

CA SiteMinder[®]

Web エージェント インストール ガイド

r12.0 SP3



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。

CA の事前の書面による承諾を受けずに本ドキュメントの全部または一部を複製、譲渡、開示、変更、複製することはできません。本ドキュメントは、CA が知的財産権を有する機密情報です。ユーザは本ドキュメントを開示したり、(i) 本ドキュメントが関係する CA ソフトウェアの使用について CA とユーザとの間で別途締結される契約または (ii) CA とユーザとの間で別途締結される機密保持契約により許可された目的以外に、本ドキュメントを使用することはできません。

上記にかかわらず、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す。

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。

本ドキュメントの制作者は CA 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 © 2010 CA. All rights reserved. 本書に記載された全ての製品名、サービス名、商号およびロゴは各社のそれぞれの商標またはサービスマークです。

CA 製品リファレンス

このマニュアルが参照している CA の製品は以下のとおりです。

- CA SiteMinder[®]
- CA Identity Manager
- 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。

目次

第 1 章: 準備	13
Web エージェントのインストールを準備する方法.....	14
サポートされているオペレーティング システムと Web サーバ.....	15
Web エージェントインストールのための Windows システムを準備する方法.....	15
Web エージェントインストールのための UNIX システムを準備する方法.....	18
AIX 要件.....	19
Web エージェントインストールのための Linux システムを準備する方法.....	19
Web エージェントインストールのための Domino システムを準備する方法.....	20
その他の Web サーバの準備.....	21
すべての Web エージェントに対する一般的な準備.....	22
エージェントインストールを完了するのに必要な情報の収集.....	22
任意の WebAgentTrace.conf ファイル変更の保存.....	22
Web サーバへの適切なエージェントのインストール.....	23
ポリシー サーバ要件.....	24
すべてのエージェントで必要とされるエージェント設定パラメータ.....	25
Domino Web エージェントで必要とされるエージェント設定パラメータ.....	26
IIS Web エージェントで必要とされるエージェント設定パラメータ.....	27
パスワード サービスの準備.....	28
JSP パスワード サービスの ServletExec の CLASSPATH を修復する (Windows).....	28
パスワード サービスとフォーム ディレクトリ.....	29
第 2 章: Windows システムに Web エージェントをインストールする	31
Windows での GUI モードインストールの実行.....	31
Windows での無人インストール.....	34
Windows での無人インストールの準備.....	35
Windows での無人インストールの実行.....	36
Windows 上で進行中の無人インストールを停止する方法.....	37
Windows での Web エージェントの再インストール.....	37
インストール履歴ログ ファイル.....	38
Windows 上でトラステッド ホストとしてシステムを登録する.....	39
インストールおよび設定ログ ファイル.....	43

SmHost.conf ファイルの変更 (Windows)	44
登録ツールを使用したトラステッド ホストの再登録 (Windows)	46
1 つのシステムに複数のトラステッド ホストを登録する (Windows)	49

第 3 章: UNIX システムに Web エージェントをインストールする 51

UNIX システムへの Web エージェントのドキュメントのインストール	51
UNIX システムへの Web エージェントのインストール	53
UNIX での GUI モードインストールの実行	54
UNIX でのコンソール モードインストールの実行	57
UNIX の無人インストール	58
インストール後の Web エージェント環境変数の設定	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	62
インストール履歴ログ ファイル	62
UNIX への Web エージェントの再インストール	63
UNIX 上でトラステッド ホストとしてシステムを登録する	63
GUI またはコンソール モードでのトラステッド ホストの登録	64
SmHost.conf ファイルの変更 (UNIX)	68
登録ツールを使用したトラステッド ホストの再登録 (UNIX)	70
1 つのシステムに複数のトラステッド ホストを登録する (UNIX)	74

第 4 章: Web エージェントの r12.0 SP3 へのアップグレード 75

Web エージェントのアップグレードの準備をする方法	75
アップグレード手順の確認	76
カスタマイズされたファイルのバックアップ	76
アップグレードの間のパスワード サービスとフォーム テンプレートの変更	76
アップグレードの後に設定ウィザードを実行した場合の結果	76
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	77
既存の読み取り専用ファイルの置換	77
Windows システムでの Web エージェントの r12.0 SP3 へのアップグレード	78
UNIX システムでの Web エージェントの r12.0 SP3 へのアップグレード	80

第 5 章: IIS Web エージェントの設定 83

IIS 7.5 に SiteMinder Web エージェントを設定する方法	83
IIS 7.5 Web Server への役割サービスの追加	84
SiteMinder Web エージェント用のクラシック モード アプリケーション プールの設定	86

SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動.....	87
IIS Web エージェントの設定ウィザードの実行	88
SiteMinder で保護する IIS 7.5 Web サイトへのエージェント ISAPI フィルタの追加.....	89
SiteMinder で保護する IIS 7.5 Web サイトへのハンドラ マッピングの追加	91
アプリケーション プール識別情報に対する SiteMinder SmHost.conf ファイルおよびログ ディレクトリの許可	94
Windows 認証方式用の仮想ディレクトリの作成と設定 (IIS 7.5)	96
IIS 7.0 に SiteMinder Web エージェントを設定する方法.....	97
IIS 7.x Web サーバへの役割サービスの追加	98
SiteMinder Web エージェント用のクラシック モード アプリケーション プールの設定	100
SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動.....	101
IIS Web エージェントの設定ウィザードの実行	102
Windows 認証方式用の仮想ディレクトリの設定 (IIS 7.0)	104
SiteMinder で保護する追加の Web サイトへのハンドラ マッピングの追加	105
SiteMinder で保護する追加の Web サイトにエージェント ISAPI フィルタを追加	107
IIS 6.0 に SiteMinder Web エージェントを設定する方法.....	107
サンプルおよびエラー ファイル ディレクトリへの読み取り許可の割り当て	108
IIS がエージェント ISAPI および CGI 拡張を実行できるようにする	109
同じサーバ上の IIS 6.0 Web エージェントとサードパーティソフトウェア	110
エージェントのアップロード ファイルのサイズ制限の増加	111
IIS Web エージェントの設定ウィザードの実行	112
他のサードパーティフィルタの前にエージェント フィルタおよび拡張を配置する.....	114
Windows 認証方式の仮想ディレクトリの設定 (IIS 6.0)	116
SiteMinder Web エージェントを設定して Microsoft Outlook Web Access を保護する方法	117
SiteMinder ISAPI フィルタがリストの最初に表示されていることを確認します。.....	119
IIS で Outlook 拡張を実行できるようにする.....	120
デフォルトの Web サイトのディレクトリ場所と実行権限の設定	121
Exchange Web サイトへの ISAPI 機能拡張の追加	122
Exchange Web サイトへのディレクトリ セキュリティの設定.....	123
Exchweb Web サイトへの ISAPI 機能拡張の追加	124
Exchweb Web サイトへのディレクトリ セキュリティの設定	125
デフォルトの Web サイトのディレクトリ場所と実行権限の設定	126
SiteMinder が Outlook Web Access Web サイトを保護していることを確認します。.....	126

第 6 章: Oracle iPlanet Web エージェントの設定	127
Windows での設定ウィザードの実行	127
GUI またはコンソール モードを使用する Oracle iPlanet Web エージェントの設定	130
Oracle iPlanet Web サーバの手動設定	133
Oracle iPlanet Web サーバファイルへの変更の適用	135
第 7 章: Apache Web エージェントを設定する	137
Windows システムでの Apache Web エージェントの設定	137
UNIX システムでの Apache Web エージェントの設定方法	140
GUI またはコンソール モードを使用する Apache Web エージェントの設定	141
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	144
Apache エージェント操作に LD_PRELOAD 変数を設定する	144
Domino 6.5.3/SuSe8 Linux システムで X.509 ベースの認証方式を使用するための LD_PRELOAD を設定する	145
SuSE Linux 9 for zSeries 上の Apache エージェントに LD_ASSUME_KERNEL を設定する	145
第 8 章: Domino Web エージェントを設定する	147
Windows システムでの Domino Web エージェントの設定	147
Domino Web エージェント DLL の追加 (Windows)	148
Windows の Domino Web エージェントでの設定ウィザードの実行	149
Windows 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定 (オプション)	151
HTML フォーム認証方式を有効にするための別名設定の設定 (オプション)	151
UNIX システムに Domino Web エージェントを設定する方法	151
Domino Web エージェント DLL の追加 (UNIX)	152
UNIX システムでの Domino Web エージェントの設定方法	153
GUI またはコンソール モードでの Domino Web エージェントの設定	153
UNIX 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定 (オプション)	155
UNIX オペレーティング環境で HTML フォーム認証方式を有効にするための別名設定の設 定 (オプション)	155
第 9 章: すべての Web エージェントに利用可能な設定	157
無人モードで任意の Web エージェントを設定する方法	157
無人設定の準備	158
無人設定の実行	158
共有秘密キーのロールオーバーに対する SmHost.conf ファイル アクセス許可の確認	160

Web エージェントの再設定.....	160
追加のエージェントコンポーネントをセットアップする方法.....	161
ダイナミック ポリシー サーバクラスタ.....	162
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.....	163
第 10 章: Web エージェントの起動と停止	165
Web エージェントの有効化.....	165
Web エージェントの無効化.....	166
第 11 章: オペレーティング システムのチューニング	167
共有メモリ セグメントの調整.....	167
Solaris 10 リソース管理を調整する方法.....	170
第 12 章: パスワード サービス	171
パスワード サービスの実装.....	171
JSP パスワード サービス用の環境をセットアップする方法.....	172
IIS Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方 法.....	173
UNIX オペレーティング環境で Oracle iPlanet Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法.....	174
第 13 章: Web エージェントのアンインストール	177
Web エージェントのアンインストールに関する注意事項.....	177
Web エージェントをアンインストールする前に JRE を PATH 変数に設定する.....	178
Windows システムからの Web エージェントのアンインストール.....	179
Windows システムからのドキュメントのアンインストール.....	180
UNIX システムからの Web エージェントのアンインストール.....	181
UNIX システムからのドキュメントのアンインストール.....	183
第 14 章: トラブルシューティング	185
エージェントのスタートアップおよびシャットダウンの問題(フレームワーク エージェントのみ).....	185
LLAWP がすでに実行しているかまたはログ メッセージが適切なログ ファイルに書き込まれ ていないためにエージェントが起動しない.....	186
LLAWP でのエージェント スタートアップ/シャットダウンのトラブルシューティング.....	187

Web エージェントの開始とシャットダウンの問題 (IBM HTTP Server)	188
ホスト設定ファイルに書き込み許可がない	188
接続およびトラステッド ホスト登録の問題	188
トラステッド ホストの登録に失敗	189
トラステッド ホストからポリシー サーバへの接続がない	190
ホストは登録されたが SMHost.conf ファイルが削除されている	190
一般的なインストールの問題	191
同じシステムでの複数のインストール中に 1 つのインストールがハングする	191
インストール エラー ログの場所	191
コントロールパネルの[プログラムの追加と削除]に Web エージェントが表示されない	192
アップグレード中にエラー メッセージが表示される	192
IIS Web エージェント設定時のメタベース エラー	193
その他の問題	193
Netscape ブラウザで PDF が開きません	194
Adobe Acrobat Reader が Windows システムにインストールされない	195
Oracle iPlanet Web エージェントの問題	195
Web サーバは起動するのに Web エージェントが有効にならない	195
Web サーバが開始すると smget エラー メッセージが表示される	196
再設定された Web エージェントが動作しない	196
Sun Java System Web サーバがランタイムで失敗する	197
Apache Web エージェントの問題	197
Apache サーバは起動するが Web エージェントが有効にならない	197
起動時に Apache サーバに shmget エラーが表示される	198
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	198
Apache Web エージェントが動作しない	198
Domino Web エージェントの問題	198
Domino Web エージェントが有効になっていないのに Web サーバが開始する	199
ローカル設定モードの使用時に Domino エージェントが初期化できない	199

付録 A: 無人インストール 201

ca-wa-installer.properties ファイル	201
一般情報の変更	202
トラステッド ホストの登録	202
トラステッド ホスト登録用のポリシー サーバの識別	203
ホスト設定ファイルの指定	203

設定用の Web サーバの選択.....	204
WEB_SERVER_INFO 変数.....	206
Web サーバを再起動するように設定する (Windows のみ)	209
トラステッド ホスト名およびホスト設定オブジェクトの名前付け	209
付録 B: Sun Java System サーバ構成に追加された設定	211
Sun Java System Server 6.0 への追加	211
Windows プラットフォームへの magnus.conf ファイルの追加.....	212
UNIX プラットフォームの magnus.conf ファイルに追加されたコード	212
Windows プラットフォームへの obj.conf ファイルの追加	213
UNIX プラットフォームへの obj.conf ファイルの追加	215
Windows および UNIX プラットフォームへの mime.types ファイルの追加	217
LLAWP でのエージェント スタートアップの確認.....	217
付録 C: Apache Web エージェントを使用する Web サーバへの設定変更	219
UNIX システムに設定されている Web サーバのライブラリ パス.....	219
httpd.conf ファイルへの変更	220
DSO サポート セクションに追加されたエントリ.....	220
SmlnitFile エントリの追加	221
別名エントリの追加	222
証明書認証エントリの追加	224
付録 D: Web エージェント インストールによって追加または変更された環境変数	225
環境変数の追加または変更	225
索引	227

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。

[Web エージェントのインストールの準備をする方法 \(P. 14\)](#)

[サポートされているオペレーティング システムと Web サーバ \(P. 15\)](#)

[すべての Web エージェントに対する一般的な準備 \(P. 22\)](#)

[ポリシー サーバ要件 \(P. 24\)](#)

[パスワード サービスの準備 \(P. 28\)](#)

Web エージェントのインストールを準備する方法

Web エージェントのインストールを準備するには、以下の手順に従います。

1. 以下のタスクを実行することによって、Web サーバを準備します。
 - a. Web サーバで以下のうちの 1 つにアカウントがあることを確認します。
 - 管理者権限 (Windows システム用)。
 - ルート権限 (UNIX システム用)。
 - b. オペレーティング システムに適切なサービス パックまたはパッチがインストールされていることを確認します。
 - c. 使用している Web サーバのタイプで SiteMinder エージェントを操作するのに必要な任意のオプションまたは設定を設定します。これらのいくつかの可能な例を以下に示します (ただし必ずしもこれらに限定されません)。
 - デフォルト以外の [IIS Web](#) (P. 16) サイトを使用する。
 - [Linux System](#) (P. 20) 上で使用するために Apache Web サーバをコンパイルする。
2. すべての Web エージェント インストールについて、以下のアイテムを確認します。
 - ポリシー サーバが [インストールされていて設定されている](#) (P. 24) ことを確認する。
 - [Web エージェントのインストールを完了する](#) (P. 22) ために必要な情報を収集する。
 - [WebAgentTrace.conf](#) (P. 22) ファイルの変更を保存する。
 - [Web サーバ](#) (P. 22) に正しいエージェントを選択する。
3. (オプション) [パスワード サービス](#) (P. 28) の前提条件に適合する。
4. (オプション) 登録サービスの前提条件に適合する。

サポートされているオペレーティング システムと Web サーバ

Web エージェントをインストールする前に、サポートされているオペレーティング システムと Web サーバ設定を使用していることを確認します。SiteMinder Web エージェントおよびサポートされている Web サーバ プラットフォームのリストについては、[テクニカル サポート](#)に移動して、SiteMinder r12.0 SP3 プラットフォーム マトリックスを参照してください。

注: Web エージェントをインストールした後で、システムにインストールされている各 Oracle iPlanet および Apache Web サーバに複数の Web エージェント インスタンスを設定できます。

Web エージェント インストールのための Windows システムを準備する方法

Web エージェント インストールのための Windows システムを準備するには、環境に応じて、以下のタスクの 1 つ以上を実行する必要がある場合があります。

- 64 ビット Windows プラットフォームに Web エージェントをインストールしている場合は、[Visual C++2005 再頒布可能パッケージ](#) (P. 15)を最初にインストールする必要があります。
- [デフォルト以外の IIS Web サイト](#) (P. 16)を準備します。
- [すべてのユーザのためのサービスとして](#) (P. 17) Apache Web サーバをインストールします。

Microsoft Visual C++ 2005 再頒布可能パッケージ(x64)の前提条件

Windows 64 ビットプラットフォームに r12.0 SP3 Web エージェントをインストールする前に、Microsoft Visual C++2005 再頒布可能パッケージ(x64)をダウンロードしてインストールする必要があります。[Microsoft ダウンロード ページ](#)に移動し、「Microsoft Visual C++2005 再頒布可能パッケージ(x64)」を検索します。

デフォルト以外の IIS Web サイトを使用する方法

SiteMinder には、適切にインストールするためのデフォルトの IIS Web サイトが必要です。デフォルトでは、IIS Web サーバをインストールすると、このサイトは存在します。以下の条件のうちのいずれかが存在する場合は、SiteMinder IIS Web エージェントを設定する前にメタベースを編集します。

- デフォルトの IIS Web サイトが存在しなくなった場合。
- デフォルトの IIS Web サイトが名前を変更されている場合。
- 別の(デフォルト以外の) IIS Web サイトに SiteMinder 仮想ディレクトリをインストールする場合。

メタベースの編集に関与する実際のツールおよび手順は、使用している IIS のバージョンによって異なります。たとえば、IIS 6.0 で r12.0 SP3 SiteMinder Web エージェントを使用している場合は、以下のプロセスを使用してメタベースを編集します。

注: 詳細については、Microsoft のドキュメントを参照するか、または <http://support.microsoft.com/> に移動します。

1. 以下を実行することにより、Microsoft からメタベース エクスプローラをダウンロードしてインストールします。
 - a. [Microsoft ダウンロード](#)の Web サイトに移動します。
 - b. メタベース エクスプローラが含まれている「IIS 6.0 リソース キット ツール」を検索します。
 - c. ツールをダウンロードしてインストールします。
 - d. metabase.xml ファイルのバックアップ コピーを作成します。
2. IIS Web サーバで、IIS マネージャを開きます。SiteMinder Web エージェントをインストールする Web サイトを検索し、将来の参照用にその識別子(番号)を書き留めます。
3. IIS マネージャを閉じて、メタベース エクスプローラを開きます。
4. 以下のキーを展開します。

LM#W3SVC#
5. 手順 2 からの識別子に対応するキーを展開します。

サブ キーのリストが表示されます。

6. 手順 5 からのキーを右クリックし、[名前の変更]を選択して、キーの値を 1 に変更します。

7. 左ペインのサブ キーのリストから、以下のキーを展開します。

`root`

キーのリストがメタベース エクスプローラの右ペインに表示されます。

8. 以下のキーをダブルクリックします。

`AppRoot`

[AppRoot のプロパティ]ダイアログ ボックスが表示されます。[値のデータ] フィールドに以下の文字列が表示されます。

`/LM/W3SVC/identifier_number/Root`

9. `identifier_number` の値を 1 に変更し、次に、[OK]をクリックします。

10. メタベース エクスプローラを閉じます。

11. 設定ウィザードを実行して IIS Web エージェントを再設定します。

12. 手順 3 から 10 を繰り返しますが、1 の値を手順 2 からの元の識別子に変更します。

13. IIS Web サーバを再起動します。

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

Apache Web サーバが個別のユーザにインストールされている場合は、Web エージェント設定ウィザードは有効な Apache インストールを検出しません。

Apache Web サーバをインストールする場合は、「すべてのユーザに利用可能なサービスとしてインストールする」のオプションを選択することによって、設定中に、SiteMinder Web エージェントがユーザのシステムで既存の Web サーバを検出できるようにします。

Apache Web サーバを「手動での開始、現在のユーザのみ」オプションを使用してインストールすると、Web エージェントをインストールすることはできますが、設定ウィザードが Apache Web サーバを検出できないため、Web エージェントをサーバに設定することができません。

Web エージェント インストールのための UNIX システムを準備する方法

Web エージェント インストールのための UNIX システムを準備するには、以下のプロセスを使用します。

1. DISPLAY 変数を設定します。
2. 以下に示すように、オペレーティング システムに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。
 - 必要とされる [AIX パッチ](#) (P. 19)
 - 必要とされる [Solaris パッチ](#) (P. 18)

UNIX 上で Web 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 Web 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、Web 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。

必要とされる Solaris パッチ

Solaris マシンに Web 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する必要があります。

Solaris 9

パッチ 111711-16 が必要です。

Solaris 10

パッチ 119963-08 が必要です。

パッチ バージョンは、ルート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、「[Sun Microsystems Solution Center](#)」に移動します。

AIX 要件

AIX システム上で実行する SiteMinder Web エージェントでは以下が必要です。

- AIX システム上で再設計された (フレームワーク) SiteMinder Oracle iPlanet Web エージェントまたは Apache Web 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。

注: 詳細については、以下の Web サイトを参照してください。
<http://www-1.ibm.com/support/docview.wss?uid=swg1IY78159>

Web エージェント インストールのための Linux システムを準備する方法

Web エージェント インストールのための Linux システムを準備するには、以下のプロセスを使用します。

1. 適切な Linux パッチがインストールされていることを確認します。
2. 適切な Linux 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. 適切な Linux ツールがインストールされていることを確認します。
4. Apache Web サーバを使用している場合は、コンパイルします。

必要とされる Linux パッチ

以下の Linux パッチが必要です。

Linux Release 2.1 用

Linux Application Server 2.1 用の glibc-2.4.2-32.20

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

このコンポーネントの Linux バージョンをインストールまたはアップグレードしている場合、ホスト システム上で以下が必要になります。

`compat-libstdc++-33.3.2.3-patch_version.i386.rpm`

この rpm をインストールして、お使いのオペレーティング システム用の適切な 32 ビット C ランタイム ライブラリが使用できることを確認してください。

必要とされる Linux ツール

Red Hat Enterprise Linux 動作環境で実行中の Red Hat Apache 2.2 Web サーバに SiteMinder Web 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

Web エージェントが Linux を実行する Apache Web サーバで操作するには、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 `pthread` (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 `pthread` を使用するモジュールを初期化できません。

Web エージェントの Linux に Apache をコンパイルする方法

1. 次のコマンドを入力します。

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

2. 以下を入力することによって、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
```

Web エージェント インストールのための Domino システムを準備する方法

Web エージェント インストールのために Domino システムを準備するには、以下のいずれかのシステムに適切なアイテムがインストールされていることを確認します。

- Domino [6.5.2](#) (P. 21) に必要とされる IBM Hot Fix

Domino 6.5.2 に必要とされる IBM Hot Fix

Web エージェントを Domino 6.5.2 サーバで実行するには、IBM hot fix SPR #NORK632KQA が必要です。

この hotfix は Windows と UNIX プラットフォームに適用されます。

その他の Web サーバの準備

以下のセクションでは、さまざまな Web サーバのインストール準備について説明します。

Apache Web エージェントのログ サブディレクトリの追加

Apache Web エージェントでは、Web エージェントが正常に動作できるように、ログ サブディレクトリは Apache サーバのルート ディレクトリ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すべての Apache ベース サーバに適用されます。

IBM HTTP サーバ ログの書き込み許可の有効化

IBM HTTP Server に Web 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 Web 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。

LD_LIBRARY_PATH 変数を IBM HTTP Server 7.0 用に設定

Web エージェント設定ウィザードを実行して IBM HTTP Server 7.0 を設定する前に、LD_LIBRARY_PATH 変数を以下の例のように設定します。

```
LD_LIBRARY_PATH=home_directory_of_your_IHS_7.0_server/lib
```

すべての Web エージェントに対する一般的な準備

以下のセクションでは、すべての Web エージェントに対する一般的な準備を説明します。

エージェント インストールを完了するのに必要な情報の収集

Web エージェントをインストールする前に、以下の情報が必要です。

- エージェントをインストールすることができる SiteMinder 管理者の名前
- ホスト設定オブジェクトの名前。これはトラステッド ホストの設定を定義します。
- エージェント設定オブジェクトの名前(エージェント構成の設定が含まれています) 単一のエージェント設定オブジェクトは多数のエージェントによって参照できます。

任意の WebAgentTrace.conf ファイル変更の保存

WebAgentTrace.conf ファイルを変更していて、既存の Web エージェント上に新しい Web エージェントをインストールする場合は、WebAgentTrace.conf ファイルが上書きされます。そのため、インストールの前に WebAgentTrace.conf ファイルの名前を変更するかバックアップする必要があります。

重要: インストーラが開始すると、既存のファイルは警告なしで上書きされます。オリジナル ファイルをコピーまたはバックアップしないと、以前の設定が失われます。

インストール後、変更を新しいファイルに統合することができます。

Web サーバへの適切なエージェントのインストール

以下の Web エージェントを対応する Web サーバにインストールします。

Web エージェント	Web サーバ
IIS	Microsoft IIS
Domino	IBM Lotus Domino
Sun Java System	Oracle iPlanet
Apache	Apache、HP ベース Apache、IBM HTTP、Oracle HTTP Server Apache Web サーバについてのほとんどの情報はこれらの Web サーバに適用されます。

サポートされる Web サーバとオペレーティング システムのバージョンの詳細については、[テクニカル サポート](#)にアクセスし、SiteMinder r12.0 SP3 プラットフォーム サポート マトリックスを検索してください。

ポリシー サーバ要件

Web エージェントをインストールする前に、ポリシー サーバをインストールして設定し、**Web** エージェントをインストールするシステムと通信できるようにする必要があります。

注: 詳細については、ポリシー サーバドキュメントを参照してください。

ポリシー サーバを以下のアイテムと共に設定する必要があります。

- **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 ホストは 1 つ以上の **SiteMinder Web** エージェントがインストールされているクライアント コンピュータです。トラステッドホストは、物理システムを指します。トラステッド ホストを登録する権限を持つ管理者が必要です。

- **エージェント識別情報**

エージェント識別情報により、**Web** エージェントをホスティングする **Web** サーバ インスタンスの名前と IP アドレスの間にマッピングが確立されます。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**

このオブジェクトはポリシー サーバ上に存在し、2 つの間の初期接続がなされた後、**Web** 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。

トラステッド ホストは 1 つ以上の **SiteMinder Web** エージェントがインストールされているクライアント コンピュータです。トラステッドホストは、物理システムを指します。

このオブジェクトを、ホスト登録が成功した後でトラステッド ホストにインストールされるトラステッド ホストの環境設定ファイル (**SmHost.conf**) と混同しないでください。**SmHost.conf** ファイル内の設定を使用して、最初の接続でのみ **Web** エージェントはポリシー サーバに接続することができます。後続の接続はホスト設定オブジェクトによって管理されます。

- **エージェント設定オブジェクト**

このオブジェクトには、**Web** エージェントの設定を定義するパラメータが含まれています。必須パラメータは、**Web** エージェントをホストしている **Web** サーバのタイプによって異なります。

すべてのエージェントで必要とされるエージェント設定パラメータ

すべてのエージェントは、以下のパラメータに値を設定する必要があります。

DefaultAgentName

AgentName パラメータに指定されたエージェント名がない IP アドレスまたはインターフェースのリクエストを受け取った場合に **Web** エージェントが使用する名前を定義します。

仮想サーバを使用している場合、各仮想サーバに別々の **Web** エージェントを定義する代わりに、**DefaultAgentName** を使用することによって、**SiteMinder** 環境を迅速にセットアップできます。

重要: **DefaultAgentName** パラメータに値を指定しない場合は、**AgentName** パラメータにすべてのエージェント ID をリストする必要があります。そうしないと、ポリシー サーバは **Web** 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: 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。大文字と小文字は区別されません。たとえば、**MyAgent** と **myagent** という名前は、同じように処理されます。

前のパラメータ エントリの値は、管理 UI で定義したエージェントオブジェクトの名前と正確に一致する必要があります。

Domino Web エージェントで必要とされるエージェント設定パラメータ

すべてのエージェントで必要とされるパラメータに加えて、Domino Web エージェントは、以下のパラメータにも値を設定する必要があります。

DominoDefaultUser

Domino Web エージェントが SiteMinder が事前に Domino サーバへの別のディレクトリに対して認証したユーザを識別する名前を指定します。

重要: このパラメータがローカル設定ファイルに格納される場合は、暗号化する必要があります。このパラメータを暗号化するには、**encryptkey** ツールを使用します。パラメータを変更する場合は、ローカル設定ファイルを直接編集しないでください。

デフォルト: デフォルトなし

DominoSuperUser

Domino サーバ上のすべてのリソースへのアクセス権があるユーザを識別し、SiteMinder に正常にログインしたすべてのユーザが Domino SuperUser として Domino にログインすることを確認します。

この値は暗号化できます。

このパラメータは以下のパラメータに影響します。

- SkipDominoAuth

デフォルト: デフォルトなし

IIS Web エージェントで必要とされるエージェント設定パラメータ

すべてのエージェントで必要とされるパラメータに加えて、IIS Web エージェントは、特定の状況で以下のパラメータに値を設定する必要がある場合があります。

DefaultUserName

プロキシユーザとして IIS リソースにアクセスするために使用される Windows ユーザの名前を指定します。SiteMinder で保護されている IIS Web サーバのリソースにアクセスするユーザが、必要なサーバアクセス権限を持つユーザではないことがあります。たとえば、UNIX システム上の LDAP ユーザ ディレクトリに格納されているユーザは、IIS Web サーバの Windows システムへのアクセス権限がない場合があります。

Web エージェントが SiteMinder によってアクセスを許可されたユーザのプロキシユーザ アカウントとして機能するためには、NT 管理者によって割り当てられたこの NT ユーザ アカウントを使用する必要があります。

デフォルト: デフォルトなし

DefaultPassword

プロキシユーザとして IIS リソースにアクセスするために使用される関連する Windows ユーザのためのデフォルトのパスワードを指定します。

重要: このパラメータを暗号化する場合は、エージェント設定オブジェクトの中心に設定します。このパラメータがローカル設定ファイルに設定されると、それは暗号化されず、安全性が低下します。

デフォルト: デフォルトなし

SiteMinder で保護されている IIS Web サーバのリソースにアクセスするユーザが、必要なサーバアクセス権限を持つユーザではないことがあります。Web エージェントが SiteMinder によってアクセスを許可されたユーザーのプロキシユーザアカウントとして機能するためには、NT 管理者によって割り当てられたこの NT ユーザーアカウントを使用する必要があります。

以下のいずれかを行う予定である場合は、前のそれぞれのパラメータに値を指定しないでください。

- NTLM 認証方式を使用します。
- Windows ユーザ セキュリティ コンテキスト機能を有効にします。

パスワード サービスの準備

以下のセクションでは、パスワード サービスの前提条件およびガイドラインについて説明します。

JSP パスワード サービスの ServletExec の CLASSPATH を修復する(Windows)

Windows システムに JSP ベースのパスワード サービスをインストールし、既存のサーブレットまたは Password Services.jsp にアクセスした場合にサーブレットが見つからないとのエラー メッセージが表示された場合は、ServletExec クラスパスが正確であることを確認します。

クラスパスが正しく表示されるのに、エラーがまだ発生する場合、classpath.pref ファイルを修復する必要がある場合があります。

ServletExec クラスパスを修復する方法

1. ServletExec 管理インターフェースを使用して Java 仮想マシン用のクラスパスを定義します。詳細については、<ServletExec> のドキュメントを参照してください。

注: ServletExec ドキュメントには [New Atlanta Web サイト](#) でアクセスできます。

2. Sun Java System Web サーバまたは IIS Admin サービスを再起動します。これにより、ServletExec は classpath.pref を書き込むように強制されます。
3. Windows 2000 については、以下の手順に従います。
 - a. IIS Admin サービスを停止します。
 - b. IIS Admin サービスを手動で開始 *せず* に、World Wide Web 発行サービスを開始します。
 - c. エラーがまだ発生する場合は、引き続き以下の手順に従います。
 - d. classpath.pref ファイルをテキスト エディタで開きます。
 - e. すべてのエントリをセミコロン (;) で区切られた単一行に折り畳みます。
 - f. ファイルを保存します。
 - g. Web サーバを再起動します。

パスワード サービスとフォーム ディレクトリ

初めて Web エージェントをインストールする場合、インストール プログラムは Web エージェント ホーム ディレクトリに以下のフォルダを作成します。

- jpw_default および jpw (パスワード サービス用)
- pw_default および pw (パスワード サービス用)
- samples_default およびサンプル (標準フォーム用)

jpw、pw、およびサンプル ディレクトリはカスタマイズするテンプレートおよびフォームが含まれる作業ディレクトリです。「デフォルト」バージョンはオリジナルドキュメントのバックアップ ディレクトリです。

第 2 章: Windows システムに Web エージェントをインストールする

このセクションには、以下のトピックが含まれています。

[Windows での GUI モードインストールの実行 \(P. 31\)](#)

[Windows での無人インストール \(P. 34\)](#)

[Windows に Web エージェントを再インストールします。 \(P. 37\)](#)

[インストール履歴ログ ファイル \(P. 38\)](#)

[Windows 上でトラステッド ホストとしてシステムを登録する \(P. 39\)](#)

[1 つのシステムに複数のトラステッド ホストを登録する \(Windows\) \(P. 49\)](#)

Windows での GUI モードインストールの実行

エージェントをインストールするには、Web サーバを実行するコンピュータにログインする必要があります。

1. 実行しているすべてのアプリケーションを終了して Web サーバを停止します。
2. [テクニカル サポート](#)からインストール ファイルをダウンロードします。
3. win フォルダに移動し、ご使用のオペレーティング システム用の実行可能ファイルを実行します。

`ca-wa-version-winprocessor_type.exe`

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

インストール プログラムはファイルを準備します。

4. [概要]ダイアログ ボックスの情報を確認し、[次へ]をクリックします。
5. 使用許諾契約を読んで、許諾契約に同意する場合はラジオ ボタンをオンにします。[次へ]をクリックします。

許諾契約に同意しない場合、インストールは終了します。

6. [重要な情報]ダイアログ ボックスの注意事項を読んで、[次へ]をクリックします。
7. [インストールフォルダの選択]ダイアログ ボックスで、デフォルトの場所を使用するか、または[選択]ボタンを使用して別の場所を選択します。[次へ]をクリックします。

デフォルト以外の場所を選択した後で、デフォルトのディレクトリに戻す場合は、[デフォルトのフォルダを復元]をクリックします。

8. [ショートカットフォルダの選択]ダイアログ ボックスでエージェント設定ウィザードの場所を選択し、[次へ]をクリックします。

すべてのユーザに設定ウィザードへのアクセスを許可するには、[すべてのユーザにアイコンを作成]チェック ボックスがオンになっていることを確認します。そうでない場合は、このオプションをオフにします。

9. インストール前のサマリダイアログ ボックスで情報を確認し、[インストール]をクリックします。

注: インストール プログラムは、特定のシステム dlls の新しいバージョンがシステムにインストールされていることを検出する場合があります。その場合は、これらの新しいファイルを古いファイルで上書きするかどうか尋ねます。このメッセージが表示された場合は、[すべていいえ]を選択します。

Web エージェント ファイルは指定された場所にコピーされます。後で、[Web エージェント設定]ダイアログ ボックスが表示されます。

10. 以下のいずれかのオプションを選択します。
 - はい エージェントを今すぐ設定します。
 - いいえ。後でエージェントを設定します。

インストール プログラムがロックされたエージェント ファイルを検出した場合、それを再設定する代わりにシステムを再起動するようにプロンプトが表示されます。システムを自動的に再起動するか、後で独自に再起動するかを選択します。

重要: IIS Web サーバにエージェントをインストールしている場合は、インストール後すぐにエージェントを設定しないでください。IIS Web エージェントをインストールした後に Web サーバを再起動し、その後でお使いのバージョンの IIS 用の設定を手動で実行してください。

11. エージェントを設定しないように選択した場合は、[インストール完了]ダイアログ ボックスが表示され、システムを再起動するようにプロンプトが表示されます。
12. [終了]をクリックします。

エージェントを自動的に設定するオプションを選択した場合、インストールプログラムは Web エージェント設定ウィザードを準備し、トラステッド ホスト登録および設定プロセスを開始します。

以下の手順を実行します。

- トラステッドホストを登録する。これは、エージェントを設定する前、または後で行うことができます。しかし、トラステッド ホストが登録されていないと、エージェントはポリシー サーバと正常に通信することができません。
- Web エージェントを設定する。

インストール上の注意事項:

- インストールの後、`web_agent_home¥install_config_info` でインストール ログ ファイルを確認することができます。ファイル名は次のとおりです:

`CA_SiteMinder_Web_Agent_version_InstallLog.log`

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:¥Program Files¥CA¥webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
- インストール後すぐに Web エージェント設定ウィザードを開始しないように選択できますが、インストール後にマシンを再起動する必要がある場合があります。その場合は、エージェントを設定する準備ができたなら、ウィザードを手動で開始します。

詳細情報

[UNIX 上でトラステッド ホストとしてシステムを登録する](#) (P. 63)

[IIS Web エージェントの設定](#) (P. 83)

[Oracle iPlanet Web エージェントの設定](#) (P. 127)

[Apache Web エージェントを設定する](#) (P. 137)

[Domino Web エージェントを設定する](#) (P. 147)

[IIS 6.0 に SiteMinder Web エージェントを設定する方法](#) (P. 107)

Windows での無人インストール

1つのシステムに **Web** エージェントをインストールした後で、エージェントの無人インストール機能を使用して、他の **Web** サーバ上のインストールを自動化できます。無人インストールでは、ユーザによる操作なしで **Web** エージェントをインストールするまたはアンインストールすることができます。

無人インストールは、初期 **GUI** または **コンソール** モード **Web** エージェントインストールからの値で最初に設定されたプロパティファイルを使用します。そのため、**Web** エージェントを最初にインストールしたシステムと同じプラットフォームおよび **Web** サーバ イメージのシステムでのみ無人インストールを実行できます。たとえば、エージェントを **Oracle iPlanet Web** サーバの **Windows** システムに最初にインストールした場合は、プロパティファイルを使用して **Apache Web** サーバの **UNIX** システムで無人インストールを実行することはできません。

Windows での無人インストールの準備

無人インストールでは、`ca-wa-installer.properties` ファイルを使用してネットワーク内のすべてのエージェントに Web エージェントインストールのセットアップを伝達します。このプロパティファイルでは、インストール パラメータを定義し、ファイルおよび Web エージェント実行可能ファイルをネットワーク内の任意の Web サーバにコピーして無人インストールを実行します。

`ca-wa-installer.properties` ファイルは以下の場所にインストールされます。

`web_agent_home\install_config_info`

ファイルのデフォルトパラメータおよびパスは、初期 Web エージェントのインストール中に入力された情報を反映します。

無人インストールの準備方法

1. Web エージェントの初期インストールを実行します。
2. `ca-wa-installer.properties` ファイルを開いてファイル内のパラメータを変更します。パラメータは以下のとおりです。
 - `USER_INSTALL_DIR` - Web エージェントのインストールされた場所を指定します。インストール ディレクトリに完全パスを入力します。
 - `USER_SHORTCUTS` - Web エージェント設定ウィザードのショートカットをインストールする場所を指定します。任意の場所へのパスを入力します。(Windows のみ)
 - `USER_REQUESTED_RESTART` - インストールプログラムが必要に応じて Windows マシンを再起動するかどうか指定します。再起動を許可する場合は[はい]に設定します。(Windows のみ)
3. ファイルを保存します。

Windows での無人インストールの実行

初期 Web エージェントのインストールが完了し、必要に応じて `ca-wa-installer.properties` ファイルが変更されています。これで、このファイルを使用して次の Web エージェントのインストールを実行できます。

Web エージェントの無人インストールを実行する方法

1. Web エージェントがすでにインストールされているシステムから、以下のファイルをローカル ディレクトリにコピーします。
 - a. システムに存在する場所から `ca-wa-version-win32.exe` (エージェント実行可能ファイル)
 - b. `web_agent_home¥install_config_info` から `ca-wa-installer.properties` ファイル

2. コマンド ウィンドウを開いて、ファイルをコピーしたディレクトリへ移動します。

重要: Windows Server 2008 上で SiteMinder ユーティリティまたは実行可能ファイルを実行している場合は、管理者としてシステムにログインしている場合でも、管理者権限でコマンドライン ウィンドウを開きます。詳細については、お使いの 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。

3. 以下の例に示すように、`-f` および `-i silent` オプションを使用してインストール実行可能ファイルを実行します。

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

実行可能ファイルおよびプロパティファイルが配置されているディレクトリからインストールを実行する場合は、コマンドは次のようになります。

