディレクトリから使用可能な環境スクリプトを取得します。

SiteMinder 6.0

```
web_agent_home/nete_wa_env.sh
```

SiteMinder 12.0 以降

```
web_agent_home/ca_wa_env.sh
```

web_agent_home

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト : /opt/ca/webagent

Windows 上でエージェントをアップグレードするためにインストール ウィザードを実行する

エージェントのインストールプログラムは **Windows** オペレーティング環境を使用して、一度に 1 台のコンピュータにエージェントをインストールします。このインストールプログラムはウィザードまたはコンソールモードで実行できます。ウィザードおよびコンソールベースのインストールプログラムは、同じ設定の無人またはサイレント メソッドを使用して、後続のインストールおよび設定のための **.properties** ファイルも作成します。

たとえば、環境内のエージェントが同じ **Web** サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した **.properties** ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. **Web** サーバ上の一時ディレクトリに **Web** エージェント インストール実行可能ファイルをコピーします。
2. 以下のいずれかを実行します。
 - ウィザードベースのインストールでは、インストール実行可能ファイルを右クリックし、次に、[管理者として実行] を選択します。
 - コンソールベースのインストールでは、コマンドライン ウィンドウを開き、以下の例に示すとおり実行可能ファイルを実行します。

```
executable_file_name.exe -i console
```
3. 事前に収集した情報を使用してインストールを完了します。

UNIX/Linux 上でエージェントをアップグレードするためにインストール ウィザードを実行する

SiteMinder エージェント用のインストールプログラムは、UNIX または Linux オペレーティング環境を使用して、エージェントを一度に 1 台のコンピュータにインストールします。このインストールプログラムはウィザードまたはコンソールモードで実行できます。ウィザードおよびコンソールベースのインストールプログラムは、同じ設定の無人またはサイレントメソッドを使用して、後続のインストールおよび設定に対する `.properties` ファイルも作成します。

たとえば、環境内のエージェントが同じ Web サーババージョン、インストールディレクトリ、エージェント設定オブジェクト、およびポリシーサーバを使用すると仮定します。最初のインストールにはインストールウィザードまたはコンソールベースのインストールプログラムを使用します。その後、ウィザードまたはコンソールベースのインストールプログラムが作成した `.properties` ファイルを有するインストールプログラムを実行するために独自のスクリプトを作成できます。

次の手順に従ってください:

1. Web サーバ上の一時ディレクトリに SiteMinder エージェント インストール実行可能ファイルをコピーします。
2. root ユーザとしてログインします。
3. 以下のいずれかを実行します。
 - ウィザードベースのインストールについては、インストール実行可能ファイルを実行します。
 - コンソールベースのインストールでは、コマンドライン ウィンドウを開いて実行可能ファイルを以下の例で示されるとおりに実行します。

```
executable_file_name.exe -i console
```
4. インストールプログラムを完了するためにエージェント インストールワークシートからの情報を使用します。

UNIX/Linux 上でアップグレードしたエージェントを設定する前にライブラリパス変数を設定する

エージェント設定プログラムを実行する前に UNIX または Linux システム上でライブラリパス変数を設定します。

以下の表は、さまざまな UNIX および Linux オペレーティング環境用のライブラリパス変数のリストを示しています。

| オペレーティングシステム | ライブラリパス変数の名前 |
|--------------|-----------------|
| AIX | LIBPATH |
| Linux | LD_LIBRARY_PATH |
| Solaris | LD_LIBRARY_PATH |

ライブラリパス変数の値を `web_agent_home/bin` ディレクトリに設定します。

`web_agent_home`

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : `/opt/ca/webagent`

Windows 上でアップグレードしたエージェントを設定する

エージェント設定用の情報を収集した後、エージェント設定プログラムを実行します。このプログラムは、コンピュータ上で実行される Web サーバ用のエージェント ランタイム インスタンスを作成します。

この設定プログラムは、選択するオプションに応じてウィザードまたはコンソールベースになります。ウィザードまたはコンソールモード内の設定プログラムを 1 回実行すると、プロパティファイルが作成されます。同じオペレーティング環境を持った他のコンピュータ上で無人設定を今後実行するためにプロパティファイルを使用します。