```
ca-wa-<version>-win32.exe -f ca-wa-installer.properties -i silent
```

注: これらのファイルが存在するディレクトリにない場合は、各ファイルへのフルパスを指定する必要があります。ディレクトリパスにスペースがある場合は、パス全体を引用符で囲みます。

インストールが完了したら、コマンド プロンプトに戻ります。

4. `web_agent_home¥install_config_info` ディレクトリにある `CA_SiteMinder_Web_Agent_version_InstallLog.log` ファイルを参照することにより、インストールが正常に完了したことを確認します。このログ ファイルには、インストールの結果が含まれています。
5. トラストド ホストを登録し、Web エージェントを設定します。

注: IIS 6.0 サーバにエージェントを設定する場合は、インストール後すぐにエージェントを設定しないでください。エージェントを設定する前に実行する必要のあるいくつかのセットアップ手順があります。

詳細情報

[IIS 6.0 に SiteMinder Web エージェントを設定する方法 \(P. 107\)](#)

[無人モードで任意の Web エージェントを設定する方法 \(P. 157\)](#)

Windows 上で進行中の無人インストールを停止する方法

手動で Windows システム上の無人インストールを停止するには、以下の手順に従います。

1. Windows タスク マネージャを開きます。
2. 以下のプロセスを停止します。
 - ca-wa-version-win.exe
 - wa_install.exe

Windows での Web エージェントの再インストール

Web エージェントを再インストールして見つからないアプリケーション ファイルをリストアすることができます。この手順については、既存の Web エージェントをアンインストールする必要はありません。インストール手順を繰り返すことによって、既存の Web エージェント ファイル上で再インストールを実行するだけです。

Windows に Web エージェントを再インストールするには、以下の手順に従います。

1. 以下のバックアップ コピーを作成します。
 - Windows レジストリの設定
 - Web エージェントの構成設定。
2. GUI インストーラを使用して、Windows システムに Web エージェントをインストールします。

インストール履歴ログ ファイル

インストーラは、次の情報を備えたログ ファイルを作成します。

- 製品名
- インストールされている場所
- 完全な(フル)バージョン番号

このファイルは以下の場所で作成されます。

Windows

C:\Program Files\CA\install-info\ca-install-history.log

UNIX

/opt/ca/install-info/ca-install-history.log

Windows 上でトラステッド ホストとしてシステムを登録する

トラステッド ホストは 1 つ以上の SiteMinder Web エージェントがインストールされているクライアントコンピュータです。

Web エージェントとポリシー サーバの間の接続を確立するには、ポリシー サーバ上にトラステッド ホスト オブジェクトを作成して Web サーバを登録する必要があります。Web エージェントが動作するためには、Web サーバに、ポリシー サーバ上の対応するトラステッド ホスト オブジェクトが必要です。

注: Web エージェントをシステムにインストールして設定するたびにホストを登録するのではなく、ホストを登録するのは一度だけです。

トラステッド ホストを登録する方法

1. 必要に応じて、Web エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザード ショートカットを配置した場合、手順は異なります。

注: インストールの直後に Web エージェントを設定することを選択した場合、SiteMinder は設定ウィザードを開始します。

2. [ホスト登録]ダイアログ ボックスで、
 - a. 今すぐホストを登録する場合は[はい]を選択し、後でホストを登録する場合は[いいえ]を選択します。
 - b. [次へ]をクリックします。
3. [管理者の登録]ダイアログ ボックスで、トラステッド ホストを登録する権限を持つ管理者を識別するために以下のフィールドを完了し、[次へ]をクリックします。
 - [管理者ユーザ名] - ポリシー サーバでホストを登録することを許可された管理者の名前を入力します。

この管理者は、ポリシー サーバですでに定義されていて、トラステッド ホストの登録設定の権限を所有している必要があります。デフォルトの管理者は SiteMinder です。
 - [管理パスワード] - 管理者のパスワードを入力します。
 - [管理者パスワードの確認] - パスワードを再入力します。

- [共有秘密キーのロールオーバーの有効化] - トラステッド ホストとポリシー サーバの間の通信を暗号化するために使用される共有秘密キーを定期的に変更する場合は、このチェック ボックスをオンにします。この機能が動作するには、キーのロールオーバーがポリシー サーバで有効である必要があります。

共有秘密キーのロールオーバーを無効にするかまたは後でそれを有効にするには、トラステッド ホストを再登録するか、C および Perl スクリプト インターフェイスでポリシー 管理 API を使用して共有秘密キーのロールオーバーを有効または無効にします。

4. [トラステッド ホスト名] および [設定オブジェクト] ダイアログ ボックスで、2 つのフィールドに値を入力し、[次へ] をクリックします。
 - a. [トラステッド ホスト名] フィールドに、ポリシー サーバにトラステッド ホストを示す一意の名前を入力します。この名前は登録している物理クライアント システムと同じである必要がありません。これは、mytrustedhost など、任意の固有の名前にすることができます。

注: この名前はトラステッド ホストの中で一意である必要があり、その他の Web エージェントの名前と一致してはなりません。

- b. [ホスト設定オブジェクト] フィールドで、ポリシー サーバに指定されたホスト設定オブジェクトの名前を入力し、[次へ] をクリックします。

このオブジェクトは、トラステッド ホストとポリシー サーバの間の接続を定義します。たとえば、デフォルトを使用する場合は DefaultHostSettings を入力します。ほとんどの場合は、独自のホスト設定オブジェクトが作成されています。

注: 指定するエンタリは、ポリシー サーバに設定されたホスト設定オブジェクト エンタリと一致する必要があります。

5. [ポリシー サーバの IP アドレス] ダイアログ ボックスで、
 - a. IP アドレスまたはホスト名、およびホストを登録しているポリシー サーバの認証ポートを入力します。デフォルトのポートは 44442 です。ポートを指定しない場合、デフォルトが使用されます。

デフォルト以外のポート番号を指定することもできますが、ポリシー サーバがデフォルト以外のポートを使用するように設定されていて、トラステッド ホストを登録するときにそれを省略した場合、SiteMinder は以下のエラーを表示します。

登録に失敗しました (不正な ipAddress[:port] または認証サーバ (-1) に接続できません)

また、デフォルト以外のポートを指定した場合、そのポートがポリシー サーバの認証、認可、および監査ポートに使用されることに注意してください。ただし、統合サーバは、任意のポートの任意のエージェントリクエストに応答します。SmHost.conf ファイルのエントリは以下のようになります。

```
policyserver="ip_address,5555,5555,5555"
```

- b. [追加]をクリックします。

1 つ以上のポリシー サーバを追加できますが、ホスト登録については、リストの最初のサーバのみが使用されます。

複数のポリシー サーバが指定される場合、エージェントはそれらをブートストラップ サーバとして使用します。エージェントが開始されると、Web エージェントには接続してそのホスト設定オブジェクトを取得できるいくつかのポリシー サーバがあります。ホスト設定オブジェクトが取得された後、ブートストラップ ポリシー サーバはそのサーバ プロセスで使用されなくなります。ホスト設定オブジェクトには別のサーバのセットを含めることができます。それには、任意のブートストラップ サーバが含まれる場合も、含まれない場合もあります。

- c. [次へ]をクリックします。

6. FIPS 暗号化を使用する場合は、以下のオプションのうちの 1 つを選択します。

FIPS 互換モード(デフォルト)

非 FIPS モードを指定します。これにより、ポリシー サーバとエージェントは、既存の SiteMinder 暗号化アルゴリズムを使用して情報を読み取って書き込むことができます。組織で FIPS に準拠しているアルゴリズムの使用を必要としない場合は、ポリシー サーバとエージェントは、これ以上の設定なしに、非 FIPS モードで動作することができます。

FIPS 移行モード

FIPS 移行モードを指定します。これは、SiteMinder の以前のバージョンから FIPS のフル モードにアップグレードする場合に使用されます。ポリシー サーバとエージェントは、ユーザが FIPS 140-2 の承認済みアルゴリズムのみを使用する環境に移行しても、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。

FIPS のみのモード

FIPS フル モードを指定します。これは、ポリシー サーバとエージェントが FIPS 140-2 アルゴリズムのみを使用して、情報の読み取りと書き込みを行うことを要求されます。

重要: フル FIPS モードで実行している SiteMinder インストールでは、エージェント、エージェント API の以前のバージョンを使用するカスタムソフトウェア、および PM API またはポリシー サーバが公開するその他の API を使用するカスタムソフトウェアなどの、SiteMinder の以前のバージョンと相互操作できないか、または下位互換性がありません。これらすべてのソフトウェアを対応する SDK のバージョンと再度リンクし、フル FIPS モードに必要なサポートを獲得する必要があります。

FIPS 暗号化を使用しない場合は、デフォルトを受け入れます。

7. [次へ]をクリックします。
8. ホスト設定ファイルの `SmHost.conf` のデフォルトの場所を受け入れるか、または[選択]をクリックして別の場所を選択します。[次へ]をクリックします。
デフォルト以外の場所を選択した後で、デフォルトのディレクトリに戻す場合は、[デフォルトのフォルダを復元]をクリックします。

ホストは登録されていて、ホスト設定ファイルの `SmHost.conf` は `web_agent_home/config` 内に作成されます。このファイルは変更できません。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
 - UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
9. [続行]をクリックします。
 10. 以下の該当するタスクを実行して設定を続行します。
 - IIS Web エージェントを設定する
 - Oracle iPlanet Web エージェントの設定
 - Apache Web エージェントを設定する
 - Domino Web エージェントを設定する

インストールおよび設定ログ ファイル

インストールの結果を確認したり、Web エージェントのインストール中または設定中に特定の問題を確認するには、`web_agent_home¥install_config_info`にある以下のログ ファイルを確認します。

`ca-wa-details.log`

発生した可能性のある任意の失敗または問題に関する特定の詳細を提供します。

`CA_SiteMinder_Web_Agent_version_InstallLog.log`

正常にインストールされたコンポーネント、および失敗したコンポーネントを含め、インストールの完全な結果を提供します。

SmHost.conf ファイルの変更 (Windows)

Web エージェントおよびカスタム エージェントは、SmHost.conf ファイルの情報を使用してポリシー サーバを検索して初期接続を行うことによって、トラステッド ホストとして動作します。エージェントがポリシー サーバに接続したら、初期接続は閉じます。エージェントとポリシー サーバの間の通信は、ポリシー サーバにあるホスト設定オブジェクトの設定に基づきます。

SmHost.conf ファイルの一部を変更してエージェントからポリシー サーバへの初期接続を変更できます。

SmHost.conf ファイルを変更する方法

1. `web_agent_home/config` ディレクトリに移動します。
2. テキスト エディタで SmHost.conf ファイルを開きます。
3. 変更する以下のいずれかの設定に新しい値を入力します。

重要: ここにリストされたパラメータの設定のみが変更されます。
SmHost.conf ファイルのその他のパラメータの設定は変更されません。

hostconfigobject

トラステッド ホストとして動作しているエージェントとポリシー サーバの間の接続を定義するホスト設定オブジェクトを指定します。この名前は、管理 UI に定義された名前と一致する必要があります。

ホスト設定オブジェクトを変更する場合は、この設定を変更する必要があります。

例: `hostconfigobject="host_configuration_object"`

policyserver

トラステッド ホストが接続しようとするポリシー オブジェクトを指定します。正しい構文は次のとおりです。

`"IP_address, port, port, port"`

デフォルトポートは `44441,44442,44443` ですが、同じ数または 3 つのポートすべてに別の数を使用してデフォルト以外のポートを指定できます。統合サーバは、任意のポートの任意のエージェントリクエストに応答します。

エージェントに追加のブートストラップ サーバを指定するには、ファイルに複数のポリシー サーバエントリを追加します。複数のエントリは、接続してホスト設定オブジェクトを取得できるいくつかのポリシー サーバをエージェントに提供します。ホスト設定オブジェクトが取得されたら、ブートストラップ サーバはそのサーバプロセスに必要でなくなります。

複数のエントリはホスト登録中に、またはこのパラメータを変更することによって、追加できます。ポリシー サーバが SiteMinder 環境から削除されるか、使用されなくなった場合は、エントリを削除します。

重要: エージェントが多重プロセス Web サーバに設定されている場合は、プライマリ ポリシー サーバが失敗しても子プロセスがセカンダリポリシー サーバへの接続を確立できるように、複数のポリシー サーバエントリを指定することをお勧めします。ファイルに 1 つのポリシー サーバのみがリストされていて、そのポリシー サーバに接続できない場合は、新しい子プロセスが開始されるたびに Web エージェントを初期化できなくなります。

デフォルト: `IP_address, 44441,44442,44443`

例(単一エントリの構文): `"IP_address, port,port,port"`

例(複数エントリの構文、各ポリシー サーバを別々の行に配置する):

```
policyserver="123.122.1.1, 44441,44442,44443"  
policyserver="111.222.2.2, 44441,44442,44443"  
policyserver="321.123.1.1, 44441,44442,44443"
```

requesttimeout

トラステッド ホストとして動作するエージェントがポリシー サーバを利用できないと決定するまでに待機する間の秒の間隔を指定します。ポリシー サーバが過剰なトラフィックまたは遅いネットワーク接続が原因でビジーな場合は、タイムアウト値を増やすことができます。

デフォルト: 60

例: `requesttimeout="60"`

4. SmHost.Conf ファイルを保存して閉じます。

SmHost.Conf ファイルに対する変更が適用されます。

登録ツールを使用したトラステッド ホストの再登録 (Windows)

初めてサーバに Web エージェントをインストールする場合、トラステッド ホストとしてそのサーバを登録するようにプロンプトが表示されます。トラステッド ホストが登録されたら、後続のエージェント インストールで再登録する必要はありません。以下のように、エージェントのインストールに関係なく、トラステッド ホストを再登録する必要がある場合がいくつかあります。

- SiteMinder 環境に変更が行われた場合に、トラステッド ホストの名前を変更するとき。
- トラステッド ホストが 管理 UI で削除された場合に、トラステッド ホストを登録するとき。
- トラステッド ホストのポリシー オブジェクトがポリシー ストアから削除された場合、またはポリシー ストアが失われた場合に、トラステッド ホストを登録するとき。
- トラステッド ホストとポリシー サーバの間の接続を保護する共有秘密キーを変更するとき。
- SmHost.conf 設定ファイルが喪失した場合に、それを再作成するとき。
- 既存のトラステッド ホストを最初に削除せずに上書きするとき。

登録ツール、smregghost はトラステッド ホストを再登録します。Web エージェントをインストールすると、このツールは `web_agent_home¥bin` ディレクトリにインストールされます。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:¥Program Files¥CA¥webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

登録ツールを使用してトラステッド ホストを再登録する方法

1. コマンド プロンプト ウィンドウを開きます。
2. 以下の必要な引数を使用して、smregghost コマンドを入力します。

```
smregghost -i policy_server_IP_address:[port]  
-u administrator_username -p Administrator_password  
-hn hostname_for_registration -hc host_configuration_object
```

注: スペースで各コマンドの引数と値を区切ります。スペースが含まれているすべての値を二重引用符 (") で囲みます。以下の例を

参照してください。

```
smreghost -i 123.123.1.1 -u SiteMinder -p mypw -hn "host computer A"  
-hc DefaultHostSettings
```

以下の例では `-o` 引数が含まれています。

```
smreghost -i 123.123.1.1 -u SiteMinder -p mypw -hn "host computer A"  
-hc DefaultHostSettings -o
```

以下の引き数は `smreghost` コマンドと共に使用されます。

`-i policy_server_IP_address:port`

このホストを登録するポリシー サーバの IP アドレスを示します。デフォルトポートを使用していない場合のみ、認証サーバのポートを指定します。

デフォルト以外のポートにすることができるポート番号を指定した場合、そのポートは 3 つのポリシー サーバのサーバ(認証、認可、説明)に使用されますが、統合サーバは任意のポートの任意のエージェントリクエストに応答します。たとえば、ポート `55555` を指定した場合、`SmHost.conf` ファイルのポリシー サーバ エントリは以下を表示します。

```
"policy_server_ip_address,5555,5555,5555"
```

例: (IPv4) 127.0.0.1,55555

例: (IPv6) [2001:DB8::/32][:55555]

`-u administrator_username`

トラステッド ホストを登録する権限を持つ SiteMinder 管理者の名前を指定します。

`-p Administrator_password`

トラステッド ホストを登録することを許可された管理者のパスワードを示します。

`-hn hostname_for_registration`

登録されるホストの名前を示します。これはホストを識別する任意の名前にすることができますが、一意である必要があります。登録の後、この名前は 管理 UI の [トラステッド ホスト] リストに配置されます。

`-hc host_config_object`

ポリシー サーバに設定されたホスト設定オブジェクトの名前を示します。トラステッド ホストを登録できるようにする前に、このオブジェクトがポリシー サーバに存在する必要があります。

- sh *shared_secret*

Web エージェント用の共有秘密キーを指定します。これは、ローカル Web サーバの **SmHost.conf** ファイルに格納されています。この引数は、ローカル Web サーバ上でのみ共有秘密キーを変更します。ポリシーサーバは接続されていません。

-rs

共有秘密キーがポリシーサーバによって自動的に更新(ロールオーバー)されるかどうかを示します。この引数は、共有秘密キーを更新するようにポリシーサーバに指示します。

-f *path_to_host_config_file*

(オプション) 登録データが含まれているファイルへのフルパスを示します。デフォルトのファイルは、**SmHost.conf** です。パスを指定しない場合、ファイルは **smreghost** ツールを実行している場所にインストールされます。

既存のホスト設定ファイルと同じ名前を使用する場合、ツールはオリジナルファイルをバックアップし、バックアップファイル名に **.bk** 拡張子を追加します。

- cf *FIPS モード*

以下のいずれかの **FIPS** モードを選択してください。

- 互換 - 非 **FIPS** モードを指定します。これにより、ポリシーサーバとエージェントは、既存の **SiteMinder** 暗号化アルゴリズムを使用して情報を読み取って書き込むことができます。組織で **FIPS** に準拠しているアルゴリズムの使用を必要としない場合は、ポリシーサーバとエージェントは、これ以上の設定なしに、非 **FIPS** モードで動作することができます。
- 移行 - **FIPS** 移行モードを指定します。これは、**SiteMinder** の以前のバージョンから **FIPS** のフルモードにアップグレードする場合に使用されます。ポリシーサーバとエージェントは、ユーザが **FIPS 140-2** の承認済みアルゴリズムのみを使用する環境に移行しても、既存の **SiteMinder** 暗号化アルゴリズムを引き続き使用します。
- 専用 - **FIPS** フルモードを指定します。これは、ポリシーサーバとエージェントが **FIPS 140-2** アルゴリズムのみを使用して、情報の読み取りと書き込みを行うことを要求されます。

重要: フル FIPS モードで実行している SiteMinder インストールでは、エージェント、エージェント API の以前のバージョンを使用するカスタムソフトウェア、および PM API またはポリシー サーバが公開するその他の API を使用するカスタムソフトウェアなどの、SiteMinder の以前のバージョンと相互操作できないか、または下位互換性がありません。これらすべてのソフトウェアを対応する SDK のバージョンと再度リンクし、フル FIPS モードに必要なサポートを獲得する必要があります。

このスイッチが使用されなかったり、ユーザがモードを指定せずにスイッチを使用する場合、デフォルト設定が使用されます。

デフォルト: 互換

注: 使用されている FIPS 認定モジュールおよびアルゴリズム、保護されているデータおよび、SiteMinder 暗号境界の詳細については、「ポリシー サーバ管理ガイド」を参照してください。

-o

既存のトラステッド ホストに上書きします。この引数を使用しない場合は、smreghost コマンドを使用する前に 管理 UI を備えた既存のトラステッド ホストを削除する必要があります。この引き数で smreghost コマンドを使用することを推奨します。

トラステッド ホストは再登録されます。

1 つのシステムに複数のトラステッド ホストを登録する (Windows)

Web サーバとエージェントがインストールされている各マシンに対し、一般的に登録するトラステッド ホストは 1 つのみです。ただし、1 台のコンピュータに複数のトラステッド ホストを登録して各 SiteMinder クライアントに別々の接続を作成することができます。複数のトラステッド ホストを使用すると、ポリシー サーバとのコミュニケーションを必要とする各クライアントに対して一意の共有秘密キーおよび安全な接続が確保されます。

ほとんどのインストールでは、これは推奨された設定ではありません。しかしながら、これは、SiteMinder エージェントによって保護された各クライアントまたはクライアントのグループ アプリケーションに別々のセキュアなチャンネルを必要とするサイトのオプションです。たとえば、アプリケーション サービス プロバイダには異なるアプリケーションがインストールされた多数のクライアントコンピュータがあるとして。それぞれのアプリケーションに安全な接続を望む場合は、複数のトラステッドホストを登録すると、実現することができます。ポリシー サーバはその後、各クライアント接続に一意の共有秘密キーを発行します。

複数のトラステッドホストを登録するには、以下の方法のうちの1つを使用します。

- 設定ウィザードでの登録: トラステッドホストとして追加のサーバを登録するには、登録処理を再度実行します。ただし、SmHost.conf ファイルの場所を指定するようにプロンプトが表示された場合は、一意のパスを入力します。新しいホストを登録する場合は、既存の Web サーバの SmHost.conf ファイルを使用しないでください。使用すると、そのファイルが上書きされます。SmHost.conf という名前を使用するか、またはファイルに新しい名前を付けることができます。

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

注: ポリシー サーバでトラステッドホストを登録していて、設定ウィザードを実行して SmHost.conf ファイルで一意のパスを使用せずに次のエージェントを設定した場合は、[ホスト登録]ダイアログ ボックスに警告メッセージが表示されます。このメッセージは、以下のように表示されます。

「警告: このエージェントはポリシー サーバを使用してすでに登録されています。」

- smreghost コマンドライン ツールでの登録: 指定された任意のコンピュータで最初のエージェント インストールを完了した後で、smreghost ツールを実行します。このツールは登録する各トラステッドホストに対して実行できます。

重要: Windows Server 2008 上で SiteMinder ユーティリティまたは実行可能ファイルを実行している場合は、管理者としてシステムにログインしている場合でも、管理者権限でコマンドライン ウィンドウを開きます。詳細については、お使いの 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。

第 3 章: UNIX システムに Web エージェントをインストールする

このセクションには、以下のトピックが含まれています。

[UNIX システムへの Web エージェントのドキュメントのインストール \(P. 51\)](#)

[UNIX システムへの Web エージェントのインストール \(P. 53\)](#)

[インストール後の Web エージェント環境変数の設定 \(P. 61\)](#)

[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 \(P. 62\)](#)

[インストール履歴ログ ファイル \(P. 62\)](#)

[UNIX への Web エージェントの再インストール \(P. 63\)](#)

[UNIX 上でトラステッド ホストとしてシステムを登録する \(P. 63\)](#)

[GUI またはコンソール モードでのトラステッド ホストの登録 \(P. 64\)](#)

[1 つのシステムに複数のトラステッド ホストを登録する \(UNIX\) \(P. 74\)](#)

UNIX システムへの Web エージェントのドキュメントのインストール

Web エージェントのドキュメントを Web エージェントとは別々にインストールします。これは、デフォルトではインストールされていません。Web エージェントをインストールする前に、ドキュメントをインストールしてインストール場所を指定できるようにすることをお勧めします。

注: Web エージェントのドキュメントを既存のポリシー サーバのドキュメントと同じシステムにインストールする予定である場合、インストールにより、ポリシー サーバドキュメントと同じ場所にエージェントのマニュアルが配置されます (例: `policy_server_home/ca_documents`)。場所を指定するように求めるプロンプトは表示されません。

ドキュメントをインストールする方法

1. [テクニカル サポート](#) からドキュメント インストール プログラムをダウンロードし、次に、オペレーティング システムのディレクトリに移動します。
2. オペレーティング システムに適切なインストール ファイルをローカル ディレクトリにコピーし、そのディレクトリに移動します。

注: バイナリファイルは以下の命名規則を使用します。

- `ca-wa-version-operating_system.bin` (ほとんどのバージョン用)
- `ca-wa-version-operating_system-processor-architecture.bin` (特定のプロセッサまたはアーキテクチャタイプを必要とするバージョン用)
- `nete-wa-doc-version-linux.bin (linux 2.1)`

3. コンソール ウィンドウを開いて、バイナリファイルについての許可を確認します。 `chmod` コマンドを実行することにより、インストール ファイルに実行を追加する必要がある場合があります。例:

```
chmod +x ca-wa-version-operating_system.bin
```

4. コンソール ウィンドウから、以下のコマンドのうちの 1 つを使用して、ドキュメント インストールを実行します。

GUI モード:

```
./ca-wa-doc-version-operating_system.bin
```

コンソール モード:

```
./ca-wa-doc-version-operating_system.bin -i console
```

ドキュメントのインストールが開始されます。

5. `Enter` を押しながら全ドキュメントに目を通して、使用許諾契約を読みます。契約に同意する場合は、「Y」を入力してインストールを続行します。
6. 重要な説明を確認し、[次へ]をクリックします。
7. インストール ディレクトリを指定します。

インストール プログラムは、指定したディレクトリに r12.0 SP3 Web エージェントのドキュメントを設置します。

UNIX システムへの Web エージェントのインストール

UNIX システムの Web エージェント インストールにはいくつかのタイプがあります。

注: 64 ビット SUSE Linux 10 システムに Web エージェントをインストールするには、追加の準備が必要になります。

- グラフィカル ユーザ インターフェースからのインストール
- コマンドライン プロンプトに応答するコンソール ウィンドウからのインストール
- 管理者による無人インストールで、ユーザの操作を必要としないインストール ファイルのインストール

お使いの環境に最適なインストールの方法を選択してください。

以下の点に注意してください。

- Web エージェント インストールはいくつかのシステム環境変数を追加して変更します。
- コンソール モードで、インストール プログラムが質問をプロンプトで表示する場合、デフォルト エントリは角かっこ [] 内に表示されます。Enter キーを押してデフォルトを受け入れます。
- これらの手順では、`web_agent_home` は、SiteMinder Web エージェントのインストールされた場所を参照します。
- インストールの後、`web_agent_home` 内にインストール ログ ファイルを見つけることができます。ファイル名は、以下のとおりです。

`CA_SiteMinder_Web_Agent_version_InstallLog.log`

詳細情報

[その他の Web サーバの準備 \(P. 21\)](#)

[Web エージェント インストールによって追加または変更された環境変数 \(P. 225\)](#)

UNIX での GUI モードインストールの実行

エージェントをインストールするには、Web サーバがインストールされているアカウントにログインする必要があります。

注: 既存の r12 Web エージェントを r12 SP1 にアップグレードする場合は、ルートユーザとしてログインする必要があります。新しい r12 SP1 Web エージェントをインストールする場合は、ルート権限は必要ではありません。

UNIX で GUI モードインストールを実行する方法

1. インストールを開始する前に、次の点を確認してください。
 - Web エージェントの GUI モード インストールを実行するか、Exceed アプリケーションを使用して設定ウィザードを実行すると、ダイアログ ボックス内のテキストは利用できないフォントが原因で切り捨てられる場合があります。この制限は Web エージェントのインストールおよび設定に影響を与えません。
 - telnet または他の端末エミュレーションソフトウェアを介して Web エージェントをインストールする場合は、背後で X-Windows セッションを実行して GUI モード インストールを実行する必要があります。さらに、端末に以下のように DISPLAY 変数を設定する必要があります。

```
DISPLAY=111.11.1.12: 0.0
```

```
export DISPLAY
```

X-Windows セッションを使用せずに telnet ウィンドウから GUI モードで実行しようとする、インストーラは Java 例外をスローして終了します。
 - また、コンソール ウィンドウからコマンドライン インストールを実行することもできます。
2. 実行中のすべてのアプリケーションを終了します。
3. /tmp ディレクトリに少なくとも 300 MB の使用可能なディスク領域があることを確認してください。
4. [テクニカル サポート](#) からインストール ファイルをダウンロードします。
5. オペレーティング システムの次のディレクトリに移動します。
6. 該当するバイナリファイルをローカル ディレクトリにコピーして、そのディレクトリに移動します。

注: バイナリファイルは以下の命名規則を使用します。

- `ca-wa-version-operating_system.bin` (ほとんどのバージョン用)
- `ca-wa-version-operating_system-processor-architecture.bin` (特定のプロセッサまたはアーキテクチャタイプを必要とするバージョン用)

7. 権限に応じて、`chmod` コマンドを実行することにより、インストール ファイルに実行可能な権限を追加する必要がある場合があります。例:

```
chmod +x ca-wa-version-operating_system-processor-architecture.bin
```

8. コンソール ウィンドウを開いて、ローカル インストール ディレクトリから以下を入力します。

```
./ca-wa-version-operating_system-processor-architecture.bin
```

インストール プログラムはファイルを準備します。

9. [概要]ダイアログ ボックスで、情報を読んで[次へ]をクリックします。
10. 使用許諾契約を読んで、許諾契約に同意する場合はラジオ ボタンをオンにします。[次へ]をクリックします。
許諾契約に同意しない場合、インストールは終了します。
11. [重要な情報]ダイアログ ボックスの注意事項を読んで、[次へ]をクリックします。
12. [インストール場所の選択]ダイアログ ボックスで、デフォルトの場所を受け入れるか、または[選択]ボタンを使用して別の場所を選択します。[次へ]をクリックします。
デフォルト以外の場所を選択した後で、デフォルトのディレクトリに戻す場合は、[デフォルトのフォルダを復元]をクリックします。
13. インストール前のサマリダイアログ ボックスで情報を確認し、[インストール]をクリックします。
Web エージェント ファイルは指定された場所にインストールされます。
14. [インストール完了]ダイアログ ボックスで、[終了]をクリックします。
15. エージェントをインストールした後で、エージェント設定ウィザードを実行してトラステッド ホストを登録し、Web エージェントを設定します。

インストールおよび設定ログ ファイル

インストールの結果を確認したり、Web エージェントのインストール中または設定中に特定の問題を確認するには、`web_agent_home¥install_config_info` にある以下のログ ファイルを確認します。

`ca-wa-details.log`

発生した可能性のある任意の失敗または問題に関する特定の詳細を提供します。

`CA_SiteMinder_Web_Agent_version_InstallLog.log`

正常にインストールされたコンポーネント、および失敗したコンポーネントを含め、インストールの完全な結果を提供します。

UNIX でのコンソール モードインストールの実行

コンソール モードを使用して、UNIX システムに SiteMinder Web エージェントをインストールできます。

注: 既存の r12 Web エージェントを r12 SP1 にアップグレードする場合は、ルートユーザとしてログインする必要があります。新しい r12 SP1 Web エージェントをインストールする場合は、ルート権限は必要ではありません。

UNIX でコンソール モードインストールを実行する方法

1. 実行しているすべてのアプリケーションを終了して Web サーバを停止します。
2. /tmp ディレクトリに少なくとも 300 MB の使用可能なディスク領域があることを確認してください。
3. [テクニカル サポート](#) からインストール プログラムをダウンロードします。
4. オペレーティング システムの次のディレクトリに移動します。
5. 該当するバイナリファイルをローカル ディレクトリにコピーして、そのディレクトリに移動します。

注: バイナリファイルは以下の命名規則を使用します。

- `ca-wa-version-operating_system.bin` (ほとんどのバージョン用)
 - `ca-wa-version-operating_system-processor-architecture.bin` (特定のプロセッサまたはアーキテクチャタイプを必要とするバージョン用)
6. コンソール ウィンドウを開いて、バイナリファイルについての許可を確認します。インストール ファイルに実行権限を追加する必要がある場合があります。以下に例を示します。

```
chmod +x ca-wa-version-operating_system-processor-architecture.bin
```

7. コマンド プロンプトで、以下のコマンドを入力してコンソール モード インストールを開始します。

```
./ca-wa-version-operating_system-processor-architecture.bin
```

```
-i console
```

`-i console` コマンド引数を使用することによって、コマンド ラインからインストールを実行することができます。

インストールはファイルを準備します。

8. [概要]を確認し、Enter を押して続行します。

インストールは使用許諾契約を準備します。

9. **Enter** を押しながら全契約に目を通して、使用許諾契約を読みます。
10. 契約に同意する場合は、「Y」を入力してインストールを続行します。
11. インストールとドキュメントの詳細については、[重要な情報]セクションを確認してください。

注意事項に目を通して **Enter** キーを押し、インストールを続行します。

12. [インストール場所の選択]セクションで、インストールがエージェントファイルを配置する場所を指定します。デフォルトの場所を使用する場合は、**Enter** キーを押します。

パスを指定する場合は、それに「webagent」という単語を含める必要があります。含まれていない場合は、インストールプログラムはこのフォルダを作成し、パスにそれを追加します。たとえば、`export/ca/wa` を指定した場合、パスは `export/ca/wa/webagent` になります。ただし、パスとして `export/ca/sm_webagent` を指定した場合、インストールプログラムはこれを受け入れます。

13. インストール前のサマリ情報を確認し、**Enter** キーを押して続行します。プログラムはファイルのインストールを開始します。
14. インストールが完了したら、設定ウィザードの検索についての指示と共にメッセージが表示されます。
15. **Enter** を押してインストーラを終了します。
16. エージェントをインストールした後で、エージェント設定ウィザードを実行してトラステッド ホストを登録し、**Web** エージェントを設定します。

UNIX の無人インストール

1つのシステムに **Web** エージェントをインストールした後で、エージェントの無人インストール機能を使用して、他の **Web** サーバ上のインストールを自動化できます。無人インストールでは、ユーザによる操作なしで **Web** エージェントをインストールするまたはアンインストールすることができます。

無人インストールは、初期 **GUI** または **コンソール** モード **Web** エージェントインストールからの値で最初に設定されたプロパティファイルを使用します。そのため、**Web** エージェントを最初にインストールしたシステムと同じプラットフォームおよび **Web** サーバ イメージのシステムでのみ無人インストールを実行できます。たとえば、エージェントを **Oracle iPlanet** の **Solaris** システムにインストールし、プロパティファイルを使用して **Apache Web** サーバの **Linux** システムで無人インストールを実行することはできません。

UNIX での無人インストールの準備

無人インストールでは、`ca-wa-installer.properties` ファイルを使用してネットワーク内のすべてのエージェントに Web エージェントインストールのセットアップを伝達します。このプロパティファイルでは、インストールパラメータを定義し、ファイルおよび Web エージェント実行可能ファイルをネットワーク内の任意の Web サーバにコピーして無人インストールを実行します。

`ca-wa-installer.properties` ファイルは以下の場所にインストールされます。

`web_agent_home/install_config_info`

ファイルのデフォルトパラメータおよびパスは、初期 Web エージェントのインストール中に入力された情報を反映します。

`ca-wa-installer.properties` ファイルをインストールする方法

1. Web エージェントの初期インストールを実行します。
2. `ca-wa-installer.properties` ファイルを開いてパラメータを変更します。
パラメータは以下のとおりです。

パラメータ	意味
<code>USER_SHORTCUTS</code>	Web エージェント設定ショートカットをインストールする場所を指定します。任意の場所へのパスを入力します。 (Windows のみ)
<code>USER_INSTALL_DIR</code>	Web エージェントのインストールされた場所を指定します。インストールディレクトリに完全パスを入力します。
<code>USER_REQUESTED_RESTART</code>	インストールプログラムが必要に応じて Windows マシンを再起動するかどうか指定します。再起動を許可する場合は[はい]に設定します。(Windows のみ)

3. ファイルを保存します。

UNIX での無人インストールの実行

初期 Web エージェントのインストールが完了し、必要に応じて `ca-wa-installer.properties` ファイルが変更されています。これで、このファイルを使用して次の Web エージェントのインストールを実行できます。

Web エージェントの無人インストールを実行する方法

1. Web エージェントがすでにインストールされているシステムから、以下のファイルをローカル ディレクトリにコピーします。
 - a. システムに存在する場所から `ca-wa-version-operating_system.bin` (エージェント実行可能ファイル)
 - b. `web_agent_home/install_config_info` から `ca-wa-installer.properties` ファイルをコピーします。
2. コンソール ウィンドウを開いて、2 つのファイルをコピーしたディレクトリへ移動します。
3. 以下のように、`-f` および `-i silent` オプションを使用してインストール実行可能ファイルを実行します。

```
agent_binary -f properties_file -i silent
```

注: これらのファイルが存在するディレクトリにいない場合は、各ファイルへのフルパスを指定する必要があります。

実行可能ファイルおよびプロパティファイルが配置されているディレクトリからインストールを実行する場合は、コマンドは次のようになります。

```
./ca-wa-version-operating_system.bin -f ca-wa-installer.properties  
-i silent
```

インストールが完了したら、コマンド プロンプトに戻ります。

4. `web_agent_home/install_config_info` ディレクトリにある `CA_SiteMinder_Web_Agent_version_InstallLog.log` ファイル。このログ ファイルには、インストールの結果が含まれています。
5. トラステッド ホストを登録し、Web エージェントを設定します。

UNIX 上で進行中の無人インストールを停止する

手動でインストールを停止するには、`Ctrl+C` を押します。

インストール後の Web エージェント環境変数の設定

Web エージェントをインストールした後で、`ca_wa_env.sh` スクリプトを使用して Web エージェント環境変数を設定できます。ほとんどの UNIX プラットフォームにインストールされている Web エージェントでスクリプトを実行することで、Web エージェントと Web サーバが確実に連携できるようにします。スクリプトは Web エージェントが必要とする環境変数を設定します。

`ca_wa_env.sh` スクリプトが拡張され、以下の環境変数を設定します。

- `NETE_WA_ROOT`
- `PATH`
- `NETE_WA_PATH`
- `LD_LIBRARY_PATH`

注: Web エージェントでは、`LD_LIBRARY_PATH` には `libm.so` の以前のバージョンが含まれた他のディレクトリの前に `/usr/lib` が含まれている必要があります。

- `SHLIB_PATH:`
- `LIBPATH`

インストールの後に Web エージェント環境変数を設定するには、Web エージェントをインストールして設定した後で、以下のスクリプトを取得します。

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

ユーザ `.profile` ファイルまたは `envvars` ファイルのいずれかにスクリプトをリストできます。v6.x QMR 1 から Web エージェントにアップグレードする場合は、このスクリプトを取得する必要があります。

注: Oracle iPlanet Web サーバでは、このファイルは開始スクリプトに追加されているため、このスクリプトを実行する必要はありません。

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

apachectl スクリプトを使用して Apache サーバを実行する場合 (POSIX 上で Apache Web サーバを実行するときなど)、SiteMinder Web エージェント用の環境変数を設定する 1 行を apachectl スクリプトに追加します。

Apache サーバで apachectl スクリプトを使用する場合に Web エージェント変数を設定する方法

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. この行の前に、以下の行を追加します。

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

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: C:¥Program Files¥CA¥webagent
- UNIX インストールのデフォルトの場所: /opt/ca/webagent

インストール履歴ログ ファイル

インストーラは、次の情報を備えたログ ファイルを作成します。

- 製品名
- インストールされている場所
- 完全な (フル) バージョン番号

このファイルは以下の場所で作成されます。

Windows

```
C:¥Program Files¥CA¥install-info¥ca-install-history.log
```

UNIX

```
/opt/ca/install-info/ca-install-history.log
```

UNIX への Web エージェントの再インストール

Web エージェントを再インストールして見つからないアプリケーション ファイルをリストアすることができます。この手順については、既存の Web エージェントをアンインストールする必要はありません。インストール手順を繰り返すことによって、既存の Web エージェント ファイル上で再インストールを実行するだけです。

UNIX に Web エージェントを再インストールするには、以下の手順に従います。

1. バックアップとして Web エージェント構成設定のコピーを作成します。
2. GUI インストーラを使用して、UNIX システムに Web エージェントをインストールします。

再インストール中に、以下のいずれかによって、再インストールを確認する必要があります。

- [再インストール]ダイアログ ボックス(GUI モード)
- [アップグレード/再インストールの確認]プロンプト(コンソール モード)

UNIX 上でトラステッド ホストとしてシステムを登録する

トラステッド ホストは 1 つ以上の SiteMinder Web エージェントがインストールされているクライアント コンピュータです。トラステッドホストは、物理システムを指します。

トラステッド ホストとポリシー サーバの間の接続を確立するために、ホストをポリシー サーバと共に登録する必要があります。登録が完了したら、登録ツールは SmHost.conf ファイルを作成します。このファイルが正常に作成されたら、クライアント コンピュータはトラステッド ホストになります。

注: Web エージェントをシステムにインストールして設定するたびにホストを登録するのではなく、ホストを登録するのは一度だけです。

トラステッド ホストは、Web エージェントをインストールした直後、または後で登録することができます。ただし、ある時点で登録を実行する必要があります。

登録ツールは GUI またはコンソール モードから独立して実行できます。

GUI またはコンソール モードでのトラステッド ホストの登録

これらの手順は GUI およびコンソール モードを登録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。

- オプションに対応する数を入力してオプションを選択するように指示される場合があります。
- 各手順の後に、**Enter** キーを押してプロセスを続行します。プロンプトにより、順を追ってプロセスをガイドします。
- 入力するパスワードはすべてクリア テキストで表示されます。この問題の回避策として、インストールを GUI または無人モードで実行します。

ホストを登録する方法

1. 必要に応じて、以下のように設定ウィザードを開始します。

- a. コンソール ウィンドウを開きます。
- b. `web_agent_home/install_config_info` に移動します。
- c. 以下のいずれかのコマンドを入力します。

GUI モード: `./ca-wa-config.bin`

コンソール モード: `./ca-wa-config.bin -i console`

設定ウィザードが起動します。

2. [ホスト登録]ダイアログ ボックスで、

- a. 今すぐホストを登録する場合は[はい]を選択し、後でホストを登録する場合は[いいえ]を選択します。
- b. SiteMinder 環境で PKCS11 の暗号ハードウェアを使用している場合は、チェック ボックスをオンにします。
- c. [次へ]をクリックします。

3. 暗号ハードウェアを有効にした場合は、すべてのフィールドに入力します。有効にしなかった場合は、次の手順にスキップします。

- a. PKCS11 DLL フィールドで、PKCS11 DLL にフル パスを入力します。[選択]をクリックして DLL を検索します。
- b. オプションで、該当する場合、[トークンラベルとトークン パスフレーズ]でトークン ラベルを指定します。[トークン パスフレーズの確認]フィールドでパス フレーズを再確認し、[次へ]をクリックします。

4. [管理者の登録]ダイアログ ボックスで以下のフィールドを完了し、[次へ]をクリックします。

- [管理者ユーザ名] - ポリシー サーバでホストを登録することを許可された管理者の名前を入力します。

この管理者は、ポリシー サーバですでに定義されていて、トラステッドホストの登録設定の権限を所有している必要があります。デフォルトの管理者は **SiteMinder** です。

- [管理パスワード] - 管理者のパスワードを入力します。
- [管理者パスワードの確認] - パスワードを再入力します。
- [共有秘密キーのロールオーバーの有効化] - トラステッド ホストとポリシー サーバの間の通信を暗号化するために使用される共有秘密キーを定期的に変更する場合は、このチェック ボックスをオンにします。この機能が動作するには、キーのロールオーバーがポリシー サーバで有効である必要があります。

重要: 共有秘密キーのロールオーバーを有効にした場合、Web サーバプロセスを所有するユーザは **SmHost.conf** ファイルに書き込む許可が必要です。このユーザがこのファイルを変更できない場合、共有秘密キーのロールオーバーは更新できません。

たとえば、Oracle iPlanet および Apache Web サーバでは、ユーザ ディレクティブによって指定されたユーザは、**SmHost.conf** ファイルへの書き込み許可が必要です。**SmHost.conf** ファイルがユーザ 1 によって所有され、他のユーザが書き込み許可を持っていない場合、ユーザ 2 がサーバプロセスを所有する場合、共有される秘密ロールオーバーは **SmHost.conf** ファイルに書き込まれません。

5. [トラステッド ホスト名]および[設定オブジェクト]ダイアログ ボックスで、2 つのフィールドに値を入力し、[次へ]をクリックします。

- a. [トラステッド ホスト名]フィールドに、ポリシー サーバにトラステッド ホストを示す一意の名前を入力します。この名前は登録している物理クライアントシステムと同じである必要がありません。これは、**mytrustedhost** など、任意の固有の名前にすることができます。

注: この名前はトラステッド ホストの中で一意である必要があり、任意の **4.x Web** エージェントの名前と一致してはなりません。これは、**5.0 Web** エージェントと同じ名前にすることができますが、お勧めしません。

- b. [ホスト設定オブジェクト]フィールドで、ポリシー サーバに指定されたホスト設定オブジェクトの名前を入力し、[次へ]をクリックします。

このオブジェクトは、トラステッド ホストとポリシー サーバの間の接続を定義します。デフォルトを使用するには、**DefaultHostSettings** を入力します。ほとんどの場合は、独自のホスト設定オブジェクトを使用します。

注: 指定するエントリーは、ポリシー サーバに設定されたホスト設定オブジェクト エントリーと一致する必要があります。

6. [ポリシー サーバの IP アドレス]ダイアログ ボックスで、

- a. IP アドレスまたはホスト名、およびホストを登録しているポリシー サーバの認証ポートを入力します。デフォルトのポートは **44442** です。ポートを指定しない場合、デフォルトが使用されます。

デフォルト以外のポート番号を指定することもできますが、デフォルト以外のポートを使用していて、それを省略した場合、**SiteMinder** は以下のエラーを表示します。

登録に失敗しました (ipAddress[:port] が不正であるか、または認証サーバに接続できません (-1))

また、デフォルト以外のポートを指定した場合、そのポートがポリシー サーバの認証、認可、および監査ポートに使用されることに注意してください。ただし、統合サーバは、任意のポートの任意のエージェントリクエストに応答します。**SmHost.conf** ファイルのエントリーは以下のようになります。

```
polycyserver="ip_address,5555,5555,5555"
```

- b. [追加]をクリックします。

1 つ以上のポリシー サーバを追加できますが、ホスト登録については、リストの最初のサーバのみが使用されます。複数のエントリーを追加する場合は、カンマでそれらを区切ります。

複数のポリシー サーバが指定される場合、エージェントはそれらをブートストラップ サーバとして使用します。エージェントが開始されると、Web エージェントには接続してそのホスト設定オブジェクトを取得できるいくつかのポリシー サーバがあります。ホスト設定オブジェクトが取得された後、ブートストラップ ポリシー サーバはそのサーバ プロセスで使用されなくなります。ホスト設定オブジェクトには別のサーバのセットを含めることができます。それには、任意のブートストラップ サーバが含まれる場合も、含まれない場合もあります。

- c. [次へ]をクリックします。

7. FIPS 暗号化を使用する場合は、以下のオプションのうちの 1 つを選択します。

FIPS 互換モード(デフォルト)

非 FIPS モードを指定します。これにより、ポリシー サーバとエージェントは、既存の SiteMinder 暗号化アルゴリズムを使用して情報を読み取って書き込むことができます。組織で FIPS に準拠しているアルゴリズムの使用を必要としない場合は、ポリシー サーバとエージェントは、これ以上の設定なしに、非 FIPS モードで動作することができます。

FIPS 移行モード

FIPS 移行モードを指定します。これは、SiteMinder の以前のバージョンから FIPS のフル モードにアップグレードする場合に使用されます。ポリシー サーバとエージェントは、ユーザが FIPS 140-2 の承認済みアルゴリズムのみを使用する環境に移行しても、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。

FIPS のみのモード

FIPS フル モードを指定します。これは、ポリシー サーバとエージェントが FIPS 140-2 アルゴリズムのみを使用して、情報の読み取りと書き込みを行うことを要求されます。

重要: フル FIPS モードで実行している SiteMinder インストールでは、エージェント、エージェント API の以前のバージョンを使用するカスタムソフトウェア、および PM API またはポリシー サーバが公開するその他の API を使用するカスタムソフトウェアなどの、SiteMinder の以前のバージョンと相互操作できないか、または下位互換性がありません。これらすべてのソフトウェアを対応する SDK のバージョンと再度リンクし、フル FIPS モードに必要なサポートを獲得する必要があります。

FIPS 暗号化を使用しない場合は、デフォルト値を使用します。

8. [次へ]をクリックします。
9. ホスト設定ファイルの SmHost.conf のデフォルトの場所を受け入れるか、または[選択]をクリックして別の場所を選択します。[次へ]をクリックします。
デフォルト以外の場所を選択した後で、デフォルトのディレクトリに戻す場合は、[デフォルトのフォルダを復元]をクリックします。
ホストは登録されていて、ホスト設定ファイルの SmHost.conf は web_agent_home/config 内に作成されます。このファイルは変更できません。
10. Web エージェントを設定します。

SmHost.conf ファイルの変更 (UNIX)

Web エージェントおよびカスタム エージェントは、SmHost.conf ファイルの情報を使用してポリシー サーバを検索して初期接続を行うことによって、トラステッド ホストとして動作します。エージェントがポリシー サーバに接続したら、初期接続は閉じます。エージェントとポリシー サーバの間の通信は、ポリシー サーバにあるホスト設定オブジェクトの設定に基づきます。

SmHost.conf ファイルの一部を変更してエージェントからポリシー サーバへの初期接続を変更できます。

SmHost.conf ファイルを変更する方法

1. `web_agent_home/config` ディレクトリに移動します。
2. テキスト エディタで SmHost.conf ファイルを開きます。
3. 変更する以下のいずれかの設定に新しい値を入力します。

重要: ここにリストされたパラメータの設定のみが変更されます。
SmHost.conf ファイルのその他のパラメータの設定は変更されません。

hostconfigobject

トラステッド ホストとして動作しているエージェントとポリシー サーバの間の接続を定義するホスト設定オブジェクトを指定します。この名前は、管理 UI に定義された名前と一致する必要があります。

ホスト設定オブジェクトを変更する場合は、この設定を変更する必要があります。

例: `hostconfigobject="host_configuration_object"`

policyserver

トラステッド ホストが接続しようとするポリシー オブジェクトを指定します。正しい構文は次のとおりです。

`"IP_address, port, port, port"`

デフォルトポートは `44441,44442,44443` ですが、同じ数または 3 つのポートすべてに別の数を使用してデフォルト以外のポートを指定できます。統合サーバは、任意のポートの任意のエージェントリクエストに応答します。

エージェントに追加のブートストラップ サーバを指定するには、ファイルに複数のポリシー サーバエントリを追加します。複数のエントリは、接続してホスト設定オブジェクトを取得できるいくつかのポリシー サーバをエージェントに提供します。ホスト設定オブジェクトが取得されたら、ブートストラップ サーバはそのサーバプロセスに必要でなくなります。

複数のエントリはホスト登録中に、またはこのパラメータを変更することによって、追加できます。ポリシー サーバが SiteMinder 環境から削除されるか、使用されなくなった場合は、エントリを削除します。

重要: エージェントが多重プロセス Web サーバに設定されている場合は、プライマリ ポリシー サーバが失敗しても子プロセスがセカンダリポリシー サーバへの接続を確立できるように、複数のポリシー サーバエントリを指定することをお勧めします。ファイルに 1 つのポリシー サーバのみがリストされていて、そのポリシー サーバに接続できない場合は、新しい子プロセスが開始されるたびに Web エージェントを初期化できなくなります。

デフォルト: `IP_address, 44441,44442,44443`

例(単一エントリの構文): `"IP_address, port,port,port"`

例(複数エントリの構文、各ポリシー サーバを別々の行に配置する):

```
policyserver="123.122.1.1, 44441,44442,44443"  
policyserver="111.222.2.2, 44441,44442,44443"  
policyserver="321.123.1.1, 44441,44442,44443"
```

requesttimeout

トラステッド ホストとして動作するエージェントがポリシー サーバを利用できないと決定するまでに待機する間の秒の間隔を指定します。ポリシー サーバが過剰なトラフィックまたは遅いネットワーク接続が原因でビジーな場合は、タイムアウト値を増やすことができます。

デフォルト: 60

例: `requesttimeout="60"`

4. SmHost.Conf ファイルを保存して閉じます。

SmHost.Conf ファイルに対する変更が適用されます。

登録ツールを使用したトラステッド ホストの再登録 (UNIX)

初めてサーバに Web エージェントをインストールする場合、トラステッド ホストとしてそのサーバを登録するようにプロンプトが表示されます。トラステッド ホストが登録されたら、後続のエージェント インストールで再登録する必要はありません。以下のように、エージェントのインストールに関係なく、トラステッド ホストを再登録する必要がある場合がいくつかあります。

- SiteMinder 環境に変更が行われた場合に、トラステッド ホストの名前を変更するとき。
- トラステッド ホストが 管理 UI で削除された場合に、トラステッド ホストを登録するとき。
- トラステッド ホストのポリシー オブジェクトがポリシー ストアから削除された場合、またはポリシー ストアが失われた場合に、トラステッド ホストを登録するとき。
- トラステッド ホストとポリシー サーバの間の接続を保護する共有秘密キーを変更するとき。
- SmHost.conf 設定ファイルが喪失した場合に、それを再作成するとき。
- 既存のトラステッド ホストを最初に削除せずに上書きするとき。

登録ツール、smregghost はトラステッド ホストを再登録します。Web エージェントをインストールすると、このツールは `web_agent_home/bin` ディレクトリにインストールされます。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

登録ツールを使用してトラステッド ホストを再登録する方法

1. コマンド プロンプト ウィンドウを開きます。
2. ライブラリパス環境変数に Web エージェントの `bin` ディレクトリへのパスが含まれていることを確認します。
3. 以下の 2 つのコマンドを入力します。

```
library_path_variable=${library_path_variable}:web_agent_home/bin
export library_path_variable
```

たとえば、Solaris システムでは、以下の 2 つのコマンドを入力します。

```
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/usr/ca/webagent/bin
export LD_LIBRARY_PATH
```

以下のリストに、各オペレーティング システムの異なる変数を表示します。

Solaris

```
LD_LIBRARY_PATH
```

HP-UX

```
SHLIB_PATH:
```

Linux

```
LD_LIBRARY_PATH
```

AIX

```
LIBPATH
```

4. 以下の例に示すように、以下の必要な引数を使用して、`smreghost` コマンドを入力します。

```
smreghost -i policy_server_IP_address:[port]
-u administrator_username -p Administrator_password
-hn hostname_for_registration -hc host_configuration_object
```

注: スペースで各コマンドの引数と値を区切ります。スペースが含まれているすべての値を二重引用符(")で囲みます。以下の例を参照してください。

```
smreghost -i 123.123.1.1 -u SiteMinder -p mypw -hn "host computer A"
-hc DefaultHostSettings
```

-o 引数の例:

```
smreghost -i 123.123.1.1 -u SiteMinder -p mypw -hn "host computer A"
-hc DefaultHostSettings -o
```

以下の引き数は `smreghost` コマンドと共に使用されます。

`-i policy_server_IP_address:port`

このホストを登録するポリシー サーバの IP アドレスを示します。デフォルトポートを使用していない場合のみ、認証サーバのポートを指定します。

デフォルト以外のポートにすることができるポート番号を指定した場合、そのポートは 3 つのポリシー サーバのサーバ(認証、認可、説明)に使用されますが、統合サーバは任意のポートの任意のエージェントリクエストに応答します。たとえば、ポート 55555 を指定した場合、`SmHost.conf` ファイルのポリシー サーバ エントリは以下を表示します。

"policy_server_ip_address,5555,5555,5555"

例: (IPv4) 127.0.0.1,55555

例: (IPv6) [2001:DB8::/32][:55555]

-u *administrator_username*

トラステッド ホストを登録する権限を持つ SiteMinder 管理者の名前を指定します。

-p *Administrator_password*

トラステッド ホストを登録することを許可された管理者のパスワードを示します。

-hn *hostname_for_registration*

登録されるホストの名前を示します。これはホストを識別する任意の名前にすることができますが、一意である必要があります。登録の後、この名前は 管理 UI の[トラステッド ホスト]リストに配置されます。

-hc *host_config_object*

ポリシー サーバに設定されたホスト設定オブジェクトの名前を示します。トラステッド ホストを登録できるようにする前に、このオブジェクトがポリシー サーバに存在する必要があります。

-sh *shared_secret*

Web エージェント用の共有秘密キーを指定します。これは、ローカル Web サーバの **SmHost.conf** ファイルに格納されています。この引数は、ローカル Web サーバ上でのみ共有秘密キーを変更します。ポリシー サーバは接続されていません。

-rs

共有秘密キーがポリシー サーバによって自動的に更新(ロールオーバー)されるかどうかを示します。この引数は、共有秘密キーを更新するようにポリシー サーバに指示します。

-f *path_to_host_config_file*

(オプション) 登録データが含まれているファイルへのフルパスを示します。デフォルトのファイルは、**SmHost.conf** です。パスを指定しない場合、ファイルは **smreghost** ツールを実行している場所にインストールされます。

既存のホスト設定ファイルと同じ名前を使用する場合、ツールはオリジナル ファイルをバックアップし、バックアップ ファイル名に **.bk** 拡張子を追加します。

- cf FIPS モード

以下のいずれかの FIPS モードを選択してください。

- 互換 - 非 FIPS モードを指定します。これにより、ポリシー サーバとエージェントは、既存の SiteMinder 暗号化アルゴリズムを使用して情報を読み取って書き込むことができます。組織で FIPS に準拠しているアルゴリズムの使用を必要としない場合は、ポリシー サーバとエージェントは、これ以上の設定なしに、非 FIPS モードで動作することができます。
- 移行 - FIPS 移行モードを指定します。これは、SiteMinder の以前のバージョンから FIPS のフル モードにアップグレードする場合に使用されます。ポリシー サーバとエージェントは、ユーザが FIPS 140-2 の承認済みアルゴリズムのみを使用する環境に移行しても、既存の SiteMinder 暗号化アルゴリズムを引き続き使用します。
- 専用 - FIPS フル モードを指定します。これは、ポリシー サーバとエージェントが FIPS 140-2 アルゴリズムのみを使用して、情報の読み取りと書き込みを行うことを要求されます。

重要: フル FIPS モードで実行している SiteMinder インストールでは、エージェント、エージェント API の以前のバージョンを使用するカスタムソフトウェア、および PM API またはポリシー サーバが公開するその他の API を使用するカスタムソフトウェアなどの、SiteMinder の以前のバージョンと相互操作できないか、または下位互換性がありません。これらすべてのソフトウェアを対応する SDK のバージョンと再度リンクし、フル FIPS モードに必要なサポートを獲得する必要があります。

このスイッチが使用されなかったり、ユーザがモードを指定せずにスイッチを使用する場合、デフォルト設定が使用されます。

デフォルト: 互換

注: 使用されている FIPS 認定モジュールおよびアルゴリズム、保護されているデータおよび、SiteMinder 暗号境界の詳細については、「ポリシー サーバ管理ガイド」を参照してください。

-o

既存のトラステッド ホストに上書きします。この引数を使用しない場合は、smreghost コマンドを使用する前に 管理 UI を備えた既存のトラステッド ホストを削除する必要があります。この引き数で smreghost コマンドを使用することを推奨します。

トラステッド ホストは再登録されます。

1つのシステムに複数のトラステッドホストを登録する (UNIX)

Web サーバとエージェントがインストールされている各マシンに対し、一般的に登録するトラステッドホストは1つのみです。ただし、1台のコンピュータに複数のトラステッドホストを登録して各 SiteMinder クライアントに別々の接続を作成することができます。複数のトラステッドホストを使用すると、ポリシーサーバとのコミュニケーションを必要とする各クライアントに対して一意の共有秘密キーおよび安全な接続が確保されます。

ほとんどのインストールでは、これは推奨された設定ではありません。しかしながら、これは、SiteMinder エージェントによって保護された各クライアントまたはクライアントのグループアプリケーションに別々のセキュアなチャンネルを必要とするサイトのオプションです。たとえば、アプリケーション サービスプロバイダには異なるアプリケーションがインストールされた多数のクライアントコンピュータがあるとします。それぞれのアプリケーションに安全な接続を望む場合は、複数のトラステッドホストを登録すると、実現することができます。ポリシーサーバはその後、各クライアント接続に一意の共有秘密キーを発行します。

複数のトラステッドホストを登録するには、以下の方法のうちの1つを使用します。

- 設定ウィザードでの登録: トラステッドホストとして追加のサーバを登録するには、登録処理を再度実行します。ただし、SmHost.conf ファイルの場所を指定するようにプロンプトが表示された場合は、一意のパスを入力します。新しいホストを登録する場合は、既存の Web サーバの SmHost.conf ファイルを使用しないでください。使用すると、そのファイルが上書きされます。SmHost.conf という名前を使用するか、またはファイルに新しい名前を付けることができます。

注: ポリシーサーバでトラステッドホストを登録していて、設定ウィザードを実行して SmHost.conf ファイルで一意のパスを使用せずに次のエージェントを設定した場合は、[ホスト登録]ダイアログボックスに警告メッセージが表示されます。このメッセージは、以下のように表示されます。

警告: このエージェントはポリシーサーバを使用してすでに登録されています。

- smregghost コマンドラインツールでの登録: 指定された任意のコンピュータで最初のエージェントインストールを完了した後で、smregghost ツールを実行します。このツールは登録する各トラステッドホストに対して実行できます。

第 4 章: Web エージェントの r12.0 SP3 へのアップグレード

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 エージェントのアップグレードの準備をする方法 \(P. 75\)](#)

[Windows システムでの Web エージェントの r12.0 SP3 へのアップグレード \(P. 78\)](#)

[UNIX システムでの Web エージェントの r12.0 SP3 へのアップグレード \(P. 80\)](#)

Web エージェントのアップグレードの準備をする方法

以下のプロセスを使用して、Web エージェントをアップグレードする準備を行うことができます。

1. 「SiteMinder アップグレード ガイド」でアップグレード処理を確認します。
2. Web サーバ上の任意のカスタマイズされたファイルをバックアップします。
3. アップグレード中に発生するパスワード サービスとフォーム テンプレートの変更を確認します。
4. アップグレードの後に Web エージェント設定ウィザードを実行する場合、発生するさまざまな Web エージェント設定ファイルへの変更を確認します。
5. LD_PRELOAD 変数を設定して既存の Web エージェントとの競合を回避させます。
6. アップグレード中に既存の読み取り専用ファイルを置換します (プロンプトが表示された場合)。

アップグレード手順の確認

Web エージェントをアップグレードする前に、「SiteMinder アップグレード ガイド」のアップグレードプロセスを確認してください。このガイドでは、重要な概要情報と Web エージェントをアップグレードする前に完了する必要がある重要なタスクについて説明します。

注: 最後に Web エージェントをインストールしてから、Web サーバ自体をアップグレードしている場合は、エージェントのアップグレードが機能しない場合があります。Web サーバのバージョンが最後の Web エージェントインストール以降同じままである場合のみ、アップグレードが確保されます。

カスタマイズされたファイルのバックアップ

カスタマイズされたファイルは、アップグレードによって上書きされる場合があります。アップグレードする前に、エージェント設定ファイルやホスト設定ファイルなどの設定済みファイルをバックアップします。

アップグレードの間のパスワード サービスとフォーム テンプレートの変更

パスワード サービスとフォーム テンプレートでは、`jpw_default`、`pw_default` および `samples_default` ディレクトリがアップグレードされます。ただし、カスタマイズされたファイルが含まれている可能性のあるこれらのディレクトリ (`jpw`、`pw` およびサンプル) のデフォルト以外のバージョンは、変更されません。

アップグレードの後に設定ウィザードを実行した場合の結果

Web エージェントをアップグレードした後に Web エージェント設定ウィザードを実行すると、以下が発生します。

- SiteMinder は、現在の Web エージェント設定ファイル (`WebAgent.conf`) のコピーを保存します。
- SiteMinder は、`IgnoreExt` 行および `BadURLCharacters` 行を新しい `WebAgent.conf` ファイルにコメント行として移動することによって、容易にカスタム エlementを追加できるようにします。

注: SiteMinder は、トラステッド ホスト設定ファイル (`SmHost.conf`) のコピーを保存しません。

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。

既存の読み取り専用ファイルの置換

Web エージェントをアップグレードする場合、読み取り専用ファイルを置換するかどうかを尋ねるメッセージが表示される場合があります。[すべてはい]を選択します。

Windows システムでの Web エージェントの r12.0 SP3 へのアップグレード

Web サーバのバージョンが Web エージェントの最後のインストール以降変更されていない場合は、SiteMinder メディアの実行可能ファイルは既存の SiteMinder Web エージェントを r12.0 SP3 にアップグレードします。

最後に Web エージェントをインストールしてから、Web サーバ自体をアップグレードしている場合は、エージェントのアップグレードが機能しない場合があります。Web サーバのバージョンが最後の Web エージェント インストール以降同じままである場合のみ、アップグレードが確保されます。ただし、hotfix を適用している場合は、アップグレードできます。

重要: r12.0 SP3 にアップグレードする前に任意の 5.x Web エージェント オプション パックを削除します。6.x Web エージェント オプション パックは、r12.0 SP3 にアップグレードする前に削除する必要はありません。Web エージェント オプション パックの削除および再インストールの詳細については、「SiteMinder Web Agent Option Pack Guide」を参照してください。

以下の点について考慮してください。

- インストール プログラムがロックされたエージェント ファイルを検出した場合、それを再設定する代わりにシステムを再起動するようにプロンプトが表示されます。システムをすぐにまたは後で再起動するかどうか選択します。
- Oracle iPlanet Web サーバにエージェントをインストールする場合、httpd.exe サービスが smconapi.dll を見つけることができないと述べるエラーメッセージが表示される場合があります。このメッセージが表示される場合は、Web エージェント設定ウィザードを起動する前にシステムを再起動します。

Windows で Web エージェントをアップグレードする方法

1. 実行しているすべてのアプリケーションを終了して Web サーバを停止します。
2. [テクニカル サポート](#) からインストール プログラムをダウンロードします。

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

3. win32 フォルダに移動し、ca-wa-version_number-win32.exe ファイルをダブルクリックします。

インストールウィザードが起動します。

4. [概要]ダイアログ ボックスで、情報を読んで[次へ]をクリックします。
5. 使用許諾契約の内容を読みます。ラジオ ボタンをクリックして使用許諾契約の条件を受け入れて、[次へ]をクリックします。
6. [重要な情報]ダイアログ ボックスの注意事項を読んで、[次へ]をクリックします。
7. [ショートカットフォルダの選択]ダイアログ ボックスでエージェント設定ウィザードの場所を選択し、[次へ]をクリックします。

すべてのユーザにショートカット経由で設定ウィザードへのアクセスを許可するには、[すべてのユーザにアイコンを作成]チェック ボックスをオンにします。それ以外の場合は、チェック ボックスをオフにします。

アップグレードプログラムは既存の Web エージェントを検索して、[アップグレードの確認]ダイアログ ボックスを表示します。

8. [アップグレードの確認]ダイアログ ボックスで、以下のオプションのうちの 1 つを選択し、[次へ]をクリックします。
 - アップグレードを続行する - Web エージェントを r12.0 SP3 にアップグレードします。
 - [アップグレードを中止する] - Web エージェントをアップグレードせずにアップグレード手順を終了します。
9. インストール前のサマリダイアログ ボックスで、インストール設定が正しいことを確認し、[インストール]をクリックします。

新しい Web エージェントファイルは指定された場所にコピーされます。

注: インストール プログラムは、特定のシステム .dlls の新しいバージョンがシステムにインストールされていることを検出する場合があります。これらの新しいファイルを古いファイルで上書きするようにプロンプトが表示された場合は、[すべていいえ]をクリックします。

10. [インストール完了]ダイアログ ボックスで、システムをすぐにまたは後で再起動するかどうか選択します。次に、[終了]をクリックします。
11. Web エージェント設定ウィザードでアップグレードした Web エージェントを再設定します。

注: トラストド ホストを再登録する必要はありません。

UNIX システムでの Web エージェントの r12.0 SP3 へのアップグレード

Web サーバのバージョンが Web エージェントの最後のインストール以降変更されていない場合は、SiteMinder メディアの実行可能ファイルは既存の SiteMinder Web エージェントを r12.0 SP3 にアップグレードします。

最後に Web エージェントをインストールしてから、Web サーバ自体をアップグレードしている場合は、エージェントのアップグレードが機能しない場合があります。Web サーバのバージョンが最後の Web エージェント インストール以降同じままである場合のみ、アップグレードが確保されます。ただし、hotfix を適用している場合は、アップグレードできます。

重要: r12.0 SP3 にアップグレードする前に任意の 5.x Web エージェント オプション パックを削除します。6.x Web エージェント オプション パックは、r12.0 SP3 にアップグレードする前に削除する必要はありません。Web エージェント オプション パックの削除および再インストールの詳細については、「SiteMinder Web Agent Option Pack Guide」を参照してください。

UNIX システムで Web エージェントをアップグレードする方法

注: 既存の r12 Web エージェントを r12 SP1 にアップグレードする場合は、ルートユーザとしてログインする必要があります。新しい r12 SP1 Web エージェントをインストールする場合は、ルート権限は必要ではありません。

1. 実行しているすべてのアプリケーションを終了して Web サーバを停止します。
2. [テクニカル サポート](#) からインストール プログラムをダウンロードします。
3. 使用しているオペレーティング システムでの適切なディレクトリに移動します。
4. 該当するバイナリ ファイルをローカル ディレクトリにコピーして、そのディレクトリに移動します。ファイル名は以下の規則を使用します。

`ca-wa-version-operating_system.bin`

5. 権限に応じて、`chmod` コマンドを実行することにより、インストール ファイルに実行可能な権限を追加する必要がある場合があります。例:

```
chmod +x ca-wa-version-operating_system.bin
```

6. コンソール ウィンドウを開いて、インストール プログラムの場所から以下を入力します。

`./ca-wa-version-operating_system.bin`

7. [概要]ダイアログ ボックスで、情報を読んで[次へ]をクリックします。
8. 使用許諾契約を読んで、許諾契約に同意する場合はラジオ ボタンをクリックします。[次へ]をクリックします。
9. [重要な情報]ダイアログ ボックスの注意事項を読んで、[次へ]をクリックします。
[アップグレードの確認]ダイアログ ボックスが表示されます。
10. [アップグレードの確認]ダイアログ ボックスで、以下のうちの 1 つを選択し、[次へ]をクリックします。
 - アップグレードを続行する - Web エージェントを r12.0 SP3 にアップグレードします。
 - インストールを中止する - Web エージェントをアップグレードせずにアップグレード手順を終了します。
11. インストール前のサマリダイアログ ボックスで、インストール設定が正しいことを確認し、[インストール]をクリックします。
新しい Web エージェントファイルは指定された場所にコピーされます。
12. [インストール完了]ダイアログ ボックスで、[終了]をクリックします。
Web エージェントのアップグレードが完了します。アップグレードされた 5.x Web エージェントのシステムが事前にトラステッド ホストとして登録されていない場合は、ある時点でシステムで登録する必要があります。

第 5 章: IIS Web エージェントの設定

このセクションには、以下のトピックが含まれています。

[IIS 7.5 に SiteMinder Web エージェントを設定する方法 \(P. 83\)](#)

[IIS 7.0 に SiteMinder Web エージェントを設定する方法 \(P. 97\)](#)

[IIS 6.0 に SiteMinder Web エージェントを設定する方法 \(P. 107\)](#)

[SiteMinder Web エージェントを設定して Microsoft Outlook Web Access を保護する方法 \(P. 117\)](#)

IIS 7.5 に SiteMinder Web エージェントを設定する方法

IIS 7.5 Web サーバ用の SiteMinder Web エージェントを設定するには、以下の手順に従います。

1. 以下の前提条件を確認します。
 - SiteMinder Web エージェントがインストールされていること。
 - SiteMinder Web エージェントを実行するコンピュータがインストール後に (SiteMinder Web エージェント設定ウィザードを実行することなく) 再起動されたこと。
 - Web サーバ (IIS) 役割が Web サーバに追加されていること。
2. IIS 7.5 Web サーバに役割サービスを追加します。
3. SiteMinder のクラシック モード アプリケーション プールを設定します。
4. SiteMinder で保護するアプリケーションをクラシック モード アプリケーション プールへ移動させます。
5. Web エージェント設定ウィザードを実行します。
6. SiteMinder で保護する IIS 7.5 Web サーバ上のすべての Web サイトに ISAPI フィルタを追加します。
7. SiteMinder で保護する IIS 7.5 Web サーバ上のすべての Web サイトにハンドラ マッピングを追加します。
8. 以下のディレクトリにクラシック モード アプリケーション プールの ID 許可を付与します。
 - `web_agent_home¥config`
 - `web_agent_home¥log`

9. SiteMinder Windows 認証方式を使用している場合は、仮想ディレクトリを作成して設定します。

詳細情報

[IIS Web エージェントで必要とされるエージェント設定パラメータ \(P. 27\)](#)

IIS 7.5 Web Server への役割サービスの追加

IIS 7.5 Web サーバ上で SiteMinder Web エージェントを操作する前に、SiteMinder Web エージェントによって必要とされる役割サービスを利用するために Web サーバを設定する必要があります。

IIS 7.5 Web Server に役割サービスを追加する方法

1. Windows Server 2008 システムで、[スタート]-[管理ツール]-[サーバー マネージャ]をクリックします。

注: [ユーザ アカウント コントロール]ダイアログ ボックスが表示されたら、[続行]をクリックします。
[サーバー マネージャ]が開きます。
2. [役割]を展開します。

インストールされた役割のリストが表示されます。
3. [Web サーバー (IIS)]をクリックします。

IIS 概要情報が表示されます。
4. [役割の概要]セクションで、[役割の追加]をクリックします。

役割の追加ウィザードが開始されます。
5. ウィザードを使用して以下の役割サービスを IIS 7.5 Web サーバに追加します。
 - ASP.NET
 - CGI
 - ISAPI 機能拡張
 - ISAPI フィルタ
 - IIS 管理コンソール
 - Windows 認証 (SiteMinder Windows 認証方式の場合)役割サービスが IIS Web サーバに追加されます。

Windows 認証役割サービスの IIS 7.x Web サーバへの追加

SiteMinder Windows 認証方式を使用するには、Windows 認証役割サービスを IIS 7.x Web サーバに追加します。SiteMinder を IIS 7.x Web サーバに追加する場合は、他にも追加する役割サービスがあります。すでに IIS 7.x Web サーバ上で SiteMinder を使用しており、SiteMinder Windows 認証方式を追加するだけの場合は、以下の手順に従います。

Windows 認証役割サービスの IIS 7.x Web サーバへの追加

1. Windows Server 2008 システムで、[スタート]-[管理ツール]-[サーバー マネージャ]をクリックします。

注: [ユーザ アカウント コントロール]ダイアログ ボックスが表示されたら、[続行]をクリックします。

[サーバー マネージャ]が開きます。
2. [役割の概要]セクションで[役割]を見つけ、リストし、[Web サーバー(IIS)]リンクをクリックします。

Web サーバ(IIS) ページが表示されます。
3. [役割サービスの追加]をクリックします。

[役割サービスの追加]ウィザードが開始します。
4. ウィザードを使用して以下の役割サービスをお使いの IIS 7.x Web サーバに追加します。
 - Windows 認証

認証役割サービスがお使いの IIS 7.x Web サーバに追加されました。

詳細情報:

[IIS 7.x Web サーバへの役割サービスの追加](#) (P. 98)

SiteMinder Web エージェント用のクラシック モード アプリケーション プールの設定

IIS 7.x Web サーバ上のアプリケーション プールは以下のいずれかのモードを使用します。

- 統合モード(デフォルト)
- クラシック モード

IIS 7.x Web サーバでは、SiteMinder Web エージェントはクラシック モードで設定されたアプリケーション プール上でのみ作動します。IIS 7.x Web サーバのデフォルト アプリケーション プールは統合モードを使用します。IIS 7.5 Web サーバ上で SiteMinder Web Agent を使用するには、以下のいずれかを実行します。

- IIS 7.x サーバ上で統合モードで実行する必要があるアプリケーションがない場合は、デフォルトのアプリケーション プール モード (DefaultAppPool) をクラシック モードに切り替えます。
- アプリケーション プールを統合モードで使用する他のアプリケーションと同じ IIS サーバ上で SiteMinder Web エージェントを実行する場合は、以下の手順で、SiteMinder 用にクラシック モードのアプリケーション プールを新規作成します。

SiteMinder Web エージェント用にクラシック モード アプリケーション プールを設定する方法

1. IIS マネージャを開きます。
IIS マネージャの開始ページが表示されます。
2. Web サーバを展開します。
[アプリケーション プール]アイコンおよび[Web サイト]フォルダが表示されます。
3. [アプリケーション プール]アイコンを右クリックし、[アプリケーション プールの追加]を選択します。
[アプリケーション プールの追加]ダイアログ ボックスが表示されます。カーソルは[名前]フィールドにあります。
4. 新しいアプリケーション プールの名前を入力します。認識するのが簡単な「SiteMinder」のような名前を使用することをお勧めします。
5. [マネージ パイプライン モード]ドロップダウンリストをクリックし、「クラシック」を選択します。
6. [OK]をクリックします。

Web エージェント用にクラシック モード アプリケーション プールが作成されました。

SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動

IIS 7.x Web サーバでクラシック モード アプリケーション プールを作成した後、アプリケーションを現在のアプリケーション プールから SiteMinder によって使用されるクラシック モード アプリケーション プールに移動させる必要があります。

SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動方法

1. IIS マネージャを開きます。
2. [接続] ペインで、Web サーバを展開します。
[アプリケーション プール] アイコンおよび[サイト] フォルダが表示されます。
3. アプリケーション プールをクリックします。
アプリケーション プールのリストが表示されます。
4. 移動させるアプリケーションが含まれるアプリケーション プールを右クリックして、[アプリケーションの表示]を選択します。たとえば、SiteMinder で保護するアプリケーションが現在デフォルト アプリケーション プール内にある場合は、デフォルト アプリケーション プールを右クリックします。
アプリケーションのリストが表示されます。
5. 移動させるアプリケーションを右クリックし、[アプリケーション プールの変更]を選択します。
[アプリケーション プールの選択] ダイアログ ボックスが表示されます。
6. アプリケーション プール ドロップダウン リストをクリックし、SiteMinder 用に作成したクラシック モード アプリケーション プールを選択します。
7. [OK] をクリックします。
[アプリケーション プールの選択] ダイアログ ボックスが閉じます。アプリケーションは新しいアプリケーション プールに移動しました。

IIS Web エージェントの設定ウィザードの実行

エージェントを設定する前に、エージェントがインストールされているシステムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

IIS Web エージェントを設定する方法

1. 必要に応じて、**Web** エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

注: インストールの直後に **Web** エージェントを設定することを選択した場合、SiteMinder はウィザードを自動的に開始します。

2. トラステッド ホストが登録されている場合は、次の手順にスキップします。そうでない場合は、[ホスト登録]ダイアログ ボックスで[いいえ]を選択して登録をスキップし、[次へ]をクリックします。
3. **Web** エージェントと共に設定する **Web** サーバインスタンスを選択します。
すでにサーバを **Web** エージェントと共に設定しており、設定ウィザードを実行して追加の **Web** サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した **Web** サーバのリストを示します。
 - a. 次のいずれかをオンにします。
上書き - サーバ インスタンスの設定を上書きします。
保存 - **Web** サーバの設定を保存します。
重要: 以前に設定したサーバのチェック ボックスをオフにすると、**Web** エージェントはこのサーバから削除されます。
 - b. [次へ]をクリックします。
4. [エージェント設定オブジェクト]フィールドで、この **Web** サーバ インスタンスのエージェント設定オブジェクトの名前を入力し、[次へ]をクリックします。

この名前は、ポリシー サーバですすでに定義されているエージェント設定オブジェクトに一致する必要があります。たとえば、デフォルトを使用する場合は `IISDefaultSettings` を入力します。

5. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェント ファイルがインストールされます。

6. インストールが完了したら、[終了]をクリックします。

注: エージェントが設定されたら、エージェントとトレースメッセージが適切にログ記録されるように、マシンを再起動する必要があります。

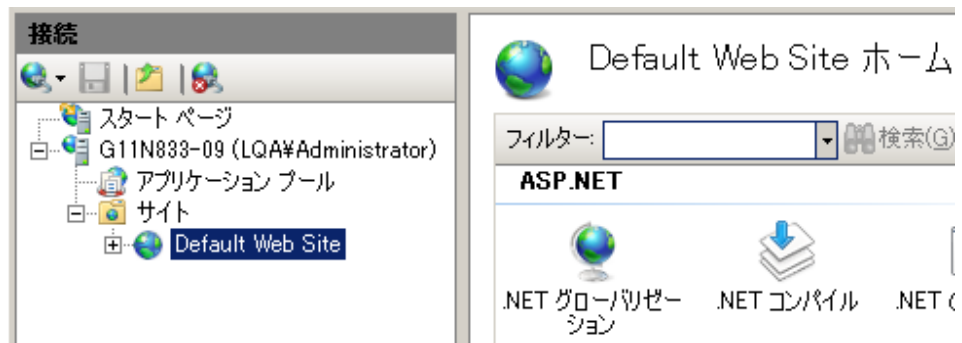
詳細情報

[Windows システムに Web エージェントをインストールする \(P. 31\)](#)

SiteMinder で保護する IIS 7.5 Web サイトへのエージェント ISAPI フィルタの追加

IIS 7.5 上で SiteMinder Web エージェントを実行するには、保護する各 Web サイトの最上位レベルのフォルダに SiteMinder ISAPI フィルタを追加します。このフィルタは Web エージェント ISAPI スクリプトおよび他のファイルを実行します。

以下の図では、「デフォルト Web サイト」という名前の最上位レベルフォルダが表示されています。これに対して手動で ISAPI フィルタを追加する必要があります。



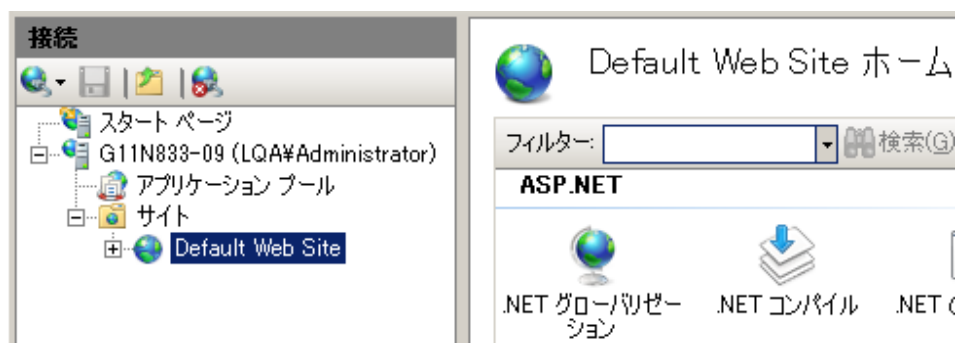
SiteMinder で保護する IIS 7.5 Web サイトにエージェント ISAPI フィルタを追加する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開きます。
注: [ユーザ アカウント コントロール] ダイアログ ボックスが表示されたら、[続行] をクリックします。
2. [接続] ペインで、Web サーバを展開します。
[サイト] フォルダが表示されます。
3. [サイト] フォルダを展開し、次に、SiteMinder で保護する追加の Web サイトのアイコンをクリックします。
4. IIS セクションの下で、[ISAPI フィルタ] をダブルクリックします。
[ISAPI フィルタ] 画面が表示されます。
5. アクション ペインの下で、[追加] をクリックします。
[ISAPI フィルタの追加] ダイアログ ボックスが表示されます。
6. フィルタの名前を入力します。認識するのが簡単な「SiteMinder ISAPI Filter」のような名前を使用することをお勧めします。
7. 省略記号ボタン ([実行可能ファイル] フィールドの右) をクリックします。
[開く] ダイアログ ボックスが表示されます。
8. 以下のファイルに移動します。
`web_agent_home\bin\ISAPI6WebAgent.DLL`
9. [開く] をクリックします。
ISAPI6WebAgentDLL.dll ファイルは [ISAPI フィルタの追加] ダイアログ ボックスに表示されます。
10. [OK] をクリックします。
[ISAPI フィルタの追加] ダイアログ ボックスが閉じて、SiteMinder ISAPI フィルタがリストに表示されます。
11. SiteMinder で保護する他の Web サイトについても、手順 3 から 10 を繰り返します。
エージェント ISAPI フィルタが追加されました。

SiteMinder で保護する IIS 7.5 Web サイトへのハンドラ マッピングの追加

IIS 7.5 上で SiteMinder Web エージェントを実行するには、保護する各 IIS 7.5 Web サイトの最上位レベルのフォルダに SiteMinder ハンドラ マッピングを追加します。

以下の図では、「デフォルト Web サイト」という名前の最上位レベル フォルダが表示されています。これに対して手動でハンドラ マッピングを追加する必要があります。



SiteMinder で保護する IIS 7.5 Web サイトにハンドラ マッピングを追加する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開きます。
注: [ユーザ アカウント コントロール] ダイアログ ボックスが表示されたら、[続行] をクリックします。
2. [接続] ペインで、**Web** サーバを展開します。
[サイト] フォルダが表示されます。
3. [サイト] フォルダを展開し、次に、SiteMinder で保護する Web サイトのアイコンをクリックします。
4. IIS セクションの下で、[ハンドラ マッピング] アイコンをダブルクリックします。
インストールされたハンドラ マッピングのリストが表示されます。
5. [無効] リストで、以下のハンドラ マッピングをクリックします。
ISAPI-d11
6. [アクション] ペインで、[機能のアクセス許可の編集] をクリックします...
[機能のアクセス許可の編集] ダイアログ ボックスが表示されます。
7. 以下のチェック ボックスをオンにします。
 - 読み取り
 - スクリプト
 - 実行
8. [OK] をクリックします。
[機能のアクセス許可の編集] ダイアログ ボックスが閉じます。
9. [アクション] ペインで、[ワイルドカード スクリプト マップの追加] をクリックします。
[ワイルドカード スクリプト マップの追加] ダイアログ ボックスが表示されます。
10. 省略記号ボタン ([実行可能ファイル] フィールドの右) をクリックします。
[開く] ダイアログ ボックスが表示されます。
11. 以下のファイルに移動します。
`web_agent_home¥bin¥ISAPI6WebAgent.DLL`
注: `web_agent_home` 変数のデフォルト値は `C:¥Program Files¥CA¥webagent` です。
12. [開く] をクリックします。

[ワイルドカード スクリプト マップの追加]ダイアログ ボックスに ISAPI6WebAgentDLL.dll ファイルが表示されます。

13. [名前]フィールドに、マッピングの名前を入力します。認識するのが簡単な「handler-wa」のような名前を使用することをお勧めします。

14. [OK]をクリックします。

確認のダイアログが表示されます。

15. [はい]をクリックします。

[ワイルドカード スクリプト マップの追加]ダイアログ ボックスが閉じて、マッピングはリストに表示されます。ハンドラ マッピングは保護される Web サイトに追加されます。

16. SiteMinder で保護する IIS 7.5 Web サーバ上の他の全 Web サイトについて、手順 3 から 15 を繰り返します。

ハンドラ マッピングが追加されます。

アプリケーション プール識別情報に対する SiteMinder SmHost.conf ファイルおよびログ ディレクトリの許可

IIS 7.5 Web サーバ上のすべてのアプリケーションプール識別情報には、IIS Web サーバをホストしているコンピュータ上で以下の SiteMinder アイテムの許可が必要です。

- SmHost.conf ファイル
- /log ディレクトリ

アプリケーション プールに対して SiteMinder SmHost.conf ファイルおよびログ ディレクトリの許可を付与する方法

1. 以下のファイルに移動します (開かないでください)。

`web_agent_home\config\SmHost.conf`

2. ファイルを右クリックして、[プロパティ]を選択します。

SmHost.conf の [プロパティ] ウィンドウが表示されます。

3. [セキュリティ] タブをクリックします。
4. [グループ] または [ユーザ名] ペインで **SYSTEM** が選択されていることを確認し、次に、[編集] をクリックします。

注: [ユーザ アカウント コントロール] ダイアログ ボックスが表示されたら、[続行] をクリックします。

SmHost.conf の [許可] ダイアログ ボックスが表示されます。

5. [追加] をクリックします。

[ユーザー、コンピュータ、またはグループの選択] ダイアログ ボックスが表示されます。

6. 以下の手順を実行します。

- a. [場所] をクリックします。

[場所] ダイアログ ボックスが表示されます。

- b. お使いのコンピュータ名 (リストの最上位) を選択し、[OK] をクリックします。

[場所] ダイアログ ボックスが閉じ、コンピュータ名が [場所の指定] フィールドに表示されます。

- c. [選択するオブジェクト名の入力] フィールドに、以下の形式でアプリケーション プール名を入力します。

`IIS AppPool¥Application_Pool_Name`

たとえば、デフォルト アプリケーション プールを追加するには、以下のように入力します。

IIS AppPool¥DefaultAppPool

- d. [名前の確認]をクリックし、[OK]をクリックします。

[ユーザー、コンピュータ、またはグループの選択]ダイアログ ボックスが閉じます。選択したアプリケーション プールと共に SmHost.conf の許可が表示されます。

7. 許可リストで、以下のチェック ボックスをオンにします。

- 読み取り
- 読み取りと実行
- 書き込み

8. [OK]をクリックします。

SmHost.conf の[許可]ダイアログ ボックスが閉じます。

9. [OK]をクリックします。

SmHost.conf の[プロパティ]ウィンドウが閉じます。

10. 以下のディレクトリに移動します(開かないでください)。

web_agent_home¥log

11. ディレクトリを右クリックして、[プロパティ]を選択します。

注: [ユーザ アカウント コントロール]ダイアログ ボックスが表示されたら、[続行]をクリックします。

12. 手順 3 ~ 9 を繰り返します。

アプリケーション プール識別情報に対して SiteMinder SmHost.conf ファイルおよびログ ディレクトリの許可が付与されました。

Windows 認証方式用の仮想ディレクトリの作成と設定 (IIS 7.5)

<stmdnr>Windows 認証方式を使用するには、IIS 7.5 Web サーバ上で仮想ディレクトリを設定します。仮想ディレクトリには、認証情報に関して NT チャレンジおよびレスポンスが必要です。

Windows 認証方式用の仮想ディレクトリを作成および設定する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開きます。
2. 左側のペインで、以下のアイテムを展開します。

- Web サーバ アイコン
- サイトフォルダ
- 既定の Web サイト アイコン

3. `siteminderagent` 仮想ディレクトリを右クリックし、次に、[仮想ディレクトリの追加]を選択します。

[仮想ディレクトリの追加]ダイアログ ボックスが表示されます。

4. [エイリアス]フィールドに、以下を入力します。

`ntlm`

5. [参照]ボタン([物理パス]フィールドの横)をクリックし、次に、以下のディレクトリを見つけます。

`web_agent_home\samples`

仮想ディレクトリが作成されます。

6. 以下のいずれかの手順で仮想ディレクトリを設定します。

- SiteMinder Windows 認証方式で Web サイト全体のリソースをすべて保護するには、[既定の Web サイト]アイコンをクリックします。
- SiteMinder Windows 認証方式で Web サイト全体を保護しない場合は、`ntlm` 仮想ディレクトリ(手順 4 で作成済み)をクリックします。

7. [認証]アイコンをダブルクリックします。

[認証]ダイアログ ボックスが表示されます。

8. 以下の手順を実行します。

- a. [匿名認証]を右クリックし、次に、[無効]を選択します。
- b. [Windows 認証]を右クリックし、次に、[有効]を選択します..

Windows 認証方式用の仮想ディレクトリが設定されます。

注: Web サーバを再起動すると、これらの変更が反映されます。

IIS 7.0 に SiteMinder Web エージェントを設定する方法

IIS 7.0 Web サーバ用の SiteMinder Web エージェントを設定するには、以下の手順に従います。

1. 以下の前提条件を確認します。
 - SiteMinder Web エージェントがインストールされていること。
 - SiteMinder Web エージェントを実行するコンピュータがインストール後に (SiteMinder Web エージェント設定ウィザードを実行することなく) 再起動されたこと。
 - Web サーバ (IIS) の役割が Web サーバに追加されていること。
2. IIS 7.0 Web サーバに役割 (ロール) サービスを追加します。
3. SiteMinder のクラシック モード アプリケーション プールを設定します。
4. SiteMinder で保護するアプリケーションをクラシック モード アプリケーション プールへ移動させます。
5. Web エージェント設定ウィザードを実行します。
6. SiteMinder Windows 認証方式を使用する場合は、IIS Web サーバ上に仮想ディレクトリを設定します。
7. SiteMinder で保護する追加の Web サイトにハンドラ マッピングを追加します。

注: これらの設定は、Web エージェント設定ウィザードによって IIS Web サーバのデフォルト Web サイトに自動的に追加されます。

8. SiteMinder で保護する追加の Web サイトに SiteMinder ISAPI フィルタを追加します。

注: これらの設定は、Web エージェント設定ウィザードによって IIS Web サーバのデフォルト Web サイトに自動的に追加されます。

詳細情報

[IIS Web エージェントで必要とされるエージェント設定パラメータ \(P. 27\)](#)

IIS 7.x Web サーバへの役割サービスの追加

IIS 7.x Web サーバ上で SiteMinder Web エージェントを操作する前に、SiteMinder Web エージェントによって必要とされる役割サービスを利用するために Web サーバを設定する必要があります。

IIS 7.x Web サーバにロール サービスを追加する方法

1. Windows Server 2008 システムで、[スタート]-[管理ツール]-[サーバー マネージャ]をクリックします。

注: [ユーザ アカウント コントロール]ダイアログ ボックスが表示されたら、[続行]をクリックします。

[サーバー マネージャ]が開きます。

2. [役割]を展開し、[Web サーバー (IIS)]をクリックします。

Web サーバ (IIS) の概要ページが表示されます。

3. [役割の概要]セクションで、[役割の追加]をクリックします。

役割の追加ウィザードが開始されます。

4. ウィザードを使用して以下の役割サービスを IIS 7.0 Web サーバに追加します。

- ASP.NET
- CGI
- ISAPI 機能拡張
- ISAPI フィルタ
- IIS 管理コンソール
- Windows 認証 (SiteMinder Windows 認証方式の場合)

役割サービスが IIS Web サーバに追加されます。

詳細情報:

[Windows 認証役割サービスの IIS 7.x Web サーバへの追加 \(P. 85\)](#)

Windows 認証役割サービスの IIS 7.x Web サーバへの追加

SiteMinder Windows 認証方式を使用するには、Windows 認証役割サービスを IIS 7.x Web サーバに追加します。SiteMinder を IIS 7.x Web サーバに追加する場合は、他にも追加する役割サービスがあります。すでに IIS 7.x Web サーバ上で SiteMinder を使用しており、SiteMinder Windows 認証方式を追加するだけの場合は、以下の手順に従います。

Windows 認証役割サービスの IIS 7.x Web サーバへの追加

1. Windows Server 2008 システムで、[スタート]-[管理ツール]-[サーバー マネージャ]をクリックします。

注: [ユーザ アカウント コントロール]ダイアログ ボックスが表示されたら、[続行]をクリックします。

[サーバー マネージャ]が開きます。

2. [役割の概要]セクションで[役割]を見つけ、リストし、[Web サーバー(IIS)]リンクをクリックします。

Web サーバ (IIS) ページが表示されます。

3. [役割サービスの追加]をクリックします。

[役割サービスの追加]ウィザードが開始します。

4. ウィザードを使用して以下の役割サービスをお使いの IIS 7.x Web サーバに追加します。

- Windows 認証

認証役割サービスがお使いの IIS 7.x Web サーバに追加されました。

詳細情報:

[IIS 7.x Web サーバへの役割サービスの追加](#) (P. 98)

SiteMinder Web エージェント用のクラシック モード アプリケーション プールの設定

IIS 7.x Web サーバ上のアプリケーション プールは以下のいずれかのモードを使用します。

- 統合モード(デフォルト)
- クラシック モード

IIS 7.x Web サーバでは、SiteMinder Web エージェントはクラシック モードで設定されたアプリケーション プール上でのみ作動します。IIS 7.x Web サーバのデフォルト アプリケーション プールは統合モードを使用します。IIS 7.5 Web サーバ上で SiteMinder Web Agent を使用するには、以下のいずれかを実行します。

- IIS 7.x サーバ上で統合モードで実行する必要があるアプリケーションがない場合は、デフォルトのアプリケーション プール モード (DefaultAppPool) をクラシック モードに切り替えます。
- アプリケーション プールを統合モードで使用する他のアプリケーションと同じ IIS サーバ上で SiteMinder Web エージェントを実行する場合は、以下の手順で、SiteMinder 用にクラシック モードのアプリケーション プールを新規作成します。

SiteMinder Web エージェント用にクラシック モード アプリケーション プールを設定する方法

1. IIS マネージャを開きます。
IIS マネージャの開始ページが表示されます。
2. Web サーバを展開します。
[アプリケーション プール]アイコンおよび[Web サイト]フォルダが表示されます。
3. [アプリケーション プール]アイコンを右クリックし、[アプリケーション プールの追加]を選択します。
[アプリケーション プールの追加]ダイアログ ボックスが表示されます。カーソルは[名前]フィールドにあります。
4. 新しいアプリケーション プールの名前を入力します。認識するのが簡単な「SiteMinder」のような名前を使用することをお勧めします。
5. [マネージ パイプライン モード]ドロップダウンリストをクリックし、「クラシック」を選択します。
6. [OK]をクリックします。

Web エージェント用にクラシック モード アプリケーション プールが作成されました。

SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動

IIS 7.x Web サーバでクラシック モード アプリケーション プールを作成した後、アプリケーションを現在のアプリケーション プールから SiteMinder によって使用されるクラシック モード アプリケーション プールに移動させる必要があります。

SiteMinder で保護するアプリケーションのクラシック モード アプリケーション プールへの移動方法

1. IIS マネージャを開きます。
2. [接続] ペインで、Web サーバを展開します。
[アプリケーション プール] アイコンおよび[サイト] フォルダが表示されます。
3. アプリケーション プールをクリックします。
アプリケーション プールのリストが表示されます。
4. 移動させるアプリケーションが含まれるアプリケーション プールを右クリックして、[アプリケーションの表示]を選択します。たとえば、SiteMinder で保護するアプリケーションが現在デフォルト アプリケーション プール内にある場合は、デフォルト アプリケーション プールを右クリックします。
アプリケーションのリストが表示されます。
5. 移動させるアプリケーションを右クリックし、[アプリケーション プールの変更]を選択します。
[アプリケーション プールの選択] ダイアログ ボックスが表示されます。
6. アプリケーション プール ドロップダウン リストをクリックし、SiteMinder 用に作成したクラシック モード アプリケーション プールを選択します。
7. [OK] をクリックします。
[アプリケーション プールの選択] ダイアログ ボックスが閉じます。アプリケーションは新しいアプリケーション プールに移動しました。

IIS Web エージェントの設定ウィザードの実行

エージェントを設定する前に、エージェントがインストールされているシステムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

IIS Web エージェントを設定する方法

1. 必要に応じて、**Web** エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

注: インストールの直後に **Web** エージェントを設定することを選択した場合、SiteMinder はウィザードを自動的に開始します。

2. トラステッド ホストが登録されている場合は、次の手順にスキップします。そうでない場合は、[ホスト登録]ダイアログ ボックスで[いいえ]を選択して登録をスキップし、[次へ]をクリックします。
3. **Web** エージェントと共に設定する **Web** サーバ インスタンスを選択します。
すでにサーバを **Web** エージェントと共に設定しており、設定ウィザードを実行して追加の **Web** サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した **Web** サーバのリストを示します。
 - a. 次のいずれかをオンにします。
上書き - サーバ インスタンスの設定を上書きします。
保存 - **Web** サーバの設定を保存します。
重要: 以前に設定したサーバのチェック ボックスをオフにすると、**Web** エージェントはこのサーバから削除されます。
 - b. [次へ]をクリックします。
4. [エージェント設定オブジェクト]フィールドで、この **Web** サーバ インスタンスのエージェント設定オブジェクトの名前を入力し、[次へ]をクリックします。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `IISDefaultSettings` を入力します。

5. [Web サーバ設定の概要] ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール] をクリックします。

Web エージェント ファイルがインストールされます。

6. インストールが完了したら、[終了] をクリックします。

注: エージェントが設定されたら、エージェントとトレースメッセージが適切にログ記録されるように、マシンを再起動する必要があります。

詳細情報

[Windows システムに Web エージェントをインストールする \(P. 31\)](#)

Windows 認証方式用の仮想ディレクトリの設定 (IIS 7.0)

<stmdnr>Windows 認証方式を使用するには、IIS 7.0 Web サーバ上で仮想ディレクトリを設定します。仮想ディレクトリには、認証情報に関して NT チャレンジおよびレスポンスが必要です。

Windows 認証方式用の仮想ディレクトリを設定する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開きます。
2. 左側のペインで、以下のアイテムを展開します。
 - Web サーバアイコン
 - サイトフォルダ
 - 既定の Web サイトアイコン
3. 以下のいずれかの手順を実行します。
 - SiteMinder Windows 認証方式で Web サイト全体のリソースをすべて保護するには、[認証]アイコンをダブルクリックし、手順 4 に進みます。
 - SiteMinder Windows 認証方式で Web サイト全体を保護しない場合は、以下の手順を実行します。
 - a. 以下のフォルダを見つけます。
`¥siteminderagent¥ntlm`
 - b. ntlm フォルダをクリックします。
 - c. [認証]アイコンをダブルクリックし、手順 4 に進みます。
[認証]ダイアログ ボックスが表示されます。
4. 以下の手順を実行します。
 - a. [匿名認証]を右クリックし、次に、[無効]を選択します。
 - b. [Windows 認証]を右クリックし、次に、[有効]を選択します..
Windows 認証方式用の仮想ディレクトリが設定されます。

注: Web サーバを再起動すると、これらの変更が反映されます。

SiteMinder で保護する追加の Web サイトへのハンドラ マッピングの追加

SiteMinder で保護する IIS 7.0 Web サーバ内のすべての追加の Web サイト(デフォルトの Web サイト外の)にハンドラ マッピングが必要です。

注: これらの設定は、Web エージェント設定ウィザードによって IIS Web サーバのデフォルト Web サイトに自動的に追加されます。

以下の図では、「Example」という名の Web サイトを表示します。これには、ハンドラ マッピングを手動で追加する必要があります。



追加の Web サイトにハンドラ マッピングを追加する方法

1. インターネット インフォメーション サービス (IIS) マネージャが開きます。
注: [ユーザ アカウント コントロール] ダイアログ ボックスが表示されたら、[続行] をクリックします。
2. [接続] ペインで、Web サーバを展開します。
[サイト] フォルダが表示されます。
3. [サイト] フォルダを展開し、<stmdnr> で保護する他の Web サイトのアイコンをクリックします。
4. IIS セクションの下で、[ハンドラ マッピング] アイコンをダブルクリックします。
インストールされたハンドラ マッピングのリストが表示されます。
5. [アクション] ペインで、[ワイルドカード スクリプト マップの追加] をクリックします。
[ワイルドカード スクリプト マップの追加] ダイアログ ボックスが表示されます。

6. 省略記号ボタン ([実行可能ファイル] フィールドの右) をクリックします。
[開く] ダイアログ ボックスが表示されます。

7. 以下のファイルに移動します。