次の手順に従ってください:

1. Web サーバで以下のディレクトリを開きます。

```
web_agent_home¥install_config_info
```

```
web_agent_home
```

SiteMinder エージェントが Web サーバにインストールされるディレクトリを示します。

デフォルト (Windows32 ビット インストールのみ) : <アクセスパス>¥webagent

デフォルト (Windows64 ビット インストールのみ) : <アクセスパス>¥webagent¥win64

デフォルト (64 ビット システム [Wow64] 上で作動する Windows32 ビット アプリケーション) : C:¥Program Files (x86)¥webagent¥win32

2. 以下のいずれかのメソッドを使用します。
 - GUI ベースの設定については、手順 3 に移動します。
 - コンソールベースの設定については、手順 5 に移動します。
3. 以下の実行可能ファイルを右クリックし、次に、[管理者として実行]を選択します。

```
ca-wa-config.exe
```

4. 手順 8 に進みます。
5. 管理者権限を使用してコマンドプロンプト ウィンドウを開きます。
6. 以前にリスト表示された実行可能ファイルに移動し、以下のスイッチで実行します。

```
-i console
```

- 手順 8 に進みます。
- 設定プログラムで示されるプロンプトに従います。エージェント設定ワークシートからリクエストされた値を提供します。

エージェントランタイム インスタンスが Web サーバに対して作成されます。

UNIX/Linux 上でアップグレードしたエージェントを設定する

エージェント設定用の情報を収集した後、エージェント設定プログラムを実行します。このプログラムは、コンピュータ上で実行される Web サーバ用のエージェントランタイム インスタンスを作成します。

この設定プログラムは、選択するオプションに応じてウィザードまたはコンソールベースになります。ウィザードまたはコンソールモード内の設定プログラムを 1 回実行すると、プロパティファイルが作成されます。同じオペレーティング環境を持った他のコンピュータ上で無人設定を今後実行するためにプロパティファイルを使用します。

次の手順に従ってください:

- Web サーバで以下のディレクトリを開きます。

```
web_agent_home/install_config_info
```

```
web_agent_home
```

SiteMinder エージェントがインストールされているディレクトリを示します。

デフォルト (UNIX/Linux インストール) : /opt/ca/webagent

- 以下のいずれかのメソッドを使用します。
 - GUI ベースの設定については、手順 3 に移動します。
 - コンソールベースの設定については、手順 5 に移動します。
- 以下の実行可能ファイルを実行します。

```
ca-wa-config.bin
```

- 手順 8 に進みます。
- root 権限を使用してコマンドプロンプト ウィンドウを開きます。

6. 以前にリスト表示された実行可能ファイルに移動し、以下のスイッチで実行します。

`-i console`

7. 手順 8 に進みます。
8. 設定プログラムで示されるプロンプトに従います。エージェント設定ワークシートからリクエストされた値を提供します。

エージェントランタイムインスタンスが Web サーバに対して作成されます。

第 7 章: エージェントに対するオペレーティングシステムの調整

このセクションには、以下のトピックが含まれています。

[共有メモリ セグメントの調整 \(P. 88\)](#)

[Solaris 10 リソース管理を調整する方法 \(P. 91\)](#)

共有メモリ セグメントの調整

Apache ベースのエージェントを Solaris システムにインストールする場合は、エージェントが正しく機能するようにオペレーティング環境の共有メモリ設定を調整します。共有メモリ セグメントまたはオペレーティング環境を増やすことによって、エージェントのパフォーマンスが向上します。共有メモリ セグメントをコントロールする変数はオペレーティング環境の指定ファイルで定義されます。

AIX オペレーティング システムについては、Apache ベースのサーバを開始する前に以下のコマンドを実行する必要があります。

```
export EXTSHM=ON
```

注: Linux オペレーティング環境では共有メモリ セグメントの調整が必要となる場合があります。共有メモリ セグメント、およびそれらを調整する方法の詳細については、特定のオペレーティング環境のドキュメントを参照してください。

次の手順に従ってください:

1. ご使用のオペレーティング環境に該当する手順に従ってください。
 - Solaris: 任意のエディタを使用して、`/etc/system` ファイルを開きます。
2. 以下の方法のうちの 1 つを使用して、共有メモリ変数を変更します。
 - Solaris: 以下のリストに表示された変数を追加し、例に表示された、推奨設定を使用して、それらを設定します。以下の構文を使用します。

```
set shmsys:shminfo_shmmax=33554432
```

```
shmsys:shminfo_shmmax
```

最大の共有メモリ セグメント サイズを指定します。エージェントのリソースおよびセッション キャッシュの最大サイズを制御します。

注: 必要なメモリ セグメントの量を概算するために、各キャッシュに 4 KB/エントリを割り当てるか、または OneView モニタでキャッシュ使用状況の統計を表示します。OneView モニタを使用する方法の詳細については、「Web エージェント設定ガイド」を参照してください。

例：大きなキャッシュ容量を必要とするビジーなサイトには
33554432（32 MB）。

shmsys:shminfo_shmmin

（Solaris では必要なし）最小の共有メモリ セグメントサイズ。
エージェントのリソースおよびセッション キャッシュの最小サイ
ズを制御します。

shmsys:shminfo_shmmni

システム全体で、同時に存在できる共有メモリ セグメントの最大数を指定します。

例：（Solaris 9 以外） N/A

例：（Solaris 9） 200

shmsys:shminfo_shmseg

（Solaris 9 では必要なし）プロセスごとの共有メモリ セグメントの最大数を指定します。

例： 24

semsys:seminfo_semmni

セマフォ識別子の数を指定します。システムで実行するエージェントのすべてのインスタンスに 11 を使用します。

例：（Solaris 9 以外） 100

例：（Solaris 9） 200

semsys:seminfo_semmns

システムのセマフォの数を指定します。システムで実行するエージェントのすべてのインスタンスに 10 を使用します。

例：（Solaris 9） 100

例：（Solaris 9） 400

semsys:seminfo_semmnu

undo 機能を使用して、プロセスの数を指定します。最適なパフォーマンスを得るには、システム内で実行される Apache 子プロセスの数より常に大きくなるように `semmnu` 値を設定します。Apache ベースのサーバに対しては、`maxclients` 設定を 200 以上超える値を使用します。

例：（Solaris 9） 200

3. 変更を保存してファイルまたはユーティリティを終了します。
4. システムを再起動します。
5. コマンドを入力して変更を検証します。

```
$ sysdef -i
```

Solaris 10 リソース管理を調整する方法

エージェントのパフォーマンスを改善するためにプロジェクト レベルでリソース管理を調整します。

注: 詳細については、Solaris のマニュアルを参照してください。

Solaris 10 でリソース管理を調整する場合は、以下のプロセスを使用します。

1. Web エージェントが実行されるユーザ アカウントに関連付けられたプロジェクトを特定します。
2. そのプロジェクトの以下のリソース管理のうちのいずれかの設定を増加します。

`project.max-shm-ids`

プロジェクトの最大共有メモリ ID を指定します。

`project.max-sem-ids`

プロジェクトのセマフォ ID の最大数を指定します。

`project.max-msg-ids`

プロジェクトのメッセージキュー ID の最大数を指定します。

`project.max-shm-memory`

プロジェクトに許可された共有メモリの合計金額を指定します。

`process.max-sem-nsems`

セマフォ セットごとに許可されたセマフォの最大数を指定します。

`process.max-sem-ops`

`semop` ごとに許可されたセマフォ操作の最大数を指定します。

`process.max-msg-messages`

メッセージキューのメッセージの最大数を指定します。

`process.max-msg-qbytes`

メッセージキューのメッセージの最大バイト数を指定します。

第 8 章: Web エージェントのアンインストール

このセクションには、以下のトピックが含まれています。

[Web エージェントのアンインストールに関する注意事項 \(P. 93\)](#)

[Windows オペレーティング環境からの Web エージェントのアンインストール \(P. 94\)](#)

[UNIX システムからの Apache ベースのエージェントのアンインストール \(P. 97\)](#)

[z/OS システムからの Apache ベース エージェントのアンインストール \(P. 98\)](#)