```
web_agent_home¥bin¥ISAPI6WebAgentDLL
```

注: `web_agent_home` 変数のデフォルト値は `C:¥Program Files¥CA¥webagent` です。

8. [開く] をクリックします。

ISAPI6WebAgentDLL.dll ファイルは、[ワイルドカード スクリプト マップの追加] ダイアログ ボックスに表示されます。

9. [名前] フィールドに、マッピングの名前を入力します。認識するのが簡単な「handler-wa」のような名前を使用することをお勧めします。

10. [OK] をクリックします。

確認のダイアログが表示されます。

11. [はい] をクリックします。

[ワイルドカード スクリプト マップの追加] ダイアログ ボックスが閉じて、マッピングはリストに表示されます。ハンドラ マッピングは保護された Web サイトに追加されます。

12. SiteMinder で保護する個々の追加の Web サイトに対し、手順 3 から 11 を繰り返します。

ハンドラ マッピングが追加されます。

SiteMinder で保護する追加の Web サイトにエージェント ISAPI フィルタを追加

SiteMinder Web [set AGENT value for your book] を IIS 7.0 上の追加の(デフォルト以外) Web サイトで実行するには、保護する個々の追加の Web サイトに SiteMinder ISAPI フィルタを追加します。このフィルタは Web エージェント ISAPI スクリプトおよび他のファイルを実行します。

注: これらの設定は、Web エージェント設定ウィザードによって IIS Web サーバのデフォルト Web サイトに自動的に追加されます。

以下の図では、「Example」という名の Web サイトを表示します。これには、ISAPI フィルタを手動で追加する必要があります。



IIS 6.0 に SiteMinder Web エージェントを設定する方法

IIS 6.0 Web サーバで Web エージェントを使用できるようにするには、以下の手順を実行して前提条件を完了する必要があります。

1. サンプルおよびエラー ファイル ディレクトリに読み取り許可を割り当てる。
2. IIS が Web エージェント ISAPI および CGI 拡張を実行できるようにする。
3. (オプション) Web エージェントのアップロードされたファイルのサイズ制限を増やす。
4. Web エージェント情報を収集する。
5. IIS Web エージェントで設定ウィザードを実行する
6. 他のサードパーティフィルタの前にエージェントフィルタおよび拡張を配置する。

詳細情報

[IIS Web エージェントで必要とされるエージェント設定パラメータ \(P. 27\)](#)

サンプルおよびエラー ファイル ディレクトリへの読み取り許可の割り当て

ネットワーク サービス アカウントは、Web エージェントがフォーム認証情報コレクタ (FCC) ファイルを読み取る任意のディレクトリ、および Web エージェントが Web エージェントのカスタム エラー ファイルを読み取る任意のディレクトリへの読み取り許可を持っている必要があります。

サンプルおよびエラー ファイル ディレクトリに読み取り許可を割り当てる方法

1. Windows エクスプローラを開いて該当するディレクトリに移動します。
 - サンプル: `web_agent_home/samples`
 - カスタム エラー ファイル: 場所またはカスタム エラー ファイル。デフォルトの場所はありません。
2. ディレクトリを右クリックし、[共有]と[セキュリティ]を選択します。
3. [セキュリティ]タブを選択します。
4. [追加]をクリックします。

[ユーザ、コンピュータ、またはグループの選択]ダイアログ ボックスが開きます。
5. 以下のいずれかの操作を行います。
 - a. [このオブジェクトタイプを選択]フィールドおよび[この場所から]フィールドをデフォルト値として使用します。
 - b. [オブジェクト名を入力して選択]フィールドで、[ネットワーク サービス]を入力して[OK]をクリックします。

ディレクトリの[プロパティ]ダイアログ ボックスに戻ります。
6. [ネットワーク サービスの権限]スクロールボックスで、読み取り権限を許可します。
7. [OK] をクリックして終了します。
8. 各ディレクトリに対してこの手順を繰り返します。

IIS がエージェント ISAPI および CGI 拡張を実行できるようにする

IIS 6.0 Web サーバに特定の ISAPI および CGI 拡張を追加し、SiteMinder Web エージェントを設定する前にそれらを実行するサーバ許可を付与する必要があります。これらの拡張は Web エージェント ISAPI および CGI スクリプト、およびその他のファイルを実行します。

拡張と許可を追加する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開き、次に、エージェントに設定している Web サーバを展開します。
2. [Web サービス拡張] をダブルクリックします。
[Web サービス拡張] ペインが表示されます。
3. ISAPI Web エージェント拡張を追加するには、以下を実行します。
 - a. [新しい Web サービス拡張の追加] リンクをクリックします。
[新しい Web サービス拡張] ダイアログ ボックスが開きます。
 - b. [拡張名] フィールドで、「ISAPI6WebAgentDLL」を入力し、[追加] をクリックします。
[ファイルの追加] ダイアログ ボックスが開きます。
 - c. [参照] ボタンをクリックし、次に、`web_agent_home/bin` ディレクトリの `ISAPI6WebAgent.dll` ファイルに移動します。正しいファイルが表示されない場合は、[ファイルタイプ] ドロップダウンリストをクリックし、ISAPI dll ファイル (.dll ファイル用) または CGI exe ファイル (.exe ファイル用) のいずれかを選択します。
注: `web_agent_home` 変数のデフォルトの場所は、以下のディレクトリのいずれかです。
 - (r6.x SP6): `C:\Program Files\CA\webagent`
 - (r12.0 SP3): `/opt/ca/webagent`
 - d. [開く] をクリックします。
ファイルへのパスは [ファイルの追加] ダイアログ ボックスに表示されます。
 - e. [OK] をクリックします。
[新しい Web サービス拡張] ダイアログ ボックスに戻ります。
 - f. [拡張ステータスを許可に設定] チェック ボックスをオンにします。
 - g. [OK] をクリックします。

[新しい Web サービス拡張]ダイアログ ボックスが閉じます。

4. 手順 3 を繰り返して、以下の各 Web エージェント ファイルを追加します。両方のファイルが同じ名前を使用しても、それらが異なるディレクトリにあるため、それぞれに別の拡張を追加する必要があります。
 - `web_agent_home/pw/smpwservicescgi.exe` (拡張名の例: パスワード サービス CGI)
 - `web_agent_home/pw_default/smpwservicescgi.exe` (拡張名の例: PW デフォルト CGI)

同じサーバ上の IIS 6.0 Web エージェントとサードパーティ ソフトウェア

IIS 6.0 Web エージェントは、ISAPI フィルタおよび ISAPI 機能拡張で構成されています。大多数の Web エージェント処理が拡張で発生します。

Web エージェントが WebSphere または ServletExec などの他のサードパーティ ソフトウェアで IIS 6.0 Web サーバにインストールされた場合、エージェントには以下の制限があります。

- Web エージェント フィルタおよび Web エージェント拡張は、Web サーバにインストールされた他のサードパーティ フィルタの前に実行されるように設定する必要があります。
- Web エージェントが ISAPI 機能拡張として実行している、またはこれにより起動されたアプリケーションを保護する場合、Web エージェントを最初のワールドカード アプリケーション マップとして設定する必要があります。
- IIS 6.0 Web サーバは、サードパーティのフィルタおよび拡張の動作方法を強制しません。IIS 6.0 は、Web エージェント拡張を含め、ISAPI 機能拡張を呼び出す前に ISAPI フィルタを処理します。そのため、IIS 6.0 の SiteMinder Web エージェントは、純粋な ISAPI フィルタとして実装されたアプリケーションへのアクセスを認証または認可することができません。この制限は、他のサードパーティ サービスが、ISAPI 機能拡張が呼び出される前に、リクエストを処理してリダイレクトする ISAPI フィルタとして実装されている場合は、Web エージェントと IIS 6.0 Web サーバの他のサードパーティ サービスとの統合に影響を与えます。

エージェントのアップロード ファイルのサイズ制限の増加

IIS 6.0 Web サーバにインストールされた Web エージェントでは、ファイルのアップロードのサイズ制限は 2.5 MB です。このサイズ制限を増加すると、Web サーバの Windows レジストリに新しいキーを追加できます。

この制限よりも大きいファイルをアップロードする方法

1. レジストリ エディタを開きます。

注: 詳細については、Microsoft のドキュメントを参照するか、または <http://support.microsoft.com/> に移動します。

2. 次のディレクトリに移動します。

```
HKEY_LOCAL_MACHINE\SOFTWARE\ca\SiteMinder Web Agent\Microsoft IIS
```

3. 以下の名前を使用して、以前の場所に新しい DWORD レジストリ キーを作成します。

```
MaxRequestAllowed
```

4. このキーの値を希望するサイズ制限に対応するバイト数に設定します。

このキーの値はデフォルト制限を上書きします。このキーの値が 0 以下である場合、2.5 MB (2,500,000 B) のデフォルトが使用されます。このキーは 0 から 4294967295 までの 10 進数の値を受け入れます。

注: IIS 6.0 Web サーバにはそれ自身のサイズ制限があります。Web エージェントの制限を変更しても、IIS 6.0 制限には影響しません。IIS 6.0 サーバの制限を変更する場合は、Microsoft IIS 6.0 ドキュメントまたはオンライン ヘルプを参照してください。

5. レジストリ エディタを閉じます。

サイズ制限が変更されます。

IIS Web エージェントの設定ウィザードの実行

エージェントを設定する前に、エージェントがインストールされているシステムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

IIS Web エージェントを設定する方法

1. 必要に応じて、**Web** エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

注: インストールの直後に **Web** エージェントを設定することを選択した場合、SiteMinder はウィザードを自動的に開始します。

2. トラステッド ホストが登録されている場合は、次の手順にスキップします。そうでない場合は、[ホスト登録]ダイアログ ボックスで[いいえ]を選択して登録をスキップし、[次へ]をクリックします。
3. **Web** エージェントと共に設定する **Web** サーバ インスタンスを選択します。
すでにサーバを **Web** エージェントと共に設定しており、設定ウィザードを実行して追加の **Web** サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した **Web** サーバのリストを示します。
 - a. 次のいずれかをオンにします。
上書き - サーバ インスタンスの設定を上書きします。
保存 - **Web** サーバの設定を保存します。
重要: 以前に設定したサーバのチェック ボックスをオフにすると、**Web** エージェントはこのサーバから削除されます。
 - b. [次へ]をクリックします。
4. [エージェント設定オブジェクト]フィールドで、この **Web** サーバ インスタンスのエージェント設定オブジェクトの名前を入力し、[次へ]をクリックします。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `IISDefaultSettings` を入力します。

5. [Web サーバ設定の概要] ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール] をクリックします。

Web エージェント ファイルがインストールされます。

6. インストールが完了したら、[終了] をクリックします。

注: エージェントが設定されたら、エージェントとトレースメッセージが適切にログ記録されるように、マシンを再起動する必要があります。

詳細情報

[Windows システムに Web エージェントをインストールする \(P. 31\)](#)

他のサードパーティフィルタの前にエージェントフィルタおよび拡張を配置する

IIS 6.0 Web エージェントは、ISAPI フィルタおよび ISAPI 機能拡張で構成されています。大多数の Web エージェント処理が拡張で発生します。

Web エージェントが WebSphere または ServletExec などの他のサードパーティソフトウェアで IIS 6.0 Web サーバにインストールされた場合、エージェントには以下の制限があります。

- Web エージェントフィルタおよび Web エージェント拡張は、Web サーバにインストールされた他のサードパーティフィルタの前に実行されるように設定する必要があります。
- Web エージェントが ISAPI 機能拡張として実行している、またはこれにより起動されたアプリケーションを保護する場合、Web エージェントを最初のワールドカード アプリケーション マップとして設定する必要があります。
- IIS 6.0 Web サーバは、サードパーティのフィルタおよび拡張の動作方法を強制しません。IIS 6.0 は、Web エージェント拡張を含め、ISAPI 機能拡張を呼び出す前に ISAPI フィルタを処理します。そのため、IIS 6.0 の SiteMinder Web エージェントは、純粋な ISAPI フィルタとして実装されたアプリケーションへのアクセスを認証または認可することができません。この制限は、他のサードパーティサービスが、ISAPI 機能拡張が呼び出される前に、リクエストを処理してリダイレクトする ISAPI フィルタとして実装されている場合は、Web エージェントと IIS 6.0 Web サーバの他のサードパーティサービスとの統合に影響を与えます。

IIS 6.0 Web サーバに Web エージェントをインストールすると、エージェントのフィルタは ISAPI フィルタリストの一番上に自動的に配置されます。ただし、Web エージェントをインストールした後に他のサードパーティプラグインをインストールすると、それらのフィルタが優先される場合があります。

IIS 6.0 Web エージェントをインストールして設定した後で、siteminderagent ISAPI フィルタおよび拡張が任意のサードパーティフィルタまたは拡張の前にリストされていることを確認する必要があります。これにより、Web エージェントは、サードパーティの前にリクエストを処理できます。

他のサードパーティフィルタの前にエージェントフィルタおよび拡張を配置する方法

1. 以下の手順を実行することにより ISAPI フィルタを確認します。
 - a. IIS マネージャを開きます。
 - b. Web サイトを選択し、右クリックして[プロパティ]を選択します。

- c. [ISAPI フィルタ] タブを選択します。
 - d. フィルタのリストを確認し、**siteminderagent** がリスト内の最初のエントリであることを確認します。そうでない場合は、[上に移動] ボタンを使用してリストの一番上にそれを配置します。
 - e. [OK] をクリックします。
 - f. IIS マネージャを終了します。
2. 以下の手順を実行することにより ISAPI 機能拡張を確認します。
- a. IIS マネージャを開いて、**Web** サーバを展開します。
 - b. [デフォルト Web サイト] フォルダを右クリックし、[プロパティ] を選択します。
 - c. [ホーム ディレクトリ] タブをクリックし、次に、[設定] をクリックします。
 - d. 以下のファイルが、[ワイルドカード アプリケーション マップ (実装する順序)] フィールドの一番上にある必要があります。

`web_agent_home¥bin¥ISAPI6WebAgent.dll`

注: `web_agent_home` 変数のデフォルトの場所は、以下のディレクトリのいずれかです。

- (r6.x SP6): `C:¥Program Files¥CA¥webagent`
- (r12.0 SP3): `/opt/ca¥webagent`

Windows 認証方式の仮想ディレクトリの設定 (IIS 6.0)

<stmdnr> Windows 認証方式を使用するには、IIS 6.0 Web サーバ上で仮想ディレクトリを設定します。仮想ディレクトリには、認証情報に関して NT チャレンジおよびレスポンスが必要です。

Windows 認証方式用の仮想ディレクトリを設定する方法

1. インターネット インフォメーション サービス (IIS) マネージャを開きます。
2. 左側のペインで、以下のアイテムを展開します。
 - Web サーバアイコン
 - Web サイトフォルダ
3. 以下のいずれかの手順を実行します。
 - Web サイト全体のすべてのリソースを SiteMinder Windows 認証方式で保護するには、「Default Web Site」フォルダを右クリックし、[プロパティ]を選択して手順 4 に進みます。
 - SiteMinder Windows 認証方式で Web サイト全体を保護しない場合は、以下の手順を実行します。
 - a. 以下のフォルダを見つけます。
`¥siteminderagent¥ntlm`
 - b. ntlm フォルダを右クリックし、[プロパティ]を選択して手順 4 に進みます。

[プロパティ]ダイアログボックスが表示されます。

4. [ディレクトリ セキュリティ] タブをクリックします。
5. [匿名アクセスおよび認証制御] グループ ボックスで [編集] をクリックします。

[認証方法]ダイアログ ボックスが表示されます。

6. 以下の手順を実行します。
 - [匿名アクセスを有効にする] チェック ボックスをオフにします。
 - [統合 Windows 認証] チェック ボックスをオンにします。
7. [OK] を 2 回クリックします。

[認証方法]ダイアログ ボックスおよび [プロパティ]ダイアログ ボックスが閉じます。仮想ディレクトリが設定され、認証情報に関して NT チャレンジおよびレスポンスを求めます。

注: これらの変更を反映するため Web サーバを再起動します。

SiteMinder Web エージェントを設定して Microsoft Outlook Web Access を保護する方法

SiteMinder Web エージェントに Microsoft Outlook Web Access の Web サイトを保護させるようにするには、以下の手順に従います。

注: サポートされているコンポーネントのバージョンを特定するには、<http://www.ca.com/jp/support/> で SiteMinder r12.0 SP3 製品サポートマトリックスを参照してください。

1. 以下の前提条件をインストールするか設定します。

- a. Microsoft Exchange Server
- b. IIS 6.0 用に設定された Microsoft Web Access クライアントソフトウェア

注: Microsoft Exchange Server および Web Access クライアントコンポーネントは同じシステム、または別のシステムにインストールできます。両方のコンポーネントが同じシステムにインストールされている場合、1つの Web エージェントのみが必要です。コンポーネントが別のシステムにインストールされている場合、2つの Web エージェントが使用されます。別のシステムが使用されている場合、Exchange Server がバックエンドシステムとして動作する一方で、Web Access クライアントはフロントエンドシステムとして動作します。

c. SiteMinder ポリシー サーバは以下を伴います。

- ポリシー ストアおよびユーザ ディレクトリに使用される Microsoft Active Directory
- セッション サーバに使用される SQL Server データベース インスタンス
- 保護する Microsoft Outlook Web Access リソースに関連付けられたレルム (r6.x) またはアプリケーション (r12.0 SP3) に対して有効な永続的セッション

2. Microsoft Exchange Server をホストする IIS Web サーバで以下の手順に従います。

- a. SiteMinder ISAPI フィルタがリストの最初に表示されていることを確認します。
- b. IIS で Outlook 拡張を実行できるようにします。
- c. デフォルトの Web サイト ホーム ディレクトリの場所と実行権限の設定
- d. Exchange Web サイトへの ISAPI 機能拡張の追加

- e. Exchange Web サイトにディレクトリ セキュリティを設定します。
 - f. Exchweb 仮想サイトに ISAPI 機能拡張を設定します。
 - g. Exchweb 仮想サイトにディレクトリ セキュリティを設定します。
 - h. デフォルトの Web サイト ホーム ディレクトリの場所と実行権限の設定を設定します。
3. Microsoft Outlook Web Access クライアントをホストする IIS Web サーバ上で手順 2a から 2g を繰り返します。
 4. SiteMinder が Outlook Web Access Web サイトを保護していることを確認します。

SiteMinder ISAPI フィルタがリストの最初に表示されていることを確認します。

IIS 6.0 Web エージェントは、ISAPI フィルタおよび ISAPI 機能拡張で構成されています。大多数の Web エージェント処理が拡張で発生します。

Web エージェントが WebSphere または ServletExec などの他のサードパーティソフトウェアで IIS 6.0 Web サーバにインストールされた場合、エージェントには以下の制限があります。

- Web エージェント フィルタおよび Web エージェント拡張は、Web サーバにインストールされた他のサードパーティフィルタの前に実行されるように設定する必要があります。
- Web エージェントが ISAPI 機能拡張として実行している、またはこれにより起動されたアプリケーションを保護する場合、Web エージェントを最初のワールドカード アプリケーション マップとして設定する必要があります。
- IIS 6.0 Web サーバは、サードパーティのフィルタおよび拡張の動作方法を強制しません。IIS 6.0 は、Web エージェント拡張を含め、ISAPI 機能拡張を呼び出す前に ISAPI フィルタを処理します。そのため、IIS 6.0 の SiteMinder Web エージェントは、純粋な ISAPI フィルタとして実装されたアプリケーションへのアクセスを認証または認可することができません。この制限は、他のサードパーティサービスが、ISAPI 機能拡張が呼び出される前に、リクエストを処理してリダイレクトする ISAPI フィルタとして実装されている場合は、Web エージェントと IIS 6.0 Web サーバの他のサードパーティサービスとの統合に影響を与えます。

IIS 6.0 Web サーバに Web エージェントをインストールすると、エージェントのフィルタは ISAPI フィルタリストの一番上に自動的に配置されます。ただし、Web エージェントをインストールした後に他のサードパーティプラグインをインストールすると、それらのフィルタが優先される場合があります。

IIS 6.0 Web エージェントをインストールして設定した後で、siteminderagent ISAPI フィルタおよび拡張が任意のサードパーティフィルタまたは拡張の前にリストされていることを確認する必要があります。これにより、Web エージェントは、サードパーティの前にリクエストを処理できます。

他のサードパーティフィルタの前にエージェント フィルタおよび拡張を配置する方法

1. 以下の手順を実行することにより ISAPI フィルタを確認します。
 - a. IIS マネージャを開きます。
 - b. Web サイトを選択し、右クリックして[プロパティ]を選択します。

- c. [ISAPI フィルタ] タブを選択します。
 - d. フィルタのリストを確認し、**siteminderagent** がリスト内の最初のエントリであることを確認します。そうでない場合は、[上に移動] ボタンを使用してリストの一番上にそれを配置します。
 - e. [OK] をクリックします。
 - f. IIS マネージャを終了します。
2. 以下の手順を実行することにより ISAPI 機能拡張を確認します。
 - a. IIS マネージャを開いて、**Web** サーバを展開します。
 - b. [デフォルト Web サイト] フォルダを右クリックし、[プロパティ] を選択します。
 - c. [ホーム ディレクトリ] タブをクリックし、次に、[設定] をクリックします。
 - d. 以下のファイルが、[ワイルドカード アプリケーション マップ (実装する順序)] フィールドの一番上にある必要があります。

`web_agent_home\bin\ISAPI6WebAgent.dll`

注: `web_agent_home` 変数のデフォルトの場所は、以下のディレクトリのいずれかです。

- (r6.x SP6): `C:\Program Files\CA\webagent`
- (r12.0 SP3): `/opt/ca/webagent`

IIS で Outlook 拡張を実行できるようにする

IIS Web サーバは、Microsoft Outlook 用の Web サービス機能拡張を実行する権限を持っている必要があります。

IIS で Outlook 拡張を実行できるようにする方法

1. インターネット インフォメーション サービス (IIS) マネージャを開き、次に、エージェントに設定している Web サーバを展開します。
2. [Web サービス拡張] をダブルクリックします。
[Web サービス拡張] ペインが表示されます。
3. 以下の拡張に [許可] のステータスが表示されていることを確認します。
 - Microsoft Exchange クライアント アクセス
 - Microsoft Exchange サーバ

デフォルトの Web サイトのディレクトリ場所と実行権限の設定

IIS Web サーバのデフォルトの Web サイトには特定のディレクトリ場所と実行権限が必要です。

デフォルトの Web サイトのディレクトリ場所と実行権限を設定する方法

1. インターネット インフォメーション サービス (IIS) マネージャが開きます。
2. [既定の Web サイト]フォルダを右クリックし、[プロパティ]を選択します。
[既定の Web サイトのプロパティ]ダイアログ ボックスが表示されます。
3. [ホーム ディレクトリ]タブをクリックし、以下の設定を確認します。
 - ローカル パス: `c:\inetpub\wwwroot`
 - 実行権限: スクリプトと実行可能ファイル

デフォルトの Web サイトのディレクトリ場所と実行権限が設定されます。

Exchange Web サイトへの ISAPI 機能拡張の追加

IIS Web サーバの Microsoft Exchange Web サイトは、Microsoft Outlook Web Access と動作するために SiteMinder ISAPI 機能拡張が必要です。

Exchange Web サイトに ISAPI 機能拡張を追加する方法

1. Internet Information Services (IIS) Manager を開いて、Web Sites フォルダを展開します。
Web サイトのリストが表示されます。
2. Exchange フォルダを右クリックして、[プロパティ]を選択します。
[Exchange のプロパティ]ダイアログ ボックスが表示されます。
3. [仮想ディレクトリ]タブをクリックし、以下の設定を確認します。
 - ローカルパス: *path_to_the_exchange_folder* (例: C:\Program Files\Microsoft\Exchange Server\ClientAccess\owa)
 - アプリケーション名: Exchange
 - 実行権限: スクリプトと実行可能ファイル
4. [設定]をクリックします。
[アプリケーションの構成]ダイアログ ボックスが表示されます。
5. [挿入]をクリックします。
[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスが表示されます。
6. [参照]をクリックして、以下のファイルに移動します。
C:\Program Files\CA\webagent\bin\ISAPI6WebAgent.dll
7. [開く]をクリックします。
パスが[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスに表示されます。
8. [ファイルの存在を確認する]チェック ボックスをオフにします。
9. [OK]をクリックします。
[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスを閉じます。DLL ファイルは[ワイルドカード アプリケーション マップ (実装する順序)]リストに表示されます。
10. [OK]をクリックします。
[アプリケーションの構成]ダイアログ ボックスが閉じます。

11. [OK]をクリックします。

[Exchange のプロパティ]ダイアログ ボックスが閉じます。Exchange Web サイトに ISAPI 機能拡張が追加されます。

Exchange Web サイトへのディレクトリ セキュリティの設定

IIS Web サーバの Microsoft Exchange Web サイトは、Microsoft Outlook Web Access と動作するために特定のディレクトリのセキュリティ設定が必要です。

Exchange Web サイトにディレクトリ セキュリティを設定する方法

1. Internet Information Services (IIS) Manager を開いて、Web Sites フォルダを展開します。
Web サイトのリストが表示されます。
2. Exchange フォルダを右クリックして、[プロパティ]を選択します。
[Exchange のプロパティ]ダイアログ ボックスが表示されます。
3. [ディレクトリ セキュリティ]タブをクリックします。
4. [認証とアクセス制御]設定セクションで、[編集]をクリックします。
5. [認証方法]ダイアログ ボックスが表示されます。
6. 以下の設定を確認します。
 - [匿名アクセスを有効にする]チェック ボックスがオンになっている。
 - [認証済みアクセス]セクションのチェック ボックスがすべてオフになっている。
7. [OK]をクリックします。
[認証方法]ダイアログ ボックスが閉じます。
8. [OK]をクリックします。
[Exchange のプロパティ]ダイアログ ボックスが閉じます。Exchange Web サイトにディレクトリ セキュリティが設定されます。

Exchweb Web サイトへの ISAPI 機能拡張の追加

IIS Web サーバの Microsoft Exchweb Web サイトは、Microsoft Outlook Web Access と動作するために SiteMinder ISAPI 機能拡張が必要です。

Exchweb Web サイトに ISAPI 機能拡張を追加する方法

1. Internet Information Services (IIS) Manager を開いて、Web Sites フォルダを展開します。
Web サイトのリストが表示されます。
2. Exchweb フォルダを右クリックして、[プロパティ]を選択します。
[Exchweb のプロパティ]ダイアログ ボックスが表示されます。
3. [仮想ディレクトリ]タブをクリックし、以下の設定を確認します。
 - ローカルパス: *path_to_the_exchweb_folder*
 - アプリケーション名: Exchweb
 - 実行権限: スクリプトと実行可能ファイル
4. [設定]をクリックします。
[アプリケーションの構成]ダイアログ ボックスが表示されます。
5. [挿入]をクリックします。
[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスが表示されます。
6. [参照]をクリックして、以下のファイルに移動します。
C:\Program Files\CA\webagent\bin\ISAPI6WebAgent.dll
7. [開く]をクリックします。
パスが[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスに表示されます。
8. [ファイルの存在を確認する]チェック ボックスをオフにします。
9. [OK]をクリックします。
[アプリケーションの拡張子マッピングの追加/編集]ダイアログ ボックスを閉じます。DLL ファイルは[ワイルドカード アプリケーション マップ (実装する順序)]リストに表示されます。
10. [OK]をクリックします。
[アプリケーションの構成]ダイアログ ボックスが閉じます。

11. [OK]をクリックします。

[Exchweb のプロパティ]ダイアログ ボックスが閉じます。Exchweb Web サイトに ISAPI 機能拡張が追加されます。

Exchweb Web サイトへのディレクトリ セキュリティの設定

IIS Web サーバの Microsoft Exchweb Web サイトは、Microsoft Outlook Web Access と動作するために特定のディレクトリのセキュリティ設定が必要です。

Exchweb Web サイトにディレクトリ セキュリティを設定する方法

1. Internet Information Services (IIS) Manager を開いて、Web Sites フォルダを展開します。
Web サイトのリストが表示されます。
2. Exchweb フォルダを右クリックして、[プロパティ]を選択します。
[Exchweb のプロパティ]ダイアログ ボックスが表示されます。
3. [ディレクトリ セキュリティ]タブをクリックします。
4. [認証とアクセス制御]設定セクションで、[編集]をクリックします。
5. [認証方法]ダイアログ ボックスが表示されます。
6. 以下の設定を確認します。
 - [匿名アクセスを有効にする]チェック ボックスがオンになっている。
 - [認証済みアクセス]セクションのチェック ボックスがすべてオフになっている。
7. [OK]をクリックします。
[認証方法]ダイアログ ボックスが閉じます。
8. [OK]をクリックします。
[Exchweb のプロパティ]ダイアログ ボックスが閉じます。Exchweb Web サイトにディレクトリ セキュリティが設定されます。

デフォルトの Web サイトのディレクトリ場所と実行権限の設定

IIS Web サーバの owa Web サイトには特定のディレクトリ場所と実行権限が必要です。

owa Web サイトのディレクトリ場所と実行権限を設定する方法

1. インターネット インフォメーション サービス (IIS) マネージャが開きます。
2. [owa Web サイト]フォルダを右クリックし、[プロパティ]を選択します。
[owa Web サイトのプロパティ]ダイアログ ボックスが表示されます。
3. [ホーム ディレクトリ]タブをクリックし、以下の設定を確認します。
 - ローカル パス: *full_path_to_the_owa_folder*
 - 実行権限: スクリプトと実行可能ファイルowa Web サイトのディレクトリ場所と実行権限が設定されます。

SiteMinder が Outlook Web Access Web サイトを保護していることを確認します。

Microsoft Exchange および Microsoft Outlook Web Access の Web サイトを設定した後に、SiteMinder Web エージェントがそれらを保護していることを確認できます。

SiteMinder が Outlook Web Access Web サイトを保護していることを確認します。

1. Web エージェントの有効化
2. Outlook Web Access の受信トレイ ページを開きます。次に URL の例を示します。
`http://exchange_server_name.example.com/owa/`
SiteMinder ログイン ページが表示されます。
3. クレデンシャルを入力し、[ログイン]をクリックします。
[受信トレイ]が表示されます。

第 6 章: Oracle iPlanet Web エージェントの設定

このセクションには、以下のトピックが含まれています。

[Windows での設定ウィザードの実行 \(P. 127\)](#)

[GUI またはコンソール モードを使用する Oracle iPlanet Web エージェントの設定 \(P. 130\)](#)

[Oracle iPlanet Web サーバの手動設定 \(P. 133\)](#)

[Oracle iPlanet Web サーバファイルへの変更の適用 \(P. 135\)](#)

Windows での設定ウィザードの実行

注: SiteMinder エージェント設定ウィザードは、Oracle iPlanet (以前の Sun Java System) Web サーバ上のデフォルトの `obj.conf` ファイルのみを変更します。SiteMinder で他のインスタンスまたはリバースプロキシ展開を保護するには、デフォルトの `obj.conf` ファイルから、対応する `<instance_name>-obj.conf` ファイルに SiteMinder 設定をコピーします。たとえば、Web サーバのインストール時に `obj.conf` ファイルが作成されましたが、その後 `my_server.example.com` という名前のサーバインスタンスを追加したとします。SiteMinder で `my_server.example.com` 上のリソースを保護するには、`obj.conf` ファイルから `my_server.example.com-obj.conf` ファイルに、ウィザードによって追加された SiteMinder 設定をコピーします。

Oracle iPlanet Web サーバに Web エージェントを設定する方法

1. 必要に応じて、Web エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

注: インストールの直後に Web エージェントを設定することを選択した場合、SiteMinder はウィザードを自動的に開始します。

2. すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、[ホスト登録]ダイアログ ボックスで[いいえ]を選択して登録をスキップし、[次へ]をクリックします。

トラステッド ホストを登録する場合は、プラットフォームのインストールの章に移動します。

3. **Web エージェントと共に設定する Web サーバ インスタンス**を選択します。

すでにサーバを **Web エージェントと共に設定**しており、設定ウィザードを実行して追加の **Web サーバ インスタンス**を設定する場合は、ウィザードに **[Select One or More Instances to Overwrite]**ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した **Web サーバ**のリストを示します。

- a. 次のいずれかをオンにします。

上書き - サーバ インスタンスの設定を上書きします。

保存 - **Web サーバ**の設定を保存します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、**Web エージェント**はこのサーバから削除されます。

- b. [次へ]をクリックします。

4. [エージェント設定オブジェクト]フィールドで、この **Web サーバ インスタンス**のエージェント設定オブジェクトの名前を入力し、[次へ]をクリックします。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `iPlanetDefaultSettings` を入力します。

5. 該当する場合、[SSL 認証]ダイアログ ボックスにリストされている高度な SSL 認証方式のうちの 1 つを選択します。エージェントが高度な認証を提供していない場合は、[高度な認証なし]を選択します。[次へ]をクリックします。

選択肢を以下に示します。

- **SSL を介した HTTP 基本** - ユーザ名とパスワードに基づいてユーザを識別します。クレデンシャル配信は、暗号化された **Secure Sockets Layer (SSL)** 接続上で常に実行されます。
- **X509 クライアント証明書 - X.509 V3 クライアント証明書**に基づいてユーザを識別します。デジタル証明書はユーザのシグネチャとして機能します。証明書認証では **SSL 通信**を使用します。

- X509 クライアント証明書と HTTP Basic - X.509 クライアント証明書と基本認証を組み合わせたものです。ユーザの X.509 クライアント証明書は検証される必要があり、また、ユーザは有効なユーザ名とパスワードを提供する必要があります。
- X509 クライアント証明書または HTTP Basic - X.509 クライアント証明書と基本認証を組み合わせたものです。ユーザの X.509 クライアント証明書は検証されるか、または、ユーザは有効なユーザ名とパスワードを提供する必要があります。
- X509 クライアント証明書またはフォーム - X.509 クライアント証明書または HTML フォームの認証方式では、X.509 クライアント証明書の使用とカスタマイズされた HTML フォームの使用を組み合わせることで認証情報を収集します。この方式を使用して、ユーザの X.509 クライアント証明書が検証されるか、または、ユーザは HTML フォームによってリクエストされたクレデンシャルを提供する必要があります。
- X509 クライアント証明書およびフォーム - X.509 クライアント証明書および HTML フォームの認証方式では、X.509 クライアント証明書の使用とカスタマイズされた HTML フォームの使用を組み合わせることで認証情報を収集します。この方式を使用して、ユーザの X.509 クライアント証明書が検証され、また、ユーザは HTML フォームによってリクエストされたクレデンシャルを提供する必要があります。

注: 高度な認証方式に関する追加の情報については、「ポリシー サーバ設定ガイド」を参照してください。

6. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。
Web エージェントファイルがインストールされ、[構成の完了]ダイアログボックスが表示されます。
7. [終了]をクリックして設定ウィザードを終了します。
8. Web エージェントを以下のようにして有効にします。
 - a. 以下の場所にある WebAgent.conf ファイルを開きます。
`Sun_Java_System_server_home%servers%https-hostname%config`
 - b. EnableWebAgent パラメータを Yes に設定します。
 - c. ファイルを保存します。
9. Oracle iPlanet Web サーバファイルに変更を適用します。これは、エージェントの設定を有効にするために必要です。

詳細情報

[Oracle iPlanet Web サーバファイルへの変更の適用](#) (P. 135)

GUI またはコンソール モードを使用する Oracle iPlanet Web エージェントの設定

これらの手順は GUI およびコンソール モードを設定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。

- 対応する数を入力してオプションを選択するように指示される場合があります。たとえば、Oracle iPlanet Web サーバを選択するには、このサーバに対応する3を入力します。
- 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。
- 入力するパスワードはすべてクリア テキストで表示されます。この問題の回避策として、インストールを GUI または無人モードで実行します。

各モードのプロンプトにより、順を追ってプロセスをガイドします。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`

UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

Oracle iPlanet Web サーバに Web エージェントを設定する方法

注: SiteMinder エージェント設定ウィザードは、Oracle iPlanet (以前の Sun Java System) Web サーバ上のデフォルトの `obj.conf` ファイルのみを変更します。SiteMinder で他のインスタンスまたはリバースプロキシ展開を保護するには、デフォルトの `obj.conf` ファイルから、対応する `<instance_name>-obj.conf` ファイルに SiteMinder 設定をコピーします。たとえば、Web サーバのインストール時に `obj.conf` ファイルが作成されましたが、その後 `my_server.example.com` という名前のサーバ インスタンスを追加したとします。SiteMinder で `my_server.example.com` 上のリソースを保護するには、`obj.conf` ファイルから `my_server.example.com-obj.conf` ファイルに、ウィザードによって追加された SiteMinder 設定をコピーします。

1. 必要に応じて、設定ウィザードを開始します。
 - a. コンソール ウィンドウを開きます。
 - b. `web_agent_home/install_config_info` に移動します。
 - c. 以下のいずれかのコマンドを入力します。

GUI モード: ./<ファイル名>-wa-config.bin

コンソール モード: ./<ファイル名>-wa-config.bin -i console

- すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、ホスト登録をスキップするオプションを選択し、[次へ]をクリックします。

トラステッド ホストを登録する場合は、プラットフォームのインストールの章に移動します。

- [Web サーバの選択]ダイアログ ボックスで、iPlanet または Sun ONE Web サーバのオプションを選択して[次へ]をクリックします。
- Sun Java System Web サーバがインストールされているルート パスを指定して[次へ]をクリックします。たとえば、/opt/iPlanet/servers。
[選択]をクリックすると、ルート ディレクトリを検索することができます。
- Web エージェントと共に設定する Web サーバ インスタンスを選択します。

すでにサーバを Web エージェントと共に設定しており、設定ウィザードを実行して追加の Web サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した Web サーバのリストを示します。

- 次のいずれかをオンにします。

上書き - サーバ インスタンスの設定を上書きします。

保存 - Web サーバの設定を保存します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、Web エージェントはこのサーバから削除されます。

- [次へ]をクリックします。

- [エージェント設定オブジェクト]フィールドで、この Web サーバ インスタンスのエージェント設定オブジェクトの名前を入力します。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は iPlanetDefaultSettings を入力します。

- 該当する場合、[SSL 認証]ダイアログ ボックスにリストされている高度な SSL 認証方式のうちの一つを選択します。エージェントが高度な認証を提供していない場合は、[高度な認証なし]を選択します。選択したら、[次へ]をクリックします。

選択肢を以下に示します。

- SSL を介した HTTP 基本 - ユーザ名とパスワードに基づいてユーザを識別します。クレデンシャル配信は、暗号化された Secure Sockets Layer (SSL) 接続上で常に実行されます。
- X.509 クライアント証明書 - X.509 V3 クライアント証明書に基づいてユーザを識別します。デジタル証明書はユーザのシグネチャとして機能します。証明書認証では SSL 通信を使用します。
- X.509 クライアント証明書と HTTP Basic - X.509 クライアント証明書と基本認証を組み合わせたものです。ユーザの X.509 クライアント証明書は検証される必要があり、また、ユーザは有効なユーザ名とパスワードを提供する必要があります。
- X.509 クライアント証明書または HTTP Basic - X.509 クライアント証明書と基本認証を組み合わせたものです。ユーザの X.509 クライアント証明書は検証されるか、または、ユーザは有効なユーザ名とパスワードを提供する必要があります。
- X.509 クライアント証明書またはフォーム - X.509 クライアント証明書または HTML フォームの認証方式では、X.509 クライアント証明書の使用とカスタマイズされた HTML フォームの使用を組み合わせることで認証情報を収集します。この方式を使用して、ユーザの X.509 クライアント証明書が検証されるか、または、ユーザは HTML フォームによってリクエストされたクレデンシャルを提供する必要があります。
- X.509 クライアント証明書およびフォーム - X.509 クライアント証明書および HTML フォームの認証方式では、X.509 クライアント証明書の使用とカスタマイズされた HTML フォームの使用を組み合わせることで認証情報を収集します。この方式を使用して、ユーザの X.509 クライアント証明書が検証され、また、ユーザは HTML フォームによってリクエストされたクレデンシャルを提供する必要があります。

注: 詳細については、ポリシー サーバドキュメントを参照してください。

8. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェントファイルがインストールされ、[構成の完了]メッセージが表示されます。

9. インストールが完了したら、[終了]をクリックします。

10. Web エージェントを以下のようにして有効にします。

- a. 以下の場所にある `WebAgent.conf` ファイルを開きます。

`Sun_Java_System_server/servers/https-hostname/config`

- b. `EnableWebAgent` パラメータの値を[はい]に設定します。

- c. ファイルを保存します。
 - d. Web サーバを再起動します。
11. Oracle iPlanet Web サーバファイルに変更を適用します。これは、エージェントの設定を有効にするために必要です。

詳細情報

[Oracle iPlanet Web サーバファイルへの変更の適用 \(P. 135\)](#)

Oracle iPlanet Web サーバの手動設定

SiteMinder Web エージェント設定ウィザードでは、Oracle iPlanet Web サーバのデフォルトインスタンスのみを設定します。SiteMinder を使用するために Oracle iPlanet Web サーバに別のインスタンスを設定する場合は、そのサーバインスタンスによって使用される `obj.conf` ファイルを手動で編集する必要があります。手動設定を必要とするサーバインスタンスの例を以下に示します。

- デフォルト以外のディレクトリにインストールされたサーバ
- リバースプロキシとして設定するサーバ (`obj.conf` ファイルに SiteMinder 設定を追加する前に、Oracle iPlanet インターフェースを使用してリバースプロキシを設定することをお勧めします)。

注: SiteMinder エージェント設定ウィザードは、Oracle iPlanet (以前の Sun Java System) Web サーバ上のデフォルトの `obj.conf` ファイルのみを変更します。SiteMinder で他のインスタンスまたはリバースプロキシ展開を保護するには、デフォルトの `obj.conf` ファイルから、対応する `<instance_name>-obj.conf` ファイルに SiteMinder 設定をコピーします。たとえば、Web サーバのインストール時に `obj.conf` ファイルが作成されましたが、その後 `my_server.example.com` という名前のサーバインスタンスを追加したとします。SiteMinder で `my_server.example.com` 上のリソースを保護するには、`obj.conf` ファイルから `my_server.example.com-obj.conf` ファイルに、ウィザードによって追加された SiteMinder 設定をコピーします。

- 同じコンピュータにある仮想サーバ

Oracle iPlanet Web サーバを手動で設定する方法

1. 設定するサーバインスタンスのディレクトリを検索します。
2. `obj.conf` ファイルをテキストエディタで開きます。
3. 次の行を検索します。

```
<Object name="default">
```

4. 前の行の下に新しい行を挿入し、次に、以下のテキストを追加します。

```
AuthTrans fn="SiteMinderAgent"
```

5. 次の行を検索します。

```
AuthTrans fn="match-browser" browser="*MSIE*" ssl-unclean-shutdown="true"
```

6. 前の行の下に新しい行を挿入し、次に、以下のテキストを追加します。

```
NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi" dir="web_agent_home/pw" name="cgi"
```

```
NameTrans fn="pfx2dir" from="/siteminderagent/pw" dir="web_agent_home/pw"
```

```
NameTrans fn="pfx2dir" from="/siteminderagent/jpw" dir="web_agent_home/jpw"
```

```
NameTrans fn="pfx2dir" from="/siteminderagent/redirectjsp"
```

```
dir="web_agent_home/affwebservices/redirectjsp"
```

```
NameTrans fn="pfx2dir" from="/siteminderagent/certoptional"
```

```
dir="web_agent_home/samples"
```

```
NameTrans fn="pfx2dir" from="/siteminderagent" dir="web_agent_home/samples"
```

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

```
Windows インストールのデフォルトの場所: C:\Program
Files\CA\webagent
```

```
UNIX インストールのデフォルトの場所: /opt/ca/webagent
```

7. 次の行を検索します。

```
NameTrans fn="ntrans-j2ee" name="j2ee"
```

8. 前の行の下に新しい行を挿入し、次に、以下のテキストを追加します。

```
PathCheck fn="SmRequireAuth"
```

9. 以下の行を削除します。

```
NameTrans fn="pfx2dir" from="/mc-icons" dir="C:/Program
Files/Sun/WebServer7.0/lib/icons" name="es-internal"
```

10. 次の行を検索します。

```
ObjectType fn="force-type" type="text/plain"
```

11. 前の行の下に新しい行を挿入し、次に、以下のテキストを追加します。

```
Service method="(GET|POST)" fn="SmAdvancedAuth"
```

12. `obj.conf` ファイルを保存します。

Oracle iPlanet Web サーバは手動設定されました。

Oracle iPlanet Web サーバ ファイルへの変更の適用

Web エージェント設定ウィザードは、Oracle iPlanets.conf、obj.conf、および mime.types ファイルに変更を加えます。Oracle iPlanet Administration コンソールを使用する予定である場合は、コンソールに変更を行う前に、これらのファイルへの変更を適用する必要があります。さもないと、Web エージェント設定が失われる場合があります。設定が失われた場合は、設定ウィザードを使用して Web エージェントを再設定します。

注: エージェントが高度な認証方式をサポートするように設定された場合、Web エージェントは、Oracle iPlanet Web サーバの obj.conf ファイルに設定を追加します。エージェントが別の高度な認証方式をサポートするように再設定された場合でも、SiteMinder はこれらの設定を削除しません。管理者は、関係のなくなった設定を削除するために obj.conf ファイルを手動で編集する必要があります。

Oracle iPlanet 設定ファイルに変更を適用する方法

1. Oracle iPlanet 管理サーバ コンソールにログインします。
2. [サーバ]タブから、Web エージェントがインストールされている Web サーバを選択し、[管理]をクリックします。
3. ダイアログ ボックスの右端で、[適用]をクリックします。
変更された設定ファイルのロードに関する警告メッセージが表示されます。
4. [設定ファイルのロード]をクリックします。
5. コンソールを終了します。
6. Web サーバを再起動します。
7. 共有メモリ セグメントを調整することにより Oracle iPlanet Web エージェントを最適化します。

エージェントが設定されたら、マシンを再起動するように要求される場合があります。

詳細情報

[Sun Java System サーバ構成に追加された設定 \(P. 211\)](#)

[共有メモリ セグメントの調整 \(P. 167\)](#)

第 7 章: Apache Web エージェントを設定する

このセクションには、以下のトピックが含まれています。

[Windows システムでの Apache Web エージェントの設定 \(P. 137\)](#)

[UNIX システムでの Apache Web エージェントの設定方法 \(P. 140\)](#)

[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 \(P. 144\)](#)

[Apache エージェント操作に LD_PRELOAD 変数を設定する \(P. 144\)](#)

[SuSE Linux 9 for zSeries 上の Apache エージェントに LD_ASSUME_KERNEL を設定する \(P. 145\)](#)

Windows システムでの Apache Web エージェントの設定

エージェントを設定する前に、エージェントがインストールされているシステムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

Apache Web エージェントを設定する方法

1. 必要に応じて、Web エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

2. すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、ホスト登録をスキップするオプションを選択し、[次へ]をクリックします。

トラステッド ホストを登録する場合は、プラットフォームのインストールの章に移動します。

3. [Web サーバの選択]ダイアログ ボックスで、Apache Web サーバのラジオ ボタンをオンにして[次へ]をクリックします。

4. [Apache Web サーバパス]ダイアログ ボックスで、Apache Web サーバルートを指定します。

IBM HTTP Server または Oracle サーバのような Apache ベースのサーバにエージェントをインストールしている場合は、Web エージェントはパスを認識しない可能性があります。この場合、設定ウィザードは、以下のオプションと共に[Apache Web サーバ障害]ダイアログ ボックスを表示します。

- Apache サーバルートを再入力します。
Apache Web サーバの場合はこのオプションを選択し、ルートパスを再入力します。
- 特定の設定パスを入力します。
Apache ベースの Web サーバ (IBM HTTP、HP Apache ベース、Oracle) を使用している場合は、このオプションを選択します。Web サーバルートへのフル設定パスを入力するようにプロンプトが表示されます。
- Apache Web サーバを使用していません。
Apache の設定をスキップし、エージェントの設定を続行するにはこのオプションを選択します。

[次へ]をクリックします。

5. サーバルートパスに続いて、使用している Apache のバージョンを指定します。以下のオプションから選択します。

- Apache バージョン 2.0

6. Web エージェントと共に設定する Web サーバ インスタンスを選択します。

すでにサーバを Web エージェントと共に設定しており、設定ウィザードを実行して追加の Web サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した Web サーバのリストを示します。

- a. 次のいずれかをオンにします。

上書き - サーバ インスタンスの設定を上書きします。

保存 - Web サーバの設定を保存します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、Web エージェントはこのサーバから削除されます。

- b. [次へ]をクリックします。

7. [エージェント設定オブジェクト]フィールドで、この Web サーバ インスタンスのエージェント設定オブジェクトの名前を入力します。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `ApacheDefaultSettings` を入力します。

8. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェント ファイルがインストールされます。

9. インストールが完了したら、[終了]をクリックします。

10. Web エージェントを以下のようにして有効にします。

- a. 以下の場所にある `WebAgent.conf` ファイルを開きます。

`Apache_home¥conf`

ここで、`Apache_home` は Apache Web サーバがインストールされている場所です。

- b. `EnableWebAgent` パラメータを `Yes` に設定します。

- c. ファイルを保存して閉じます。

11. Web サーバを再起動します。

Apache Web エージェントに設定ウィザードを実行すると、Web サーバの `httpd.conf` ファイル、およびライブラリパスが変更されます。

`httpd.conf` の変更を有効にするには、Web サーバを再起動する必要があります。

詳細情報

[Apache Web エージェントを使用する Web サーバへの設定変更 \(P. 219\)](#)

[Apache Web エージェントを設定する \(P. 137\)](#)

UNIX システムでの Apache Web エージェントの設定方法

以下の設定方法が UNIX システムの Web エージェントで使用できます。

- GUI モード
- コンソール モード
- 無人モード

注:

- IBM HTTP Web サーバ、HP Apache ベースの Web サーバ、Oracle HTTP Web サーバについては、インストールする必要があるエージェントは Apache Web エージェントです。Apache Web サーバについてのすべての情報はこれらの Web サーバにも適用されます。
- エージェントを設定する前に、システムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

詳細情報

[GUI またはコンソール モードを使用する Apache Web エージェントの設定 \(P. 141\)](#)

[無人モードで任意の Web エージェントを設定する方法 \(P. 157\)](#)

GUI またはコンソール モードを使用する Apache Web エージェントの設定

これらの手順は GUI およびコンソール モードを設定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。

- 対応する数を入力してオプションを選択するように指示される場合があります。たとえば、Apache Web サーバを選択するには、このサーバに対応する 1 を入力します。
- 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。
- 入力するパスワードはすべてクリア テキストで表示されます。この問題の回避策として、インストールを GUI または無人モードで実行します。

各モードのプロンプトにより、順を追ってプロセスをガイドします。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`

UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

Apache Web エージェントを設定する方法

1. 必要に応じて、設定ウィザードを開始します。
 - a. コンソール ウィンドウを開きます。
 - b. `web_agent_home/install_config_info` に移動します。
 - c. 以下のいずれかのコマンドを入力します。

GUI モード: `./<ファイル名>-wa-config.bin`

コンソール モード: `./<ファイル名>-wa-config.bin -i console`

2. すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、ホスト登録をスキップするオプションを選択し、[次へ]をクリックします。

トラステッド ホストを登録する場合は、プラットフォームのインストールの章に移動します。

3. [Web サーバの選択]ダイアログ ボックスで、Apache Web サーバのオプションを選択して[次へ]をクリックします。

4. [Apache Web サーバパス]ダイアログ ボックスで、Apache Web サーバルート(/opt/apache2 など)を指定します。[次へ]をクリックします。

IBM HTTP Server または Oracle サーバのような Apache ベースのサーバにエージェントをインストールしている場合は、Web エージェントはパスを認識しない可能性があります。この場合、設定ウィザードは、以下のオプションと共に[Apache Web サーバ障害]ダイアログ ボックスを表示します。

- Apache サーバルートを再入力します。
Apache Web サーバの場合はこのオプションを選択し、ルートパスを再入力します。
- 特定の設定パスを入力します。
Apache ベースの Web サーバ (IBM HTTP、HP Apache ベース、Oracle) を使用している場合は、このオプションを選択します。Web サーバルートへのフル設定パスを入力するようにプロンプトが表示されます。
- Apache Web サーバを使用していません。
Apache の設定をスキップし、エージェントの設定を続行するにはこのオプションを選択します。

[次へ]をクリックします。

5. サーバルートパスに続いて、使用している Apache のバージョンを指定します。以下のオプションから選択します。

- Apache バージョン 2.0

6. Web エージェントと共に設定する Web サーバ インスタンスを選択します。

すでにサーバを Web エージェントと共に設定しており、設定ウィザードを実行して追加の Web サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した Web サーバのリストを示します。

- a. 次のいずれかをオンにします。

上書き - サーバ インスタンスの設定を上書きします。

保存 - Web サーバの設定を保存します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、Web エージェントはこのサーバから削除されます。

- b. [次へ]をクリックします。

7. [エージェント設定オブジェクト]フィールドで、この Web サーバ インスタンスのエージェント設定オブジェクトの名前を入力します。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `ApacheDefaultSettings` を入力します。

8. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェント ファイルがインストールされます。

9. インストールが完了したら、[終了]をクリックします。

10. Web エージェントを以下のようにして有効にします。

- a. 以下の場所にある `WebAgent.conf` ファイルを開きます。

`Apache_home/conf`

- b. `EnableWebAgent` パラメータに `yes` を設定します。

- c. ファイルを保存して閉じます。

11. Web サーバを再起動します。

Apache Web エージェントに設定ウィザードを実行すると、Web サーバの `httpd.conf` ファイル、およびライブラリパスが変更されます。

`httpd.conf` の変更を有効にするには、Web サーバを再起動する必要があります。

12. UNIX システムの Apache については、共有メモリ セグメントを調整することにより、Apache Web エージェントを最適化します。

詳細情報

[Apache Web エージェントを使用する Web サーバへの設定変更](#) (P. 219)

[Apache Web エージェントを設定する](#) (P. 137)

[共有メモリ セグメントの調整](#) (P. 167)

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。

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方法

1. Apache と Oracle iPlanet サーバについては、サーバの構成にインストールされた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。

Apache エージェント操作に LD_PRELOAD 変数を設定する

別のプラットフォームで動作するには Apache Web エージェントに LD_PRELOAD 変数を定義する必要があります。

Domino 6.5.3/SuSe8 Linux システムで X.509 ベースの認証方式を使用するための LD_PRELOAD を設定する

Domino 6.5.3/SuSe8 Linux で任意の X.590 ベースの認証方式で保護されたリソースにアクセスすると、Domino Server はクラッシュして NSD を生成します。

この問題を解決するには、Domino Web Server を開始する前に以下の環境変数を設定します。

```
export LD_PRELOAD=/usr/lib/libstdc++-libc6.2-2.so.3
```

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。

第 8 章: Domino Web エージェントを設定する

このセクションには、以下のトピックが含まれています。

[Windows システムでの Domino Web エージェントの設定 \(P. 147\)](#)

[UNIX システムに Domino Web エージェントを設定する方法 \(P. 151\)](#)

Windows システムでの Domino Web エージェントの設定

Windows システムに Domino Web エージェントを設定するには、以下のタスクを実行します。

- Domino Web エージェント DLL の追加
- Web エージェント設定ウィザードの実行
- (オプション) CGI ディレクトリおよび CGI URL パス設定の設定
- (オプション) HTML フォーム認証方式を有効にするための別名設定の設定

Domino Web エージェント DLL の追加 (Windows)

Domino Web エージェントが正常に動作するには、DOMINOWebAgent.dll ファイルをフィルタ DLL に追加する必要があります。Web エージェント DLL は、リスト内の最初の DLL である必要があります。

Domino Web エージェント DLL を追加する方法

1. Lotus Notes を開きます。
2. [File]-[Database]-[Open]を選択します。
3. [Server]フィールドで、Web エージェントをインストールした Domino Server を選択します。
4. [Database]スクロール ボックスで、サーバのアドレス帳を選択します。
[Filename]フィールドに、names.nsf が表示されます。
5. [Open]をクリックします。
サーバのアドレス帳が開きます。
6. 左ペインで、[Server]フォルダを展開し、[All Server Documents]アイコンをダブルクリックします。
7. サーバを選択し、[Edit Server]をクリックします。
Domino サーバの管理コンソールが開きます。
8. [Internet Protocols]タブを選択します。
9. ウィンドウの DSAPI セクションで、[DSAPI filter file names]フィールドを検索し、Domino Web エージェント DLL へのフルパスを入力します。例：

```
C:\Program Files\CAwebagent\bin\DOMINOWebAgent.dll
```
- 注: このエントリはフィルタのリストで 1 番目にある必要があります。
10. [Save and Close]をクリックします。
11. Web サーバを再起動します。
エージェントが設定されたら、マシンを再起動するように要求される場合があります。

Windows の Domino Web エージェントでの設定ウィザードの実行

エージェントを設定する前に、エージェントがインストールされているシステムをトラステッド ホストとして登録することをお勧めします。ただし、これは後で登録することもできます。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

Windows インストールのデフォルトの場所: `C:¥Program Files¥CA¥webagent`

UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

Domino Web エージェントを設定する方法

1. 必要に応じて、Web エージェント設定ウィザードを開始します。

デフォルトの方法では、[スタート]-[プログラム]-[SiteMinder]-[Web エージェント設定ウィザード]を選択します。デフォルト以外の場所にウィザードショートカットを配置した場合、手順は異なります。

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

2. すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、ホスト登録をスキップするオプションを選択し、[次へ]をクリックします。

トラステッド ホストを登録する方法の詳細については、お使いのプラットフォームのインストールの章を参照してください。

3. [Web サーバの選択]ダイアログ ボックスで、Domino Web サーバのラジオ ボタンをオンにして[次へ]をクリックします。
4. [Domino Web サーバ パス]ダイアログ ボックスで、notes.ini ファイルの場所 (C:¥Lotus¥Domino¥notesdata など)を指定し、[次へ]をクリックします。

注: インストールでは、notes.ini ファイルに WebAgent.conf へのパスを自動的に書き込みます。

5. Web エージェントと共に設定する Web サーバ インスタンスを選択します。

すでにサーバを Web エージェントと共に設定しており、設定ウィザードを実行して追加の Web サーバ インスタンスを設定する場合、ウィザードには上書きするインスタンスを選択するためのダイアログ ボックスが表示されます。このダイアログ ボックスには、設定済みの Web サーバがリスト表示されます。

- a. 次のいずれかをオンにします。
 - [上書き] - サーバ インスタンスの既存の設定を新しい設定に置換します。
 - [保存] - 変更せずに、既存の Web サーバの設定を維持します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、Web エージェントはこのサーバから削除されます。

- b. [次へ]をクリックします。

6. [エージェント設定オブジェクト]フィールドで、この Web サーバ インスタンスのエージェント設定オブジェクトの名前を入力し、[次へ]をクリックします。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は `DominoDefaultSettings` を入力します。

7. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェントファイルがインストールされます。

8. インストールが完了したら、[終了]をクリックします。

9. Web エージェントを以下のようにして有効にします。

- a. Domino Web サーバルート ディレクトリをインストールした場所にある、`WebAgent.conf` ファイルを開きます。
- b. `EnableWebAgent` パラメータを[はい]に設定します。
- c. ファイルを保存します。

詳細情報

[Domino Web エージェントで必要とされるエージェント設定パラメータ \(P. 26\)](#)

[UNIX 上でトラステッド ホストとしてシステムを登録する \(P. 63\)](#)

Windows 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定(オプション)

Domino Web エージェントに適切な cgi-bin (ScriptAlias) 設定を設定し、Domino 管理者のサーバ設定用の [Internet Protocols] タブに移動し、以下の設定を設定します。

- CGI ディレクトリ: domino¥html¥cgi-bin
- CGI URL パス: /cgi-bin

HTML フォーム認証方式を有効にするための別名設定の設定(オプション)

Domino Web エージェントを設定して HTML フォーム認証方式をサポートするには、以下のタスクを実行します。

1. Domino ドキュメント ルート(¥domino¥html¥) ディレクトリに「siteminderagent」という名前のサブディレクトリを作成します。
2. 手順 1 に作成した siteminderagent ディレクトリに agent_home¥samples のサブディレクトリをすべてコピーします。

agent_home

SiteMinder Web エージェントのインストール パスを指定します。

3. X.509 クライアント証明書および HTML フォーム認証方式をサポートするために、手順 1 で作成した siteminderagent ディレクトリに「certooptional」という名前のディレクトリを追加で作成し、さらに、そこに agent_home¥samples のすべてのサブディレクトリをコピーします。

UNIX システムに Domino Web エージェントを設定する方法

Windows システムに Domino Web エージェントを設定するには、以下のタスクを実行します。

- Domino Web エージェント DLL の追加
- Web エージェント設定ウィザードの実行
- (オプション) CGI ディレクトリおよび CGI URL パス設定の設定
- (オプション) HTML フォーム認証方式を有効にするための別名設定の設定

Domino Web エージェント DLL の追加 (UNIX)

Domino Web エージェントが正常に動作するには、`dominowebagent.so` ライブラリをフィルタ DLL に追加する必要があります。このライブラリはリストの一番上に配置される必要があります。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

Domino Web エージェント DLL を追加する方法

1. Lotus Notes を開きます。
 2. [File]-[Database]-[Open]を選択します。
 3. [Server]フィールドで、Web エージェントをインストールした Domino Server を選択します。
 4. [Database]スクロール ボックスで、サーバのアドレス帳を選択します。
[Filename]フィールドに、`names.nsf` が表示されます。
 5. [Open]をクリックします。
サーバのアドレス帳が開きます。
 6. 左ペインで、[Server]フォルダを展開し、[All Server Documents] アイコンをダブルクリックします。
 7. サーバを選択し、[Edit Server]をクリックします。
Domino サーバの管理コンソールが開きます。
 8. [Internet Protocols]タブを選択します。
 9. ウィンドウの DSAPI セクションで、[DSAPI filter file names]フィールドを検索し、Domino Web エージェント ファイルへのフル パスを入力します。例:

```
web_agent_home>/bin/dominowebagent.so
```
- 注: このエントリはフィルタのリストで 1 番目にある必要があります。
10. [Save and Close]をクリックします。
 11. Web サーバを再起動します。

UNIX システムでの Domino Web エージェントの設定方法

以下の設定方法が UNIX システムの Web エージェントで使用できます。

- GUI モード
- コンソール モード
- 無人モード

GUI またはコンソール モードでの Domino Web エージェントの設定

これらの手順は GUI およびコンソール モードを設定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。

- 対応する数を入力してオプションを選択するように指示される場合があります。たとえば、Apache Web サーバを選択するには、このサーバに対応する 1 を入力します。
- 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。
- 入力するパスワードはすべてクリア テキストで表示されます。この問題の回避策として、インストールを GUI または無人モードで実行します。

各モードのプロンプトにより、順を追ってプロセスをガイドします。

1. 必要に応じて、設定ウィザードを開始します。
 - a. コンソール ウィンドウを開きます。
 - b. `web_agent_home/install_config_info` に移動します。
 - c. 以下のいずれかのコマンドを入力します。

GUI モード: `./<ファイル名>-wa-config.bin`

コンソール モード: `./<ファイル名>-wa-config.bin -i console`

注: インストールの直後に Web エージェントを設定することを選択した場合、SiteMinder は設定ウィザードを開始します。

2. すでにホスト登録が完了している場合は、次の手順にスキップします。そうでない場合は、ホスト登録をスキップするオプションを選択し、[次へ]をクリックします。

トラステッド ホストを登録する場合は、プラットフォームのインストールの章に移動します。

3. [Web サーバの選択]ダイアログ ボックスで、Domino Web サーバのラジオ ボタンをオンにして[次へ]をクリックします。
4. [Domino Web サーバ パス]ダイアログ ボックスで、notes.ini ファイルの場所 (/local/notesdata など)を指定し、[次へ]をクリックします。

注: インストールでは、notes.ini ファイルに WebAgent.conf へのパスを自動的に書き込みます。

5. Web エージェントと共に設定する Web サーバ インスタンスを選択します。
すでにサーバを Web エージェントと共に設定しており、設定ウィザードを実行して追加の Web サーバ インスタンスを設定する場合は、ウィザードに [Select One or More Instances to Overwrite]ダイアログ ボックスが表示されます。このダイアログ ボックスは、以前に設定した Web サーバのリストを示します。

- a. 次のいずれかをオンにします。

上書き - サーバ インスタンスの設定を上書きします。

保存 - Web サーバの設定を保存します。

重要: 以前に設定したサーバのチェック ボックスをオフにすると、Web エージェントはこのサーバから削除されます。

- b. [次へ]をクリックします。

6. [エージェント設定オブジェクト]フィールドで、この Web サーバ インスタンスのエージェント設定オブジェクトの名前を入力します。

この名前は、ポリシー サーバ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に一致する必要があります。たとえば、デフォルトを使用する場合は DominoDefaultSettings を入力します。

7. [Web サーバ設定の概要]ダイアログ ボックスで、構成設定が正しいことを確認し、[インストール]をクリックします。

Web エージェント ファイルがインストールされます。

8. インストールが完了したら、[終了]をクリックします。

9. Web エージェントを以下のようにして有効にします。

- a. Domino Web サーバ ルート ディレクトリにある、WebAgent.conf ファイルを開きます。

- b. EnableWebAgent パラメータを Yes に設定します。

- c. ファイルを保存します。

10. Domino Web サーバが AIX 上で実行する場合、以下の手順でエージェントを設定した後にランタイムリンクを有効にする必要があります(これを行う必要があるのは一度だけです)。

a. 以下のコマンドを実行します。

```
cd domino_server_path/lotus/notes/latest/ibmpow
/usr/bin/rtl_enable ./http -brtl
mv ./http ./http.orig
mv ./http.new ./http
```

b. Domino Web サーバを開始します。

UNIX 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定(オプション)

Domino Web エージェントに適切な `cgi-bin` (`ScriptAlias`) 設定を設定し、Domino 管理者のサーバ設定用の `[Internet Protocols]` タブに移動し、以下の設定を設定します。

- CGI ディレクトリ: `domino/html/cgi-bin`
- CGI URL パス: `/cgi-bin`

UNIX オペレーティング環境で HTML フォーム認証方式を有効にするための別名設定の設定(オプション)

Domino Web エージェントを設定して HTML フォーム認証方式をサポートするには、以下のタスクを実行します。

1. Domino ドキュメントルート (`/domino/html`) ディレクトリに「`siteminderagent`」という名前のサブディレクトリを作成します。
2. 手順 1 に作成した `siteminderagent` ディレクトリに `agent_home/samples` のサブディレクトリをすべてコピーします。

agent_home

SiteMinder Web エージェントのインストールパスを指定します。

3. X.509 クライアント証明書および HTML フォーム認証方式をサポートするために、手順 1 で作成した `siteminderagent` ディレクトリに「`certoptional`」という名前のディレクトリを追加で作成し、さらに、そこに `agent_home/samples` のすべてのサブディレクトリをコピーします。

第 9 章: すべての Web エージェントに利用可能な設定

このセクションには、以下のトピックが含まれています。

[無人モードで任意の Web エージェントを設定する方法 \(P. 157\)](#)

[共有秘密キーのロールオーバーに対する SmHost.conf ファイル アクセス許可の確認 \(P. 160\)](#)

[Web エージェントの再設定 \(P. 160\)](#)

[追加のエージェントコンポーネントをセットアップする方法 \(P. 161\)](#)

[ダイナミック ポリシー サーバクラスタ \(P. 162\)](#)

無人モードで任意の Web エージェントを設定する方法

1 つのシステムに Web エージェントをインストールしたら、エージェントの無人設定機能を使用して、他の Web サーバ上で Web エージェント設定を自動化することができます。無人設定では、ユーザの操作なしに Web エージェントを設定することができます。

無人モードで任意の Web エージェントを設定するには、以下のプロセスを使用します。

1. 無人設定を準備します。
2. 無人設定を実行します。

無人設定の準備

無人設定では、`ca-wa-installer.properties` ファイルを使用してネットワーク内のすべてのエージェントに Web エージェント設定のセットアップを伝達します。設定では、プロパティファイル内に設定パラメータを定義し、次に、ネットワーク内の任意の Web サーバにファイルをコピーして無人設定を実行します。

初期 Web エージェント インストールおよび設定を実行する場合、`ca-wa-installer.properties` ファイルは以下の場所にインストールされています。

`web_agent_home/install_config_info`

ファイルのデフォルトパラメータおよびパスは、初期 Web エージェントのインストールおよび設定の間に入力された情報を反映します。

`ca-wa-installer.properties` ファイルをシステムで利用可能にする方法

1. Web エージェントの初期インストールを実行します。
2. `ca-wa-installer.properties` ファイルを開き、必要に応じて設定パラメータを変更します。
3. ファイルを保存します。

無人設定の実行

無人設定を実行する前に、以下のタスクを完了する必要があります。

- 初期(無人) Web エージェント インストール
- 初期(無人) Web エージェント設定
- `ca-wa-installer.properties` ファイルの変更
このファイルを使用して Web エージェントの無人設定を実行します。
- 無人設定を実行するシステムでのインストール(有人または無人) このインストールは設定実行可能ファイルを利用可能にします。

Web エージェントの無人設定を実行する方法

1. Web エージェントがすでにインストールされているシステムの `web_agent_home/install_config_info` から無人設定を実行するシステムのローカル ディレクトリに `ca-wa-installer.properties` ファイルをコピーします。
2. コンソール ウィンドウを開いて、`web_agent_home/install_config_info` に移動します。

重要: Windows Server 2008 上で SiteMinder ユーティリティまたは実行可能ファイルを実行している場合は、管理者としてシステムにログインしている場合でも、管理者権限でコマンドライン ウィンドウを開きます。詳細については、お使いの 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。

注: 設定実行可能ファイルはこのディレクトリに残す必要があるため、`install_config_info` ディレクトリから無人設定を実行する必要があります。

3. 以下のコマンドを実行します。

```
agent_config_executable -f properties_file -i silent
```

たとえば、`install_config_info` ディレクトリにプロパティ ファイルをコピーした場合、コマンドは以下ようになります。

Windows:

```
ca-wa-config.exe -f ca-wa-installer.properties -i silent
```

UNIX の場合

```
ca-wa-config.bin -f ca-wa-installer.properties -i silent
```

`install_config_info` ディレクトリにプロパティ ファイルをコピーしない場合は、コマンドにこのファイルへのフル パスを指定します。ディレクトリ パスにスペースがある場合は、パス全体を引用符で囲みます。

設定が完了したら、コマンド プロンプトに戻ります。

4. `web_agent_home/install_config_info` ディレクトリにある `CA_SiteMinder_Web_Agent_version_InstallLog.log` ファイルを参照することにより、設定が正常に完了したことを確認します。このログ ファイルには、設定の結果が含まれています。

共有秘密キーのロールオーバーに対する SmHost.conf ファイル アクセス許可の確認

共有秘密キーのロールオーバーを有効にした場合、Web サーバプロセスを所有するユーザは SmHost.conf ファイルに書き込む許可が必要です。このユーザがこのファイルを変更できない場合、共有秘密キーのロールオーバーは更新できません。

たとえば、Oracle iPlanet および Apache Web サーバでは、ユーザ ディレクティブによって指定されたユーザは、SmHost.conf ファイルへの書き込み許可が必要です。SmHost.conf ファイルがユーザ 1 によって所有され、他のユーザが書き込み許可を持っていない場合、ユーザ 2 がサーバプロセスを所有する場合は、共有秘密キーのロールオーバーは更新できません。

Web エージェントの再設定

Web エージェントを以下の理由で再設定します。

- Web エージェントをアップグレードしており、現在、設定を更新する必要がある。
- 以前に Web エージェントに定義された構成設定を変更する必要がある。
- 全 Web エージェントをアンインストールすることなく、Web エージェントから構成設定を削除する必要がある (Web エージェントは後で再度設定する必要がある場合があります)。
- 設定済みのサーバと同じシステムにインストールされた別の Web サーバに Web エージェントを設定する。

任意のモードで Web エージェントを再設定するには、設定ウィザードを再実行します。エージェントの再設定には、追加の手順も、プロンプトもありません。

追加のエージェントコンポーネントをセットアップする方法

Web エージェント設定ウィザードでは、基本的なエージェント設定を順を追ってガイドします。ただし、ウィザードを使用せずに設定できるエージェントコンポーネントもあります。

すべての **SiteMinder Web** エージェントはリソースを保護し、フォーム認証情報コレクタ (FCC) および **SSL** 資格情報コレクタ (SCC) として動作し、シングルサインオンのための **Cookie** プロバイダとして機能します。**Web** エージェントは、これらの 1 つ以上のロールで同時に機能することができます。

インストールでは、フォーム認証情報コレクタとして動作するなど、これらの機能のいくつかは、自動的にセットアップされますが、**Cookie** プロバイダのような他の機能には追加の設定が必要です。

以下のようにして、任意の追加のコンポーネントをセットアップできます。

- フォーム認証情報コレクタとしてエージェントを設定する
フォーム認証情報コレクション用のライブラリおよびファイルは、インストール中に自動的にセットアップされます。
- **SSL** 資格情報コレクタとしてエージェントを設定する
設定ウィザードでの初期エージェント設定中にエージェントが **SSL** 資格情報コレクションを実行するかどうか指定します。
- 複数の **Cookie** ドメインのシングルサインオンに **Cookie** プロバイダとしてエージェントを設定する

Cookie プロバイダを使用すると、エージェントは複数の **Cookie** ドメイン環境にシングルサインオンを実装することができます。すべての **Web** エージェントは **Cookie** プロバイダとして動作することができますが、ドメイン内のすべての **Cookie** プロバイダは同じドメイン名を使用する必要があります。エージェントの構成に設定されている **Cookie** プロバイダ URL は、どのエージェントが **Cookie** プロバイダであるかを示します。**Cookie** プロバイダであるエージェントを特定した後で、そのエージェントの URL を入力することにより、シングルサインオン環境の他のすべてのエージェントが **Cookie** プロバイダをポイントするように設定します。

ダイナミック ポリシー サーバ クラスタ

SiteMinder の旧バージョンでは、特定のポリシー 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場合でも、Web エージェントによって自動的に検出されませんでした。Web 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。

SiteMinderr12.0 SP3 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個々のポリシー サーバの追加または削除が Web エージェントによって自動検出されます。

たとえば、Web 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

サーバ 192.168.2.103 のオペレーティング システムをアップグレードするためにこのサーバを削除する場合、ダイナミック ポリシー サーバ クラスタが有効になっていると、再起動しなくてもクラスタのメンバシップの変更が Web エージェントによって認識されます。

以下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 サーバの設定の変更 (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。

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方法

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
```

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:¥Program Files¥CA¥webagent`
 - UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
2. 以下のタスクのいずれかを実行します。
 - この Web エージェントが過去に一度もポリシー サーバのダイナミッククラスタに接続されたことがない場合、以下の 1 行をファイル内の任意の場所に作成します。

```
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。
 3. `SmHost.conf` ファイルを保存し、テキストエディタを閉じます。
 4. 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 10 章: Web エージェントの起動と停止

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 エージェントの有効化](#) (P. 165)

[Web エージェントの無効化](#) (P. 166)

Web エージェントの有効化

Web エージェントのパラメータを設定して、Web 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。

Web エージェントを有効にする方法

1. WebAgent.conf ファイルを開きます。
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. エージェント設定オブジェクトまたはローカル設定ファイル内の以下のパラメータのいずれかの設定を変更した場合は、Web サーバを再起動します。
 - AgentConfigObject
 - CacheAnonymous
 - HostConfFile (IIS 6.0 エージェントおよび Apache エージェントのみ)
 - MaxResourceCacheSize
 - MaxSessionCacheSize
 - PostPreservationFile
 - ResourceCacheTimeout

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およびポリシー サーバとの通信を停止する場合は、Web エージェントを無効にする必要があります。

Web エージェントを無効にする方法

1. WebAgent.conf ファイルを開きます。
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. エージェント設定オブジェクトまたはローカル設定ファイル内の以下のパラメータのいずれかの設定を変更した場合は、Web サーバを再起動します。
 - AgentConfigObject
 - CacheAnonymous
 - HostConfFile (IIS 6.0 エージェントおよび Apache エージェントのみ)
 - MaxResourceCacheSize
 - MaxSessionCacheSize
 - PostPreservationFile
 - ResourceCacheTimeout

Web エージェントが無効になります。

第 11 章: オペレーティング システムのチューニング

このセクションには、以下のトピックが含まれています。

[共有メモリ セグメントの調整 \(P. 167\)](#)

[Solaris 10 リソース管理を調整する方法 \(P. 170\)](#)

共有メモリ セグメントの調整

Solaris システムに Apache または Oracle iPlanet Web エージェントをインストールする場合、Web 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 システムの共有メモリ設定を調整する必要があります。オペレーティング システムの共有メモリセグメントを増やすことによって、Web 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 システム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 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 を使用している場合は、共有メモリセグメントを調整する必要がある場合があります。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 システムのドキュメントを参照してください。

共有メモリ セグメントを増加させる方法

1. ご使用のオペレーティング システム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、**semmnu** は、システムで実行する **Apache** の子プロセスまたは **Oracle iPlanet Web** サーバ プロセスの数よりも大きくする必要があります。 **Apache** サーバでは、この値は、**maxclients** 設定よりも **200** 以上上回る必要があります。 **Oracle iPlanet Web** サーバでは、この値は、**maxprocs** 設定よりも **200** 以上上回る必要があります。

例: (Solaris 9) 200

例: (Solaris 9) 400

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

Web エージェントのパフォーマンスを改善する必要がある場合は、プロジェクトレベルでリソース管理を調整することをお勧めします。

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するユーザ 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 12 章: パスワード サービス

このセクションには、以下のトピックが含まれています。

[パスワード サービスの実装 \(P. 171\)](#)

[JSP パスワード サービス用の環境をセットアップする方法 \(P. 172\)](#)

[IIS Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法 \(P. 173\)](#)

[UNIX オペレーティング環境で Oracle iPlanet Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法 \(P. 174\)](#)

パスワード サービスの実装

SiteMinder パスワード サービスでは、LDAP ユーザ ディレクトリまたは ODBC データベースを使用して、ユーザ パスワードを管理することができます。

以下のメカニズムをパスワード管理の実装に利用できます。

パスワード サービス CGI

(デフォルト) カスタマイズ可能な HTML フォームを使用してパスワード サービスを実装します。この実装は、`.template` 形式のように事前にカスタマイズされたパスワード サービスをサポートします。

FCC-based パスワード サービス

SiteMinder 形式を使用して、パスワード サービスを実装します。

注: 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。

パスワード サービス サブレット

Web サイトの必要を満たすためにカスタマイズできる標準的な JSP の形式を使用して、パスワード サービスを実装します。JSP 形式のパスワード サービスを使用するには、Web サーバおよびサブレット エンジンの両方を変更する必要があります。

注: サポートされているサブレット エンジンのリストについては、<http://www.ca.com/jp/support/> で「SiteMinder サポート マトリックス」を参照してください。

JSP パスワード サービス用の環境をセットアップする方法

JSP 形式のパスワード サービスを使用するには、以下の手順に従って、Web サーバおよびサーブレット エンジンの両方を変更する必要があります。

1. サーブレット エンジン クラスパスに以下のパスワード サービス JAR ファイルを追加する必要があります。

```
web_agent_home¥jpw¥jpw.jar  
web_agent_home¥Java¥servlet.jar  
web_agent_home¥Java¥jsafe.jar
```

2. 以下の行を追加することによって、サーブレット エンジンを読み出すファイルを更新して JSP パスワード サービス サーブレットを読み出します。

```
/siteminderagent/pwservlet/PSWDChangeServlet=PSWDChangeServlet
```

3. JSP パスワード サービス用のサーブレット エンジンを設定します。詳細については、サーブレット エンジンのマニュアルを参照してください。

注: サポートされているサーブレット エンジンのリストについては、<http://www.ca.com/jp/support/> で「SiteMinder サポート マトリックス」を参照してください。

IIS Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法

SiteMinder JSP ベースのパスワード サービス用の ServletExec サブレット エンジンを設定するには、以下の手順に従います。

注: ServletExec ドキュメントには [New Atlanta Web サイト](#) でアクセスできます。

1. ServletExec 管理インターフェースを開きます。
2. 仮想マシンのクラスパスに以下のアイテムを追加します。

`web_agent_home¥jpw¥jpw.jar`

`web_agent_home¥java¥jsafe.jar`

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:¥Program Files¥CA¥webagent`
 - UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
3. Web アプリケーションのトップレベル `web.xml` ファイルに以下の変更を行います。
 - a. 以下のサブレットを追加します。
 - サブレット名: `PSWDChangeServlet`
 - サブレット クラス: `PSWDChangeServlet`
 - b. 以下のサブレット マッピングを作成します。
 - URL パターン: `/siteminderagent/pwservlet/PSWDChangeServlet`
 - サブレット名: `PSWDChangeServlet`
 4. Web アプリケーションの各 `web.xml` ファイルに対して手順 3 を繰り返します。

UNIX オペレーティング環境で Oracle iPlanet Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法

UNIX オペレーティング環境で Oracle iPlanet Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定するには、以下の手順に従います。

1. Oracle iPlanet Web サーバを使用して SiteMinder Web エージェントが実行する Web サーバ インスタンスに以下の変更を行います。
 - a. 以下のレガシー サブレット属性を追加します。
 - サブレット名: PSWDChangeServlet
 - サブレットコード: PSWDChangeServletServlet
 - クラスパス: `web_agent_home/jpw/jpw.jar`

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
 - UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
- a. レガシー サブレットの Servlet Virtual Path Translation に以下の仮想パスを追加します。

```
/siteminderagent/pwservlet/PSWDChangeServlet and Servlet Name:
PSWDChangeServlet
```
 - b. Oracle iPlanet サブレット エンジンが無効にします。

2. ServletExec ASAPI ソフトウェアをインストールします。

3. テキスト エディタを使用して `ServletExec_installation_directory/se-instance_name/StartServletExec` ファイルを以下の変更によって更新します。

- a. `PORT="8888"` を検索して、Web サーバとの通信ポートを任意の空きポート(例: `PORT="7777"`)に変更します。
- b. `CLASSPATH` の末尾に以下のエントリを追加することにより、`CLASSPATH` 定義を拡張します。

```
web_agent_home/jpw/jpw.jar
```

```
web_agent_home/java/jsafe.jar
```

- c. 以下の行の後にディレクトリ エントリを追加することによりドキュメント ディレクトリ定義を拡張します。

```
$SENAME $HOMEDIR $MIMEFILE $DOCROOTDIR -port $PORT  
$SEOPTS -addl "/siteminderagent/jpw=web_agent_home/jpw"
```

注: エントリの最後に 2 つの引用符があります。

4. ServletExec Servlet エンジンを実行し、管理インターフェースを使用して以下の手順に従います。

注: ServletExec ドキュメントには [New Atlanta Web サイト](#) でアクセスできます。

- a. 以下のサーブレットを追加します。

- サーブレット名: PSWDChangeServlet
- サーブレット クラス: PSWDChangeServlet

- b. 以下のサーブレット エイリアスを追加します。

- alias: /siteminderagent/pwservlet/PSWDChangeServlet
- サーブレット名: PSWDChangeServlet

5. Web エージェントが実行する Oracle iPlanet Web サーバの magnus.conf ファイルに以下の変更を行います。

- a. 以下の行については、すでに定義されているエージェントシステムのアドレスと一致するように IP アドレスとポート番号を変更します。

```
Init fn="ServletExecInit" instance_name.instances="IP_address:7777"
```

6. Web エージェントが実行する Oracle iPlanet サーバ インスタンスの obj.conf ファイルに以下のエントリを追加します。

```
NameTrans fn="assign-name" from="/siteminderagent/pwservlet/PSWDChangeServlet"  
name="instance_name"
```

以下のブロックにエントリを挿入します。

```
<Object name="default">
```

```
AuthTrans fn="SiteMinderAgent"
```

```
NameTrans fn="assign-name" from="/servlet/*" name="instance_name"
```

```
NameTrans fn="assign-name" from="/siteminderagent/pwservlet/PSWDChangeServlet"  
name="instance_name"
```

```
-----
```

```
-----
```

```
</Object>
```

7. Oracle iPlanet Web サーバを再起動し、次に、ServletExec Servlet エンジンを実行します。

第 13 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 \(P. 177\)](#)

[Web エージェントをアンインストールする前に JRE を PATH 変数に設定する \(P. 178\)](#)

[Windows システムからの Web エージェントのアンインストール \(P. 179\)](#)

[UNIX システムからの Web エージェントのアンインストール \(P. 181\)](#)

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。
- アンインストールに必要であるため、JRE が Web エージェントシステムにインストールされることを確認します。サポートされているバージョンについては、[テクニカル サポート](#)の「SiteMinder r12.0 SP3 プラットフォーム マトリックス」を参照してください。

Web エージェントをアンインストールする前に JRE を PATH 変数に設定する

Windows および UNIX システムで、Web エージェントをアンインストールする場合、JRE が PATH 変数にあることを確認します。ない場合は、アンインストールプログラムが停止し、以下のエラー メッセージのうちの 1 つが発行されます。

- "Could not find a valid Java virtual machine to load. You need to reinstall a supported Java virtual machine."
- "No Java virtual machine could be found from your PATH environment variable. You must install a VM prior to running this program."

JRE を PATH 変数に設定する方法

Windows 側

- [コントロール パネル]に移動します。
- [システム]をダブルクリックします。
- [環境変数]ダイアログ ボックスで、PATH システム変数への JRE の場所を追加します。

例: C:\j2sdk1.5.0_01\jre\bin

Solaris の場合

以下の 2 つのコマンドを実行します。

- PATH=\$PATH:JRE/bin

ここで、JRE は JRE の場所です。

例: /usr/bin/j2sdk1.5.0_01/jre

- export PATH

Windows システムからの Web エージェントのアンインストール

アンインストールする前に、バックアップとしてレジストリ設定および Web エージェント構成設定のコピーを作成することができます。

Windows システムから Web エージェントをアンインストールする方法

1. Web サーバを停止します。
2. [スタート]-[コントロール パネル]-[プログラムの追加と削除]をクリックします。
[プログラムの追加と削除]ダイアログ ボックスが表示されます。
3. プログラムリストをスクロールし、CA SiteMinder Web エージェントのバージョンを選択します。
4. [変更と削除]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。
5. [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。
6. アンインストールが完了すると、ダイアログ ボックスが表示され、システムを今すぐ再起動するか、または後で再起動するかを選択し、[終了]をクリックします。
7. Web サーバを再起動します。
Web エージェントはアンインストールされます。

Windows システムからのドキュメントのアンインストール

ドキュメント アンインストール プログラムを実行すると、ca_documents ディレクトリからすべての製品のマニュアルが削除されます。

ドキュメントをアンインストールする方法

1. Web サーバを停止します。
2. [コントロール パネル]を開きます。
3. [プログラムの追加と削除]を選択します。
4. プログラムリストをスクロールし、Web エージェントの CA SiteMinder ドキュメント バージョンを選択します。
5. [変更と削除]をクリックします。
6. ダイアログ ボックスの情報を確認してアンインストールを確認します。
7. [アンインストール]をクリックします。
ドキュメントが削除されます。
8. [終了]をクリックして、インストーラを終了します。

UNIX システムからの Web 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 モードをアンインストール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。

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。

- 対応する数を入力してオプションを選択するように指示される場合があります。
- 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 を押して次のプロセスに進みます。各モードのプロンプトにより、順を追ってプロセスをガイドします。

注: アンインストールする前に、バックアップとして Web エージェント構成設定のコピーを作成することができます。

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. PATH 環境変数に JRE を指定して Web エージェントをアンインストールします。Java 仮想マシンが見つからなかったとのエラー メッセージを受け取った場合は、PATH 変数に JRE を以下のように追加します。

```
PATH=/jre_home/bin: ${PATH}
```

```
export PATH
```

jre_home は JRE の場所です。

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へ移動します:
`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 プログラムが開始します。

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. オプションで、Apache Web エージェントをアンインストールする場合は、設定ウィザードで追加された `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。

注: Oracle iPlanet Web サーバの場合、`obj.conf`、`magnus.conf`、および `mime.types` ファイルは Web エージェントをインストールする前に元の設定にリストアされます。

UNIX システムからのドキュメ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 モードをアンインストール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。

- 対応する数を入力してオプションを選択するように指示される場合があります。
- 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 を押して次のプロセスに進みます。各モードのプロンプトにより、順を追ってプロセスをガイドします。

UNIX システムからドキュメントをアンインストールする方法

1. 以下のディレクトリに移動します。

```
documentation_home/install_config_info/ca-wa-doc-uninstall
```

2. 以下のいずれかのコマンドを入力します。

- GUI モード: `./uninstall`
- コンソール モード: `./uninstall -i console`

アンインストール プログラムが開始し、アンインストールを確認するためにダイアログ ボックスが表示されます。

3. [アンインストール]をクリックします。

ドキュメントが削除されます。

4. [終了]をクリックして、インストーラを終了します。

ドキュメントを再インストールするには、製品の該当するドキュメントプログラムを実行します。

第 14 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。

[エージェントのスタートアップおよびシャットダウンの問題\(フレームワーク エージェントのみ\)](#) (P. 185)

[接続およびトラステッド ホスト登録の問題](#) (P. 188)

[一般的なインストールの問題](#) (P. 191)

[その他の問題](#) (P. 193)

[Oracle iPlanet Web エージェントの問題](#) (P. 195)

[Apache Web エージェントの問題](#) (P. 197)

[Domino Web エージェントの問題](#) (P. 198)

エージェントのスタートアップおよびシャットダウンの問題(フレームワーク エージェントのみ)

Web エージェントがインストールした後で開始しない、またはシャットダウンできない場合は、以下のエラー ログを確認します。

- Windows で、イベントビューアのアプリケーション ログを確認します。
- UNIX で、メッセージはサーバの標準エラー処理によって処理されます。Apache 2.0 では、エラーは Web サーバ エラー ログに書き込まれます。
- Windows または UNIX で、Low Level Agent Worker Process (LLAWP) を実行して問題を分離します。

LLAWP がすでに実行しているかまたはログ メッセージが適切なログ ファイルに書き込まれていないためにエージェントが起動しない

UNIX で有効

症状:

以下のような問題が 1 つ以上あります。

- LLAWP プロセスがすでに実行されているため、Web エージェントが開始しません。
- Web エージェントは開始しますが、ログ メッセージが 2 番目のエージェント インスタンスのログ ファイルに書き込まれます。

解決方法:

この問題は、同じコンピュータ上の複数のディスクが同じマウントポイントを使用する場合に発生する場合があります。Web エージェントは、ディレクトリの inode を使用してシステムリソースを割り当てます。また、inode が同じである場合は、リソースの衝突が発生してエラーが表示される結果になります。この問題を修正するには、以下の手順に従います。

1. Web サーバに新しいサブディレクトリを作成します(これにより、一意の inode が作成されます)。
2. Web エージェントの `ServerPath` パラメータに表示されているパスを変更してそれが新しいサブディレクトリをポイントするようにします。

注: 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。

LLAWP でのエージェント スタートアップ/シャットダウンのトラブルシューティング

エージェントが正常に開始またはシャットダウンしない場合は、コマンドラインから **Low Level Agent Worker Process (LLAWP)** を実行できます。

LLAWP はプロセス間のエージェント管理を処理します。IIS 6.0 では、Web エージェントが最初のリクエストを受け取った後、LLAWP は開始されます。Apache 2.0 では、Apache Web サーバが開始すると、LLAWP プロセスは自動的に開始されます。

コマンドラインから LLAWP を実行することによって、診断プロセスから Web サーバを除去します。これにより、Web エージェントの問題を分離します。エラーメッセージは、Windows のイベントログ、または UNIX システムのコンソールに書き込まれます。

LLAWP のシャットダウン

Web サーバをシャットダウンするときに、LLAWP プロセスが正常にシャットダウンしない場合は、コマンドラインから LLAWP をシャットダウンします。これにより、WebAgent.conf ファイルに関連付けられた、実行するワーカ プロセスがシャットダウンします。

LLAWP をシャットダウンするには、以下の構文のコマンドを使用します。

```
LLAWP path_to_WebAgent.conf -web_server_type -shutdown
```

以下に例を示します。

```
LLAWP /usr/apache/conf/WebAgent.conf -APACHE20 -shutdown
```

注: スペースを含んでいる環境設定ファイル名およびバージョン文字列は、"value with spaces" のように引用符で囲む必要があります。

LLAWP プロセスは、シャットダウンするまでに数秒かかります。

コマンドラインを使用して **kill -9** コマンドの代わりに LLAWP をシャットダウンします。これにより、プロセスは、Web エージェントが使用した共有システムリソースをクリーンアップします。

Web エージェントの開始とシャットダウンの問題 (IBM HTTP Server)

Web エージェントがインストールした後で開始しない、またはシャットダウンできない場合は、以下のエラー ログを確認します。

- Windows で、イベントビューアのアプリケーション ログを確認します。
- UNIX で、メッセージはサーバの標準エラー処理によって処理されます。

ホスト設定ファイルに書き込み許可がない

症状:

Web エージェント ログに以下のエラーが表示されます。

```
Siteminder Web Agent not having write permissions on host configuration file.
```

解決方法:

Web サーバが操作するアカウントに `SmHost.conf` ファイルへの書き込み許可があることを確認します。たとえば IIS Web サーバで、`NETWORK SERVICE` アカウントに `SmHost.conf` ファイルへの書き込み許可があることを確認します。

接続およびトラステッド ホスト登録の問題

このセクションでは、トラステッド ホスト登録に関連するトラブルシューティング情報について説明します。

トラステッド ホストの登録に失敗

症状:

トラステッド ホストを登録することができません。

解決方法:

以下の点をチェックしてください。

- ポリシー サーバがターゲット サーバにインストールおよび設定されていて、サーバの IP アドレスが正しく、ポリシー サーバが実行されていることを確認します。
- SiteMinder 管理者名およびパスワードをチェックして、これらが正しいことを確認します。
- エージェントのインストールと設定中に、指定されたホスト設定オブジェクトおよびエージェント設定オブジェクトがポリシー サーバに定義されていることを確認します。
- 既存のトラステッド ホストによってすでに使用されているトラステッド ホストの名前を使用している可能性があります。トラステッド ホストに一意の名前を使用して再登録します。

トラステッド ホストからポリシー サーバへの接続がない

症状:

トラステッド ホストがポリシー サーバに接続することができません。

解決方法:

以下の手順を実行します。

- `WebAgent.conf` ファイルの `EnableWebAgent` パラメータが[はい]に設定されることを確認します。
- `web_agent_home/config` の `SmHost.conf` ファイルを確認します。このファイルの存在は、トラステッド ホストが正常に登録されたことを示します。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`
- エージェントがインストールされているホストが、トラステッド ホストとして登録されていることを確認します。
- エージェント設定オブジェクトに `DefaultAgentName` が指定されていることを確認します。また、必要最小パラメータが特定の Web サーバに設定されていることを確認します。
- ポリシー サーバが実行していることを確認します。

ホストは登録されたが `SmHost.conf` ファイルが削除されている

症状:

トラステッド ホストは登録されているのに、`SmHost.conf` ファイルが削除されています。

解決方法:

管理 UI で、ファイルが削除されたホスト名に対応するトラステッド ホストオブジェクトを削除します。smregghost ツールを使用して、ホストを再登録します。

一般的なインストールの問題

このセクションでは、インストールに関連するトラブルシューティング情報について説明します。

同じシステムでの複数のインストール中に1つのインストールがハングする

症状:

同じシステム上で複数のインストールを同時に実行していて、インストールがハングアップします。

解決方法:

リストされている順序で以下のタスクを試みます。

1. システムを再起動してから、インストールを再試行してください。
2. ZeroG レジストリファイルの名前を変更し、インストールを再試行します。レジストリファイルは以下の場所にあります。

- Windows: C:¥Program Files¥ZeroG Registry¥com.zerog.registry.xml
- UNIX: \$HOME/.com.zerog.registry.xml or /var/.com.zerog.registry.xml

インストールが行われている間、レジストリファイルはロックされるため、複数のインストールが同時に実行している場合、インストールがハングアップするため、このファイルに書き込むことができません。

インストール エラー ログの場所

症状:

インストール中に失敗した内容を参照します。

解決方法:

`web_agent_home/install_config_info` にある `ca-wa-details.log` ファイルを参照してください。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: C:¥Program Files¥CA¥webagent
- UNIX インストールのデフォルトの場所: /opt/ca/webagent

コントロールパネルの[プログラムの追加と削除]に Web エージェントが表示されない

症状:

SiteMinder Web エージェントがリストに表示されないので、コントロールパネルの[プログラムの追加と削除]リストから Web エージェントをアンインストールすることができません。

解決方法:

エージェントを以下のようにして削除します。

1. レジストリ エディタを開きます。
2. HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥Uninstall¥SiteMinder¥WebAgent に移動します。
3. UninstallString エントリ全体を強調表示し、それをコピーします。
4. DOS ウィンドウを開いて、DOS プロンプトでウィンドウに UninstallString を貼り付けます。
5. ENTER キーを押します。

エージェントはアンインストールされます。

アップグレード中にエラー メッセージが表示される

Windows、UNIX で有効

症状:

アップグレード中に以下のエラーが表示されます。

ComponentMoveData エラー -115

解決方法:

以下の手順を実行します。

1. [OK]をクリックしてエラー メッセージを閉じます。
2. ポリシーサーバー管理コンソールを起動します。
3. コンソールから、ポリシー サーバを停止します。
4. 管理コンソールを閉じます。
5. アップグレードを再度実行すると、このエラー メッセージは表示されなくなります。

IIS Web エージェント設定時のメタベース エラー

症状:

IIS サーバに Web エージェントを設定しようとする、以下のようなメタベース エラー メッセージが表示されます。

メタベースにエンTRIESを追加できません : siteminderagent ==

解決方法:

次の手順に従います。

1. 再度設定ウィザードを実行し、IIS Web サーバを(そのチェック ボックスをオフにすることによって)削除します。

以下のようなエラーが表示されます。

メタベースからENTRIESを削除できません : siteminderagent ==

2. メタベースを編集することにより、デフォルト以外の Web サイトを使用するように IIS Web サーバを設定します。

その他の問題

このセクションでは、その他の問題に関連するトラブルシューティング情報について説明します。

Netscape ブラウザで PDF が開きません

UNIX で有効

症状:

Netscape ブラウザを使用して、UNIX システムのオンライン マニュアル インデックス HTML ページから PDF ファイルを開くことができません。

解決方法:

doc_index.htm ページでリンクをクリックした後で .pdf ファイルが開かない場合は、Netscape Navigator でヘルパアプリケーションとして Acrobat Reader を設定します。このオプションを設定すると、.pdf ファイルを表示することを要求するたびに、Netscape は自動的に Acrobat Reader を起動します。

ヘルパアプリケーションとして Acrobat Reader を設定する方法

1. Navigator で、[Edit]-[Preferences]に移動します。
2. [Netscape Preferences]ダイアログ ボックスで、[Navigator]-[Applications]を選択します。
3. [Applications]-[Specify helper applications for different file types]の下で、[Portable Document Format]を選択し、[Edit]をクリックします。
4. [Netscape Applications]ダイアログ ボックスで、[Applications]を選択し、それを以下のように設定します。

`Acrobat_Reader_home/bin/acroread %s`

たとえば、デフォルトの場所に Acrobat Reader をインストールした場合は、この値を以下に設定します。

`/usr/local/Acrobat4/bin/acroread %s`

5. [OK]をクリックしてこれらのダイアログ ボックスを閉じます。

このオプションを設定した後で、Navigator は Acrobat Reader を起動し、/tmp ディレクトリに .pdf ファイルを開きます。

Adobe Acrobat Reader が Windows システムにインストールされない

Windows で有効

症状:

Windows システムに Adobe Acrobat Reader をインストールすることができません。

解決方法:

ポリシー サーバが実行している間に、Acrobat Reader インストールプログラムがハングアップする場合は、ポリシー サーバ管理コンソールを使用してサーバを停止すると、インストールプログラムが開始されるはずですが。

Oracle iPlanet Web エージェントの問題

このセクションでは、Oracle iPlanet Web エージェントに関連するトラブルシューティング情報について説明します。

Web サーバは起動するのに Web エージェントが有効にならない

症状:

Web サーバが開始したのに Web エージェントが無効です。

解決方法:

WebAgent.conf ファイルを開いて、EnableWebAgent パラメータを[はい]に設定します。

Web サーバが開始すると smget エラー メッセージが表示される

Oracle iPlanet Web サーバで有効

症状:

Web サーバを開始すると、

「shmget failed」というメッセージが表示されます。大きすぎるキャッシュを作成しようとし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。

詳細情報:

[共有メモリ セグメントの調整 \(P. 167\)](#)

[Solaris 10 リソース管理を調整する方法 \(P. 170\)](#)

再設定された Web エージェントが動作しない

Oracle iPlanet Web サーバで有効

症状:

Web エージェント設定の変更は obj.conf ファイルにありません。Web エージェントは操作できません。

解決方法:

エージェント設定によって行われた obj.conf への変更が適用される前に、Oracle iPlanet 管理コンソールを使用してサーバの変更が行われました。Web エージェントを再設定します。

Sun Java System Web サーバがランタイムで失敗する

症状:

Oracle iPlanet Web サーバがランタイムで失敗します。

解決方法:

Oracle iPlanet サーバの `magnus.conf` ファイルの `StackSize` 設定を 256 KB の値に増やします。`magnus.conf` ファイルは以下に配置されています。

`Sun_Java_System_home/web_server_instance/config`

詳細情報

[共有メモリセグメントの調整](#) (P. 167)

Apache Web エージェントの問題

このセクションでは、Apache Web エージェントのトラブルシューティング情報をリストします。

Apache サーバは起動するが Web エージェントが有効にならない

症状:

Web サーバが開始したのに Apache エージェントが無効です。

解決方法:

以下のタスクを実行します。

- `WebAgent.conf` ファイルを開いて `EnableWebAgent` を [はい] に設定します。
- `mod_so` Apache のモジュールを含めるために Apache Web サーバをコンパイルします。
- `httpd.conf` ファイルを変更して `mod_sm SiteMinder` エージェント モジュールをロードできるようにします。
- `SmlInitFile` エントリを使用して、エージェントを初期化するために `httpd.conf` ファイルを変更します。
- HP-UX システムの Apache エージェントでは、`httpd.conf` ファイルを変更して `mod-hpaCCso.c SiteMinder` エージェント モジュールを追加します。

起動時に Apache サーバに shmget エラーが表示される

症状:

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

症状:

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

症状:

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。

Domino Web エージェントの問題

このセクションでは、Domino Web エージェントに関連するトラブルシューティング情報について説明します。

Domino Web エージェントが有効になっていないのに Web サーバが開始する

Domino で有効

症状:

Web サーバが開始したのに Domino Web エージェントが無効です。

解決方法:

以下の手順を実行します。

- WebAgent.conf ファイルで、EnableWebAgent パラメータを[はい]に設定します。
- DOMINOWebAgent.dll ファイルがフィルタ DLL に追加されたことを確認します。Web エージェント DLL は、リスト内の最初の DLL である必要があります。

ローカル設定モードの使用時に Domino エージェントが初期化できない

Domino で有効

症状:

Domino エージェントはローカル設定モードで初期化することができません。

解決方法:

WebAgent.conf ファイルへのフルパスが notes.ini ファイルに追加されていることを確認します。

付録 A: 無人インストール

このセクションには、以下のトピックが含まれています。

[ca-wa-installer.properties ファイル \(P. 201\)](#)

[一般情報の変更 \(P. 202\)](#)

[トラステッド ホストの登録 \(P. 202\)](#)

[トラステッド ホスト登録用のポリシー サーバの識別 \(P. 203\)](#)

[ホスト設定ファイルの指定 \(P. 203\)](#)

[設定用の Web サーバの選択 \(P. 204\)](#)

[Web サーバを再起動するように設定する \(Windows のみ\) \(P. 209\)](#)

[トラステッド ホスト名およびホスト設定オブジェクトの名前付け \(P. 209\)](#)

ca-wa-installer.properties ファイル

ca-wa-installer.properties ファイルは Web エージェントのインストールおよび設定中に生成されます。これには、パラメータ、パス、およびインストールおよび設定中に入力されたパスワードが含まれています。

無人のインストールと設定中に、このプロパティファイルは、GUI またはコンソール モードのインストールでエンド ユーザによって入力される設定を提供します。デフォルトでは、ca-wa-installer.properties ファイルには、初期インストールからの設定が含まれています。デフォルト プロパティファイルを使用して同じ設定のインストールを実行したり、環境に合わせて変更するテンプレートとしてファイルを使用することができます。

無人インストールは、初期 GUI またはコンソール モード Web エージェント インストールからの値で最初に設定されたプロパティファイルを使用します。そのため、Web エージェントを最初にインストールしたシステムと同じプラットフォーム および Web サーバ イメージのシステムでのみ無人インストールを実行できます。たとえば、エージェントを Oracle iPlanet Web サーバの Solaris システムにインストールし、プロパティファイルを使用して Apache Web サーバの Linux システムで無人インストールを実行することはできません。

一般情報の変更

プロパティファイルの全般情報セクションでは、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
USER_INSTALL_DIR	無人インストールが Web エージェントを配置する場所。以下に例を示します。 <code>C:\Program Files\ca\webagent</code>
USER_SHORTCUTS	インストールが設定ウィザードにショートカットを配置する場所。以下に例を示します。 <code>C:\Documents and Settings\jdoe\Start Menu\Programs</code>

トラステッドホストの登録

プロパティファイルのトラステッドホスト登録セクションでは、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
HOST_REGISTRATION_YES	インストールでトラステッドホスト登録プロセスを行うかどうかを示します。例を以下に示します。 <code>HOST_REGISTRATION_YES=1</code>
ADMIN_REG_NAME	トラステッドホストを登録する権限を持つ管理者の名前。例を以下に示します。 <code>ADMIN_REG_NAME=siteminder</code>
ADMIN_REG_PASSWORD	トラステッドホストを登録する権限を持つ管理者のパスワード。この値はインストールプログラムによって暗号化されます。例を以下に示します。 <code>ADMIN_REG_PASSWORD=ENC:nGDaSDy1H7qZqcdbkJKPEQ</code> パスワードを変更するには、エージェントを再設定するか、またはクリアテキストに新しいパスワードを入力することによりこのパラメータを変更します。

パラメータ	説明とサンプル値
SHARED_SECRET_ROLLOVER_YES	共有秘密キーのロールオーバーを有効にします。これは、トラステッドホストとポリシー サーバの間の通信を暗号化する秘密キーを定期的に変更します。 デフォルトは 0 です。このパラメータを 1 に設定して共有秘密キーのロールオーバーを有効にします。例を以下に示します。 SHARED_SECRET_ROLLOVER_YES=1

トラステッド ホスト登録用のポリシー サーバの識別

トラステッド ホスト登録用のポリシー サーバをリストするセクションで、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
IP_ADDRESS_STRING	トラステッド ホストを登録するポリシー サーバの IP アドレスを指定します。フェールオーバー用の複数のブートストラップ サーバを保有するには、複数のアドレスをカンマで区切って指定できます。例を以下に示します。 IP_ADDRESS_STRING=111.11.1.11, 122.123.2.34

ホスト設定ファイルの指定

[ホスト設定ファイルの場所]セクションで、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
SM_HOST_FILENAME	ホスト設定ファイル、SmHost.conf の名前を付けます。例を以下に示します。 SM_HOST_FILENAME=SmHost.conf
SM_HOST_DIR	SmHost.conf ファイルがインストールされているディレクトリを識別します。デフォルトの例を以下に示します。 SM_HOST_DIR=C:\Program Files\ca\webagent\config

設定用の Web サーバの選択

プロパティファイルのトラステッド ホスト登録セクションでは、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
APACHE_SELECTED APACHE_WEBSERVER_ROOT	設定する Apache Web サーバ、およびサーバのルート ディレクトリを示します。たとえば UNIX システムの場合は以下のように指定します。 APACHE_SELECTED=0 APACHE_WEBSERVER_ROOT=/export/agent5qa/apache
IPLANET_SELECTED IPLANET_WEBSERVER_ROOT	UNIX システムの場合 設定する Oracle iPlanet Web サーバ、およびサーバのルート ディレクトリを示します。たとえば UNIX システムの場合は以下のように指定します。 IPLANET_SELECTED=1 IPLANET_WEBSERVER_ROOT=/export/agent5qa/sunonewebserver
DOMINO_SELECTED DOMINO_WEBSERVER_ROOT	UNIX システムの場合 設定する Apache Web サーバ、およびサーバのルート ディレクトリを示します。たとえば UNIX システムの場合は以下のように指定します。 DOMINO_SELECTED=0 DOMINO_WEBSERVER_ROOT=

パラメータ	説明とサンプル値
WEB_SERVER_INFO	<p data-bbox="599 352 1430 495">WEB_SERVER_INFO 設定には、SiteMinder Web エージェントで設定された Web サーバに関する情報が含まれています。この設定をファイルで編集するか、または Web エージェント設定を再実行することにより、該当する値を使用してこの文字列を再生成することができます。</p> <p data-bbox="599 512 1430 617">WEB_SERVER_INFO エントリはセミコロンによって区切られた、Web サーバのセットで構成されます。各 Web サーバは、カンマ区切り値で構成されます。</p> <p data-bbox="599 634 1430 814">重要: WEB_SERVER_INFO 設定は、同じマシンであっても、ある Web サーバから別の Web サーバに変更できますが、ユーザ自身の責任で設定を変更できます。値を変更するときに間違えると、エージェントインストーラが失敗する可能性があるか、または不適当なデータでエージェントが設定される場合があります。</p> <p data-bbox="599 831 1146 858">WEB_SERVER_INFO 設定は以下のとおりです。</p> <p data-bbox="599 875 1430 1079">WEB_SERVER_INFO=<i>server_instance,web_server_config_dir,web_server_listing,service_name,web_server_type,web_server_version,web_server_path,empty_string,empty_string,selected_web_server,existing_server_config,preserve_web_server,document_selection,OneView_Monitor_config,confirm_web_server_config,advanced_auth_scheme,agent_config_obj</i></p>

詳細情報

[WEB_SERVER_INFO 変数](#) (P. 206)

WEB_SERVER_INFO 変数

WEB_SERVER_INFO 変数およびデフォルト値は以下のとおりです。

server_instance

Web サーバ インスタンスを示します。

例: https-server1

web_server_config_dir

Web サーバの構成ディレクトリへのパスを示します。

例: /usr/iplanet/servers/https-server1/config

web_server_listing

設定中に設定する利用可能な Web サーバのリストに Web サーバが表示される方法を示します。

例: https-server1 (Oracle iPlanet 6.0)

service_name

Web サーバ サービス名を示します。

例: https-server1

web_server_type

Web サーバのタイプを示します。以下のタイプから選択します。

- apache
- domino
- IIS
- iplanet
- sunone

Oracle iPlanet Web サーバには、iplanet または sunone を使用します。

例: sunone

web_server_version

Web サーバ バージョンを示します。

例: 6.0

web_server_path

web_server_instance ルートへのパスを示します。

例: /usr/iplanet/servers/https-server1

web_agent_operating_system

Web エージェントが使用するオペレーティング システムのタイプを示します。

制限: Windows、Unix

例: Windows

empty_string

将来使用できるように空の文字列が保存されることを示します。

例: +EMPTYSTR+

selected_web_server

選択された Web サーバがエージェントで設定する必要があるかどうかを示します。

制限: 1 (はい) または 0 (いいえ)

existing_server_config

前の Web サーバ設定は、既存のエージェント設定があるかどうかを示します。

制限: 1 (はい) または 0 (いいえ)

preserve_web_server

Web エージェントでの指定された Web サーバの設定を新しい設定で上書きするか保存するかを示します。

制限: 1 (保存) または 0 (上書き)

document_selection

ポリシー サーバのみに使用されます。Web エージェントはこのエントリを無視します。デフォルトを受け入れます。

制限: 1 (はい) または 0 (いいえ)

OneView_Monitor_config

ポリシー サーバのみに使用されます。Web エージェントはこのエントリを無視します。デフォルトを受け入れます。

制限: 1 (はい) または 0 (いいえ)

confirm_web_server_config

選択された Web サーバがエージェントで設定する必要があるかどうかを確認します。

制限: 1 (はい) または 0 (いいえ)

advanced_auth_scheme

使用された高度な認証方式(ある場合)を指定します。以下のいずれかのオプションを選択します。

- SSL を介した HTTP 基本
- X509 クライアント証明書
- X509 クライアント証明書および HTTP 基本
- X509 クライアント証明書または HTTP 基本
- X509 クライアント証明書またはフォーム
- X509 クライアント証明書およびフォーム
- 高度な認証なし

agent_config_object

使用するエージェント設定オブジェクトを示します。

例: `iplanetdefaultsettings`

以下にファイルの例を示します。