Web エージェントのアンインストールに関する注意事項

以下の点に注意してください。

- インストールされたすべての Web サーバのすべての Web エージェントがアンインストールされます。
- パスワードサービスおよびフォームディレクトリ (`pw_default`、`jpw_default`、`samples_default`) が削除されます。ただし、これらのディレクトリにはカスタマイズされたファイルが含まれている場合があるため、これらのディレクトリのデフォルト以外のコピー (`pw`、`jpw`、サンプル) は削除されません。

Windows オペレーティング環境からの Web エージェントのアンインストール

Windows オペレーティング環境から SiteMinderWeb エージェントをアンインストールする前に、レジストリ設定および Web エージェント構成設定のバックアップ コピーを作成することを考慮してください。

以下の点に注意してください。

- インストールされたすべての Web サーバのすべての Web エージェントがアンインストールされます。
- パスワードサービスおよびフォーム ディレクトリ (`pw_default`、`jpw_default`、`samples_default`) が削除されます。ただし、これらのディレクトリにはカスタマイズされたファイルが含まれている場合があるため、これらのディレクトリのデフォルト以外のコピー (`pw`、`jpw`、サンプル) は削除されません。

次の手順に従ってください:

1. Web サーバを停止します。
2. 以下のいずれかの手順を使用して、サーバ上のエージェントの設定を削除します。
 - ウィザードを使用して Web エージェントを設定解除するには、手順 3 に進みます。
 - コンソールベースのプログラムを使用してエージェントを設定解除するには、手順 6 に進みます。
3. [スタート] - [すべてのプログラム] - [CA] - [CA SiteMinder®] をクリックします。

Web エージェント設定ウィザードへのショートカットが表示されません。

4. ショートカットを右クリックし、[管理者として実行]を選択します。

重要: Windows Server 2008 上でこのウィザードを実行している場合は、実行ファイルを管理者権限で実行します。管理者としてシステムにログインしている場合でも、これらの権限を使用します。詳細については、お使いの CA SiteMinder® コンポーネントの「リリースノート」を参照してください。

Web エージェント設定ウィザードが開始します。

5. Web サーバ上で設定されたエージェント インスタンスのチェックボックスをオフにし、ウィザードを完了します。
6. root 権限を使用してコマンドプロンプト ウィンドウを開きます。
7. ca-wa-config.exe ファイルに移動し、以下のスイッチで実行します。
-i console
8. Web サーバ上で設定されたエージェント インスタンスを設定解除します。設定プログラムが完了するのを待ってから、手順 9 に進みます。
9. 以下のいずれかの手順を選択します。
 - ウィザードを使用して Web エージェントを削除するには、手順 10 に進みます。
 - コンソールベースのプログラムを使用して Web エージェントを削除するには、手順 15 に進みます。
10. [スタート] - [コントロールパネル] - [プログラムと機能] の順にクリックします。
インストールされたプログラムのリストが表示されます。
11. [SiteMinder Web エージェント バージョン番号] をクリックします。
12. [アンインストール/変更] をクリックします。
アンインストール ウィザードが表示されます。
13. [SiteMinder Web エージェントのアンインストール] ダイアログボックスの情報を確認し、次に、[アンインストール] をクリックします。
ウィザードが Web エージェントを削除します。
14. ウィザードが完了するのを待ってから、手順 17 に進みます。
15. コマンドライン ウィンドウを開きます。
16. 以下のディレクトリに移動します。

`web_agent_home`

`web_agent_home`

SiteMinder エージェントが Web サーバにインストールされるディレクトリを示します。

デフォルト (SiteMinder IIS Web エージェントの Windows 32 ビットインストールのみ) : C:¥Program Files¥CA¥webagent

デフォルト (Windows 64 ビットインストール[IIS 用 SiteMinder Web エージェントのみ]) : C:¥Program Files¥CA¥webagent¥win64

デフォルト (64 ビット システムで稼働している Windows 32 ビット アプリケーション [IIS 用 SiteMinder Web エージェントを持つ Wow64 のみ]) : C:\Program Files (x86)\webagent\win32

17. 以下のコマンドを実行します。

```
ca-wa-uninstall.cmd -i console
```

18. アンインストールプログラムが完了するのを待ってから、手順 19 に進みます。

19. Web サーバを開始します。

重要: Web エージェントをアンインストールした後に次の場所から ZeroG レジストリ ファイルを削除します : C:\Program Files\ZeroG Registry\com.zerog.registry.xml

UNIX システムからの Apache ベースのエージェントのアンインストール

これらの手順は GUI およびコンソール モードを削除するためのものです。

注: 64 ビット SUSE Linux 10 システムから Web エージェントを削除するには、追加の準備が必要になります。

以下の点に注意してください。

- インストールされたすべての Web サーバのすべての Web エージェントがアンインストールされます。
- パスワードサービスおよびフォーム ディレクトリ (`pw_default`、`jpw_default`、`samples_default`) が削除されます。ただし、これらのディレクトリにはカスタマイズされたファイルが含まれている場合があるため、これらのディレクトリのデフォルト以外のコピー (`pw`、`jpw`、サンプル) は削除されません。

2 つのモードの手順は同じですが、コンソール モードについては以下に示す例外があります。

- 対応する番号を入力して、必要なオプションを選択します。
- 以下の手順で述べるように、各手順の後で [次へ] をクリックする代わりに **Enter** キーを押して次のプロセスに進みます。

注: アンインストールの前に、バックアップとしてエージェント構成設定をコピーすることをお勧めします。

次の手順に従ってください:

1. Web サーバを停止します。
2. UNIX システムにログインします。
3. 削除するエージェントの設定を削除するために、設定ウィザードを実行します。
4. Web エージェントがインストールされているディレクトリに移動します。

`web_agent_home/install_config_info/ca-wa-uninstall`

5. 必要に応じて、「`chmod +x uninstall`」を入力することにより、アンインストールプログラムの実行権限があることを確認します。
6. コンソール ウィンドウから、以下のコマンドのうちの **1** つを入力します。
 - GUI モード : `./uninstall`
 - コンソール モード : `./uninstall -i console`アンインストールプログラムが起動します。
7. ダイアログ ボックスの情報を読んで、**Web** エージェントの削除を確認し、[アンインストール] をクリックします。**Web** エージェントはシステムから削除されます。
8. [終了] をクリックして、アンインストールプログラムを終了します。
9. (オプション) **Apache** ベースのエージェントについては、設定ウィザードが追加した `httpd.conf` ファイルから行を削除します。
10. ホーム ディレクトリに変更します (現在のディレクトリは削除されています)。
11. **Web** サーバを再起動します。

z/OS システムからの Apache ベース エージェントのアンインストール

これらの手順は **GUI** および **コンソールモード** を削除するためのものです。2 つのモードの手順は同じですが、コンソールモードについては以下に示す例外があります。

- 対応する番号を入力して、必要なオプションを選択します。
- 以下の手順で述べるように、各手順の後で [次へ] をクリックする代わりに **Enter** キーを押して次のプロセスに進みます。

注: アンインストールの前に、バックアップとしてエージェント構成設定をコピーすることをお勧めします。

次の手順に従ってください:

1. root ユーザとしてログインします。
2. Web サーバを停止します。
3. 削除するエージェントの設定を削除するために、設定ウィザードを実行します。
4. シェルを開き、SiteMinder エージェントがインストールされているディレクトリに移動します。

```
web_agent_home/install_config_info/ca-wa-uninstall
```

5. 必要に応じて、アンインストールプログラムでの実行権限を確認するには以下のコマンドを入力します。

```
chmod +x installation_media
```

```
installation_media
```

ポリシー サーバ インストーラの実行ファイルを指定します。

6. 以下のいずれかの手順を実行します。
 - GUI モードを使用してエージェントをインストールした場合は、コンソール ウィンドウから以下のコマンドを入力します。

```
./ca-wa-uninstall.sh
```

アンインストールプログラムが起動します。
 - コンソール モードを使用してエージェントをインストールした場合は、設定ウィザードを使用してエージェントを未設定状態にし、**web_agent_home** ディレクトリを手動で削除します。
7. ダイアログ ボックスの情報を読んで SiteMinder エージェントの削除を確認し、[アンインストール] をクリックします。SiteMinder エージェントはシステムから削除されます。
8. [終了] をクリックして、アンインストールプログラムを終了します。
9. (オプション) 設定ウィザードで追加された行を **httpd.conf** ファイルから削除します。
10. ホーム ディレクトリに変更します (現在のディレクトリは削除されています)。
11. Web サーバを再起動します。

第 9 章: トラブルシューティング

このセクションには、以下のトピックが含まれています。

[開始しない Apache ベースのサーバ \(P. 101\)](#)

[起動時に Apache サーバに shmget エラーが表示される \(P. 102\)](#)

[Apache エージェントは有効になっているがデフォルトのサーバ ページまたは保護されたリソースにアクセスできない \(P. 102\)](#)

[Apache Web エージェントが動作しない \(P. 102\)](#)

[英語以外の入力文字にジャンク文字が含まれる \(P. 103\)](#)

開始しない Apache ベースのサーバ

Windows で有効

問題の状況:

mod_sm24 モジュールをロードした後に、Apache ベースのサーバが開始に失敗しました。

以下のメッセージが Windows イベント ビューアに表示されます。

障害が発生しているアプリケーション名: httpd.exe、バージョン: 2.4.9.0、タイプ スタンプ: 0x5326d7a4

障害が発生しているモジュール名: mod_sm24.dll、バージョン: 12.5.4.914、タイプ スタンプ: 0x5379ded8

例外コード: 0xc00000fd

障害が発生しているアプリケーションのパス: C:\Apache2\bin\httpd.exe

障害が発生しているモジュールのパス: C:\Program Files\CA\Webagent\win64\bin\mod_sm24.dll

解決方法:

以下のタスクを実行します。

1. Visual Studio コマンドライン ウィンドウを開きます。
2. 以下のコマンドを実行します。

```
editbin /STACK:524288 httpd.exe
```

3. スタック予約スペースが 80000 (16 進) に更新されたことを確認します。

起動時に Apache サーバに shmget エラーが表示される

問題の状況:

Web サーバを開始すると、次のようなメッセージが表示されます。
「shmget に失敗しました。

作成中のキャッシュが大きすぎるか、apachectl 再起動を行っている可能性があります。」

解決方法:

共有メモリ セグメントに推奨された調整を行います。

Apache エージェントは有効になっているがデフォルトのサーバページまたは保護されたリソースにアクセスできない

問題の状況:

Web エージェントを有効にしてから、デフォルトの Web サーバページまたは保護されているリソースにアクセスすることができません。

解決方法:

共有メモリ セグメントに推奨された調整を行います。

Apache Web エージェントが動作しない

問題の状況:

Apache Web エージェントが動作していません。

解決方法:

Apache オペレーティング システムの共有メモリを調整します。

英語以外の入力文字にジャンク文字が含まれる

UNIX/Linux 上で有効

問題の状況:

英語以外の入力文字の一部がコンソール ウィンドウに正しく表示されません。 -

解決方法:

コンソール ウィンドウのターミナルの設定を確認します。コンソールが入力文字の高位 (8) ビットをクリアしないことを確認します。以下のコマンドを実行します。

```
stty - istrip
```


付録 A: ワークシート

このセクションには、以下のトピックが含まれています。

[エージェントインストールワークシート \(P. 105\)](#)

[エージェント設定ワークシート \(P. 105\)](#)

エージェント インストール ワークシート

以下の表を使用して、エージェント インストール プログラムが必要とする情報を記録します。

| 必要な情報 | 使用する値 |
|--------------|-------|
| インストールディレクトリ | |

エージェント設定ワークシート

以下の表を使用して、エージェント設定プログラムが必要とする情報を記録します。

| 必要な情報 | 使用する値 |
|--------------------|-------|
| ホスト登録 (はい/いいえ) | |
| 管理者ユーザ名 | |
| 管理者パスワード | |
| 共有秘密キーのロールオーバーの有効化 | |
| トラステッドホスト名 | |
| ホスト設定オブジェクト | |
| IP アドレス | |
| FIPS モード設定 | |
| SmHost.conf ファイル名 | |

| 必要な情報 | 使用する値 |
|---------------------|-------|
| SmHost.conf ファイルの場所 | |
| サーバの選択 | |
| エージェント設定オブジェクト名 | |
| 高度な認証方式のダイアログ | |