```
WEB_SERVER_INFO=;https-server1,/usr/iplanet/servers/https-server1/config,https-server1 (iPlanet
6.0),https-server1,iplanet,6.0,/usr/iplanet/servers/https-host,Unix,+EMPTYSTR+,1,
0,1,0,0,1,HTTP Basic over
SSL,agent1,0,undefined,ENC:6f1I5TLVEpuSBHpf4GrASg==,;https-host2,/usr/iplanet/ser
vers/https-host2/config,https-host2 (Netscape ES
6.0),https-host2,iplanet,6.0,/usr/iplanet/servers/https-iplanetdefaultsettings,+E
MPTYSTR+,+EMPTYSTR+,1,0,0,0,1,No advanced
authentication,host2,0,undefined,ENC:6f1I5TLVEpuSBHpf4GrASg==
```

Web サーバを再起動するように設定する (Windows のみ)

トラステッド ホスト登録用のポリシー サーバをリストするセクションで、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
USER_REQUESTED_RESTART	設定プロセスの後で必要とされる場合は、インストール プログラムが Windows マシンを再起動できるようにします。 再起動を許可する場合は[はい]に設定します。それ以外の場合は、[いいえ]に設定します。

トラステッド ホスト名およびホスト設定オブジェクトの名前付け

トラステッド ホストおよびホスト設定オブジェクトの名前を付けるセクションでは、以下のテーブルの設定を変更できます。

パラメータ	説明とサンプル値
TRUSTED_HOST_NAME	トラステッド ホストの名前を付けます。一意の名前を指定する必要があります。以下に例を示します。 TRUSTED_HOST_NAME=mytrustedhost
CONFIG_OBJ	ホスト設定オブジェクトを識別します。これは、トラステッド ホストおよびポリシー サーバの間の通信を定義します。以下に例を示します。 CONFIG_OBJ=MyHostSettings

付録 B: Sun Java System サーバ構成に追加された設定

このセクションには、以下のトピックが含まれています。

[Sun Java System Server 6.0 への追加](#) (P. 211)

[Windows プラットフォームへの magnus.conf ファイルの追加](#) (P. 212)

[UNIX プラットフォームの magnus.conf ファイルに追加されたコード](#) (P. 212)

[Windows プラットフォームへの obj.conf ファイルの追加](#) (P. 213)

[UNIX プラットフォームへの obj.conf ファイルの追加](#) (P. 215)

[Windows および UNIX プラットフォームへの mime.types ファイルの追加](#) (P. 217)

[LLAWP でのエージェントスタートアップの確認](#) (P. 217)

Sun Java System Server 6.0 への追加

Oracle iPlanet Web サーバ 6.0 に Web エージェントをインストールする場合、構成設定は以下のファイルに自動的に追加されます。

- magnus.conf
- obj.conf ファイル
- mime.types

Web サーバが開始すると、これらのファイルは自動的にロードされます。追加の設定は Web エージェントを初期化します。Web エージェントインストールプログラムが Web サーバの設定に情報を追加すると、この情報は Oracle iPlanet Web サーバのさまざまなバージョンに異なる方法で分割されます。

Windows プラットフォームでは、これらのファイルは、`Sun_Java_System_install_location\servers\https-hostname\config` ディレクトリにあります。

UNIX プラットフォームでは、これらのファイルは、`/usr/Sun_Java_System_install_location/servers/https-hostname/config` ディレクトリにあります。

注: `Sun_Java_System_install_location` は、コンピュータに Sun Java System サーバをインストールしたディレクトリで、`hostname` はサーバの名前です。

Windows プラットフォームへの magnus.conf ファイルの追加

以下の行が Windows プラットフォームの magnus.conf ファイルに追加されます。

```
Init fn="load-modules" shlib="C:/Program Files/ca/webagent/bin/SunOnewebAgent.dll"  
funcs="SmInitAgent,SmInitChild,SiteMinderAgent,SmRequireAuth,SmAdvancedAuth"  
Init fn=SmInitAgent config="C:/iPlanet/Servers/https-server1/config/WebAgent.conf"  
errortext="Error initializing web Agent..."  
Init fn="SmInitChild" LateInit="yes"
```

注: お使いのファイルのいくつかのエントリは表示されている例と少し異なる場合があります。

追加の行は、以下の NSAPI 機能を SiteMinder Web エージェントにロードするように Web サーバに指示します。

- SmInitAgent
- SiteMinderAgent
- SmRequireAuth
- SmAdvancedAuth

UNIX プラットフォームの magnus.conf ファイルに追加されたコード

以下の行が UNIX プラットフォームの magnus.conf ファイルに追加されます。

```
Init fn="load-modules" shlib="/usr/ca/siteminder/agents/bin/libSunOnewebAgent.so"  
funcs="SmInitAgent,SmInitChild,SiteMinderAgent,SmRequireAuth,SmAdvancedAuth"  
Init fn=SmInitAgent config="/usr/iPlanet/servers/https-yourserver/config/WebAgent.conf"  
errortext+"Error initializing web Agent..."  
Init fn="SmInitChild" LateInit="yes"
```

これらの行は、以下の NSAPI 機能を SiteMinder Web エージェントにロードするように Web サーバに指示します。

- SmInitAgent
- SmInitChild
- SiteMinderAgent
- SmRequireAuth
- SmAdvancedAuth

Windows プラットフォームへの obj.conf ファイルの追加

Web エージェントが高度な認証方式をサポートするように設定される場合、Web エージェントは Sun Java System の obj.conf ファイルに設定を追加します。エージェントが別の高度な認証方式をサポートするように再設定された場合でも、SiteMinder はこれらの設定を削除しません。関係のなくなった設定を削除するために obj.conf ファイルを手動で編集する必要があります。

ファイルのほとんどの追加の行は Web エージェント インストール プログラムによって追加されます。他の行(太字で表示)は、SiteMinder パスワード サービスの JSP バージョンに設定するサーブレット エンジンによって追加されます。

サーブレット エンジンによって追加された行は、SiteMinder Web エージェントによって追加された NameTrans fn 関数の前に配置される必要があります。

変更された obj.conf ファイルの以下の例では、smhome は、システム上で SiteMinder のインストールされた場所を示します。

注: お使いのファイルのいくつかのエントリは表示されている例と少し異なる場合があります。

```
AuthTrans fn="SiteMinderAgent"  
NameTrans fn="assign-name" from="*.jsp*" name="myservletengine"  
NameTrans fn="assign-name" from="/servlet/*" name="myservletengine"  
NameTrans fn="assign-name" from="/siteminderagent/pwservlet/*"  
name="servletengine"  
NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi"  
dir="/smhome/siteminder/webagent/pw" name="cgi"  
NameTrans fn="pfx2dir" from="/siteminderagent/pw"  
dir="/smhome/siteminder/webagent/pw"  
NameTrans fn="pfx2dir" from="/siteminderagent/certooptional"  
dir="/smhome/siteminder/webagent/samples"  
NameTrans fn="pfx2dir" from="/siteminderagent/jpw"  
dir="/smhome/siteminder/webagent/jpw"  
NameTrans fn="pfx2dir" from="/siteminderagent"  
dir="/smhome/siteminder/webagent/samples"  
  
PathCheck fn="SmRequireAuth"  
PathCheck fn="get-client-cert" dorequest="1"  
PathCheck fn="get-client-cert" require="0" dorequest="1"  
  
Service method="(GET|POST)" fn="SmAdvancedAuth"
```

以下のアイテムは、obj.conf ファイルに追加される行のコンテンツを説明します。

- `AuthTrans fn="SiteMinderAgent"` を読み取る行はデフォルトオブジェクト (`<Object name="default">`) に追加されます。これは、Web サーバの認証方式、または `AuthTrans` 関数として `SiteMinder Web` エージェントをセットアップします。
- `NameTrans fn="assign-name" from="/siteminderagent/pwservlet/*" name="myservletengine"` を読み取る行は、エンジンがそれを処理できるように、サーブレットエンジンのインスタンスに JSP パスワード サービス サーブレットをマップする Web エージェントによって追加されたフィルタです。
- `NameTrans fn="pfx2dir"` を開始するほとんどの行は、`SiteMinder` のパスワード サービス (CGI と JSP のバージョン) をサポートするようにエージェントに仮想ディレクトリとマッピングを追加します。
- 証明書ベースの認証方式を設定するように選択した場合は、`NameTrans fn="pfx2dir" from="/siteminderagent/certooptional"` を開始する行が追加されます。
- `PathCheck fn="SmRequireAuth"` を読み取る行は、デフォルトオブジェクトの任意の既存の `PathCheck` 行に追加されます。これは、ユーザがリクエストされたリソースでリクエストされたアクションを実行する許可があることを確認します。
- 設定中に、Web エージェントが高度な認証方式をサポートすることを指示した場合は、`PathCheck fn="get-client-cert" dorequest="1"` を読み取る行が追加されます。これは、証明書認証方式、証明書プラス基本認証方式、証明書およびフォーム認証方式の使用をサポートします。
- 設定中に、Web エージェントが高度な認証方式をサポートすることをインストール中に指示した場合は、`PathCheck fn="get-client-cert" require="0" dorequest="1"` を読み取る行が追加されます。これは、証明書または基本認証方式の使用、または証明書またはフォーム認証方式の使用をサポートします。

注: 高度な認証用の両方の `PathCheck` 行は「SSL を介した基本」ではコメントアウトされる必要があります。
- サービス方式を開始する行は、MIME タイプで実行する内容を Web サーバに指示するために追加されます。
- `NameTrans fn="assign-name" from="*.jsp*" name="myservletengine"` および `NameTrans fn="assign-name" from="/servlet/*" name="myservletengine"` を読み取る行は、`SiteMinder` のパスワード サービスをサポートするためにエージェントにマッピングを作成します。

UNIX プラットフォームへの obj.conf ファイルの追加

Web エージェントが高度な認証方式をサポートするように設定される場合、Web エージェントは Sun Java System の obj.conf ファイルに設定を追加します。エージェントが別の高度な認証方式をサポートするように再設定された場合でも、SiteMinder はこれらの設定を削除しません。関係のなくなった設定を削除するために obj.conf ファイルを手動で編集する必要があります。

ファイルのほとんどの追加の行は Web エージェント インストール プログラムによって追加されます。他の行(太字で表示)は、SiteMinder パスワード サービスの JSP バージョンに設定するサーブレット エンジンによって追加されます。

サーブレット エンジンによって追加された行は、SiteMinder Web エージェントによって追加された NameTrans fn 関数の前に配置される必要があります。

変更された obj.conf ファイルの以下の例では、smhome は、システム上で SiteMinder のインストールされた場所を示します。

注: お使いのファイルのいくつかのエントリは表示されている例と少し異なる場合があります。

```
AuthTrans fn="SiteMinderAgent"
```

```
NameTrans fn="assign-name" from="*.jsp*" name="myservletengine"
NameTrans fn="assign-name" from="/servlet/*" name="myservletengine"
NameTrans fn="assign-name" from="/siteminderagent/pwservlet/*"
name="servletengine"
NameTrans fn="pfx2dir" from="/siteminderagent/pwcgi"
dir="/smhome/siteminder/webagent/pw" name="cgi"
NameTrans fn="pfx2dir" from="/siteminderagent/pw"
dir="/smhome/siteminder/webagent/pw"
NameTrans fn="pfx2dir" from="/siteminderagent/certoptional"
dir="/smhome/siteminder/webagent/samples"
NameTrans fn="pfx2dir" from="/siteminderagent/jpw"
dir="/smhome/siteminder/webagent/jpw"
NameTrans fn="pfx2dir" from="/siteminderagent"
dir="/smhome/siteminder/webagent/samples"
```

```
PathCheck fn="SmRequireAuth"
```

```
#SMSSL 以下の行は「証明書」方式および「証明書および基本」方式ではコメント解除される必要があります。
```

```
PathCheck fn="get-client-cert" dorequest="1"
```

```
#SMSSL 以下の行は「証明書または基本」方式または「証明書またはフォーム」方式ではコメント解除される必要があります。
```

```
PathCheck fn="get-client-cert" require="0" dorequest="1"
```

```
#SMSSL 上記の Pathcheck 行はどちらも「SSL を介した基本認証方式」ではコメントアウトされる必要があります。
```

```
Service method="(GET|POST)" fn="SmAdvancedAuth"
```

以下のアイテムは、obj.conf ファイルに追加される行のコンテンツを説明します。

- `AuthTrans fn="SiteMinderAgent"` を読み取る行はデフォルトオブジェクト (`<Object name="default">`) に追加されます。これは、Web サーバの認証方式、または AuthTrans 関数として SiteMinder Web エージェントをセットアップします。
- `NameTrans fn="assign-name" from="/siteminderagent/pwservlet/*" name="myservletengine"` を読み取る行は、エンジンがそれを処理できるように、サーブレットエンジンのインスタンスに JSP パスワード サービス サーブレットをマップする Web エージェントによって追加されたフィルタです。
- `NameTrans fn="pfx2dir"` を開始するほとんどの行は、SiteMinder のパスワード サービス (CGI と JSP のバージョン) をサポートするようにエージェントに仮想ディレクトリとマッピングを追加します。
- 証明書ベースの認証方式を設定するように選択した場合は、`NameTrans fn="pfx2dir" from="/siteminderagent/certooptional"` を開始する行が追加されます。
- `PathCheck fn="SmRequireAuth"` を読み取る行は、デフォルトオブジェクトの任意の既存の PathCheck 行に追加されます。これは、ユーザがリクエストされたリソースでリクエストされたアクションを実行する許可があることを確認します。
- 設定中に、Web エージェントが高度な認証方式をサポートすることを指示した場合は、`PathCheck fn="get-client-cert" dorequest="1"` を読み取る行が追加されます。これは、証明書認証方式、証明書プラス基本認証方式、証明書およびフォーム認証方式の使用をサポートします。
- 設定中に、Web エージェントが高度な認証方式をサポートすることをインストール中に指示した場合は、`PathCheck fn="get-client-cert" require="0" dorequest="1"` を読み取る行が追加されます。これは、証明書または基本認証方式の使用、または証明書またはフォーム認証方式の使用をサポートします。

注: 高度な認証用の両方の PathCheck 行は「SSL を介した基本」ではコメントアウトされる必要があります。

- サービス方式を開始する行は、MIME タイプで実行する内容を Web サーバに指示するために追加されます。

Windows および UNIX プラットフォームへの mime.types ファイルの追加

以下の行がセットアッププログラムによって mime.types ファイルに追加されます。

```
type=magnus-internal/sfcc exts=sfcc
type=magnus-internal/fcc exts=fcc
type=magnus-internal/scc exts=scc
type=magnus-internal/ccc exts=ccc
```

これらの行は、Mime タイプをセットアップして高度な SiteMinder 機能をサポートします。

LLAWP でのエージェント スタートアップの確認

LLAWP プロセスを開始することにより、Web エージェントが正常に開始されているかを確認することができます。

LLAWP プロセスを開始する方法

1. 設定ウィザードで Web エージェントが設定されていることを確認します。
2. コンソール ウィンドウを開いて以下のコマンドを入力します。

```
LLAWP path_to_WebAgent.conf -web_server_type
```

web_server_type には ISAPI60 または APACHE20 を指定できます。

path_to_WebAgent.conf には、LLAWP を実行している場所からのフルパスまたは相対パスを指定できます。以下に例を示します。

- Windows の場合

```
LLAWP "C:¥Program Files¥ca¥Siteminder Web
Agent¥Bin¥IIS¥WebAgent.conf" -ISAPI60
```

- UNIX の場合

```
LLAWP /usr/apache/conf/WebAgent.conf -APACHE20
```

注: コマンドラインから LLAWP を開始した場合は、コマンドラインからシャットダウンする必要があります。

付録 C: Apache Web エージェントを使用する Web サーバへの設定変更

この付録には、Apache Web エージェントを設定する Web エージェント設定ウィザードを実行することにより、自動的に行われる変更がリストされます。これらの変更は、Apache 2.0、IBM HTTP Server および HP Apache Web サーバなど、Apache Web エージェントをサポートするすべての Web サーバに適用されます。

このセクションには、以下のトピックが含まれています。

[UNIX システムに設定されている Web サーバのライブラリパス \(P. 219\)](#)
[httpd.conf ファイルへの変更 \(P. 220\)](#)

UNIX システムに設定されている Web サーバのライブラリパス

Apache Web サーバのライブラリパスは、Apache サーバがそれによって UNIX システムに正しくライブラリをロードすることができるので、必要です。例:

```
export LD_LIBRARY_PATH web_agent_home/bin
```

ライブラリパス変数はオペレーティングシステムによって異なり、常に `web_agent_home/bin` 指している必要があります。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: C:\Program Files\CA\webagent
- UNIX インストールのデフォルトの場所: /opt/ca/webagent

以下の表に変数をリストします。

オペレーティングシステム	Path 変数
Solaris	LD_LIBRARY_PATH
HP-UX	SHLIB_PATH:
Linux	LD_LIBRARY_PATH

オペレーティング システム	Path 変数
AIX	LIBPATH

httpd.conf ファイルへの変更

設定ウィザードは `httpd.conf` 設定ファイルを変更し、Web サーバが Apache Web エージェントで動作できるようにします。

この手順の例は UNIX プラットフォーム向けですが、同じ変更は該当する Windows 構文を使用して、Windows プラットフォームに行なわれます。

注: 以下の例のとおり、`web_agent_home` 変数は、Web Agent のインストール場所を示します。

- Windows インストールのデフォルトの場所: `C:\Program Files\CA\webagent`
- UNIX インストールのデフォルトの場所: `/opt/ca/webagent`

ほとんどの Apache ベースの Web サーバでは、このファイルは `conf` ディレクトリに配置されています。

`Apache_home/conf`

注: このファイルの場所の詳細については、Web サーバのベンダーによって提供されるドキュメントを参照してください。

DSO サポート セクションに追加されたエントリ

以下の行は、動的共有オブジェクト (DSO) サポート設定セクションに追加されます。これは、ファイルのメイン サーバ設定セクションの前に配置されます。

LoadModule の追加

SiteMinder エージェントは、次をロードするために以下のモジュールのうちの 1 つを必要とします。

Apache 2.0

```
LoadModule sm_module web_agent_home/bin/libmod_sm20.so
```

Windows 上で実行する Apache 2.0

```
LoadModule sm_module web_agent_home/bin/mod_sm20.dll
```

Windows 上で実行する Apache 2.2

```
LoadModule sm_module web_agent_home/bin/mod_sm22.dll
```

ClearModuleList に追加された mod_sm.c エントリ

ディレクティブ `ClearModuleList` が DSO 設定セクションに存在する場合、太字で表示されているように、`mod_sm.c` エントリはファイルの `AddModule` セクションの最後に配置されます。

```
ClearModuleList
AddModule mod_env.c
.
.
.
AddModule mod_servletexec.c
#Siteminder
AddModule mod_sm.c
```

SmInitFile エントリの追加

ファイルのメイン サーバ セクションで、`SmInitFile` エントリが追加されます。

```
SmInitFile Apache_home/conf/WebAgent.con
```

このエントリは `LoadModule` エントリの後に配置されます。相対パスではなく、フルパスが使用されます。例:

```
SmInitFile "/export/Apache2/conf/WebAgent.conf"
```

別名エントリの追加

ファイルの別名セクションで、SiteMinder 機能を有効にするためにエントリが追加されます。

以下の点に注意してください。

- `Alias /siteminderagent/ "web_agent_home/samples/"` エントリは、別名セクションのその他のすべての別名の後に配置する必要があります。
- SiteMinder が Apache Web エージェントで SSL を介した基本または X.509 証明書ベースの認証方式計を使用するには、Apache サーバをコンパイルして `mod_ssl` モジュールを含めることによって、SSL を有効にする必要があります。このモジュールを取得するには、www.modssl.org を参照してください。
- 別名エントリはそれぞれそれぞれ独自の行に表示されます。

パスワード サービス

```
Alias /siteminderagent/pwcgi/ "<web_agent_home/pw/>"
<Directory "/export/webagent/pw/">
    Options Indexes MultiViews ExecCGI
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

```
Alias /siteminderagent/pw/ "<web_agent_home>/pw/"
<Directory "/export/webagent/pw/">
    Options Indexes MultiViews ExecCGI
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

SSL を介した基本認証方式

```
AliasMatch /siteminderagent/nocert/[0-9]+/(.*)
"<web_agent_home>/$1"
<Directory "<web_agent_home>/$1">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

X509 クライアント証明書または X509 クライアント証明書および基本認証

```
AliasMatch /siteminderagent/cert/[0-9]+/(.*)
```

```
"<web_agent_home>/$1"  
<Directory "<web_agent_home>/$1">  
    Options Indexes  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

X509 クライアント証明書または基本認証

```
AliasMatch /siteminderagent/certoptional/[0-9]+/(.*) "<web_agent_home>/$1"  
<Directory "<web_agent_home>/$1"  
    Options Indexes  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

X509 証明書またはフォーム、または X509 クライアント証明書およびフォーム認証

```
Alias /siteminderagent/certoptional/"<web_agent_home>/  
samples/"  
<Directory "<web_agent_home>/samples/"  
    Options Indexes  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

フォーム認証またはエージェントはシングル サインオンのための Cookie プロバイダです。

```
Alias /siteminderagent/ "<web_agent_home>/samples/"  
<Directory "/export/webagent/samples/">  
    Options Indexes MultiViews  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

注: これはセクションの最後に配置する必要がある別名です。

証明書認証エントリの追加

- X509 クライアント証明書、X509 クライアント証明書および基本、または X509 クライアント証明書または基本認証を使用している場合は、仮想ホストセクションの以下の SSL エンジン オプション エントリが該当する仮想ホスト(複数のホストが定義されている場合)に対してコメント解除されます。

SSLOptions+ExportCertData+StdEnvVars

注: 仮想ホストセクションに既存の SSL オプションがある場合、その既存のエントリはコメントアウトされ、新しい SSL エントリが追加されます。

- ユーザが X509 クライアント証明書またはフォーム認証を使用している場合は、仮想ホストセクションの以下の SSL エンジン Options エントリが該当する仮想ホスト(複数のホストが定義されている場合)に対してコメント解除されます。

SSLOptions+StdEnvVars+CompatEnvVars

- ファイルの仮想ホストセクションで、SSL クライアント認証タイプはオプションに設定されます。

オプションの SSLVerifyClient

付録 D: Web エージェント インストールによって追加または変更された環境変数

このセクションには、以下のトピックが含まれています。

[環境変数の追加または変更 \(P. 225\)](#)

環境変数の追加または変更

以下の環境変数が Web エージェント インストールによって追加または変更されます。

- `NETE_WA_ROOT = $INSTALL_PATH$`
- `NETE_WA_PATH=$INSTALL_PATH$/$bin`

索引

1

1 つのシステムに複数のトラステッド ホストを登録する (UNIX) - 74

1 つのシステムに複数のトラステッド ホストを登録する (Windows) - 49

A

Adobe Acrobat Reader が Windows システムにインストールされない - 195

AIX 要件 - 19

Apache Web エージェント

httpd.conf の変更 - 220

IBM HTTP Web サーバ用 - 140

Stronghold サーバ用 - 140

アンインストール、UNIX - 181

インストール - 53

共有メモリの増加 - 167

共有メモリの調整 - 167

再インストール - 63

サポートされるプラットフォーム - 15

設定 - 137, 141

設定、GUI モード - 141

設定、コンソール モード - 141

設定ウィザード、アクセス - 88, 127, 137

Apache Web エージェントが動作しない - 198

Apache Web エージェントの問題 - 197

Apache Web エージェントのログ サブディレクトリの追加 - 21

Apache Web エージェントを使用する Web サーバへの設定変更 - 219

Apache Web エージェントを設定する - 137

Apache Web サーバ

Windows へのインストール、注意 - 17

サービスとしてのインストール - 17

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 - 62

Apache エージェント操作に LD_PRELOAD 変数を設定する - 144

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 - 198

Apache サーバは起動するが Web エージェントが有効にならない - 197

B

bootstrap サーバ、設定 - 44, 68

C

ca-wa-installer.properties ファイル - 201

CA 製品リファレンス - iii

CA への連絡先 - iii

ClearModuleList に追加された mod_sm.c エントリ - 221

D

DLL

追加、Domino Web エージェント - 148

Domino 6.5.2 に必要とされる IBM Hot Fix - 21

Domino 6.5.3/SuSe8 Linux システムで X.509 ベースの認証方式を使用するための LD_PRELOAD を設定する - 145

Domino Web エージェント

DLL の追加 - 148

アンインストール、UNIX - 181

インストール、UNIX - 53

再インストール、UNIX - 63

再設定、Windows - 160

設定、UNIX - 153

設定/Windows - 149

設定ウィザード、アクセス - 149

Domino Web エージェント DLL の追加 (UNIX) - 152

Domino Web エージェント DLL の追加 (Windows) - 148

Domino Web エージェントが有効になっていないのに Web サーバが開始する - 199

Domino Web エージェントで必要とされるエージェント設定パラメータ - 26

Domino Web エージェントの問題 - 198

Domino Web エージェントを設定する - 147

DSO サポート セクションに追加されたエントリ - 220

E

Exchange Web サイトへの ISAPI 機能拡張の追加 - 122

Exchange Web サイトへのディレクトリ セキュリティの設定 - 123

Exchweb Web サイトへの ISAPI 機能拡張の追加 - 124

Exchweb Web サイトへのディレクトリ セキュリティの設定 - 125

G

GUI またはコンソール モードを使用する Oracle iPlanet Web エージェントの設定 - 130

GUI またはコンソール モードでの Domino Web エージェントの設定 - 153

GUI またはコンソール モードでのトラステッド ホストの登録 - 64

GUI またはコンソール モードを使用する Apache Web エージェントの設定 - 141

H

HP-UX

アンインストール、Sun Java System Web エージェント - 181

HTML フォーム認証方式を有効にするための別名設定の設定(オプション) - 151

httpd.conf

Apache 用の変更 - 220

httpd.conf ファイルへの変更 - 220

I

IBM HTTP Server

エージェントのインストール - 23, 53

エージェントの設定 - 140

IBM HTTP サーバログの書き込み許可の有効化 - 21

IIS 6.0 に SiteMinder Web エージェントを設定する方法 - 107

IIS 7.0 に SiteMinder Web エージェントを設定する方法 - 97

IIS 7.5 Web Server への役割サービスの追加 - 84

IIS 7.5 に SiteMinder Web エージェントを設定する方法 - 83

IIS 7.x Web サーバへの役割サービスの追加 - 98

IIS Web エージェントの設定ウィザードの実行 - 88, 102, 112

IIS Web エージェント

IIS 6.0、前提条件 - 107

再インストール - 37

再設定 - 160

設定 - 88

IIS Web エージェント設定時のメタベース エラー - 193

IIS Web エージェントで必要とされるエージェント設定パラメータ - 27

IIS Web エージェントの設定 - 83

IIS Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法 - 173

IIS がエージェント ISAPI および CGI 拡張を実行できるようにする - 109

IIS で Outlook 拡張を実行できるようにする - 120

installer.properties、説明 - 35, 158

installer.properties ファイル

説明 - 59

J

JSP パスワードサービス

必要な変更、Windows - 171

JSP パスワード サービスの ServletExec の CLASSPATH を修復する(Windows) - 28

JSP パスワード サービス用の環境をセットアップ
する方法 - 172

L

LD_LIBRARY_PATH 変数を IBM HTTP Server 7.0
用に設定 - 21

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 - 77

Linux

Apache サーバのコンパイル - 20

Linux システムでの Apache Web サーバのコン
パイル - 20

LLAWP がすでに実行しているかまたはログ メッ
セージが適切なログ ファイルに書き込まれて
いないためにエージェントが起動しない - 186

LLAWP でのエージェント スタートアップ/シャット
ダウンのトラブルシュート - 187

LLAWP でのエージェント スタートアップの確認 -
217

LLAWP のシャットダウン - 187

LoadModule の追加 - 221

M

Microsoft Visual C++ 2005 再頒布可能パッケー
ジ(x64)の前提条件 - 15

N

Netscape。iPlanet Web サーバを参照 - 88, 137

Netscape ブラウザで PDF が開きません - 194

NT。Windows を参照 - 149

O

obj.conf

エージェントによる変更 - 211

Oracle iPlanet Web エージェントの設定 - 127

Oracle iPlanet Web エージェントの問題 - 195

Oracle iPlanet Web サーバファイルへの変更の
適用 - 135

Oracle iPlanet Web サーバの手動設定 - 133

S

ServletExec

クラスパスの修復、DMS - 28

SiteMinder ISAPI フィルタがリストの最初に表示
されていることを確認します。 - 119

SiteMinder Web エージェントを設定して
Microsoft Outlook Web Access を保護する方
法 - 117

SiteMinder Web エージェント用のクラシック
モード アプリケーション プールの設定 - 86,
100

SiteMinder が Outlook Web Access Web サイト
を保護していることを確認します。 - 126

SiteMinder で保護する IIS 7.5 Web サイトへの
エージェント ISAPI フィルタの追加 - 89

SiteMinder で保護する IIS 7.5 Web サイトへの
ハンドラ マッピングの追加 - 91

SiteMinder で保護するアプリケーションのクラ
シック モード アプリケーション プールへの移
動 - 87, 101

SiteMinder で保護する追加の Web サイトに
エージェント ISAPI フィルタを追加 - 107

SiteMinder で保護する追加の Web サイトへの
ハンドラ マッピングの追加 - 105

SiteMinder 管理者

ホストの登録のため - 24

SmHost.conf

作成、UNIX - 63

説明 - 44, 68

変更、Windows - 44, 68

用途 - 63

SmHost.conf ファイルの変更 (UNIX) - 68

SmHost.conf ファイルの変更 (Windows) - 44

SmlnitFile エントリの追加 - 221

Solaris 10 リソース管理を調整する方法 - 170

SSL 認証方式、設定 - 127, 130

SSL を介した HTTP 基本認証方式 - 127, 130

Stronghold Web サーバ

Apache エージェントの使用 - 140

エージェントのインストール - 23, 53

Sun Java System Server 6.0 への追加 - 211

Sun Java System Web エージェント

アンインストール、UNIX - 181

共有メモリの増加 - 167

共有メモリの調整 - 167

再インストール、UNIX - 63

再インストール、Windows - 37

再設定、Windows - 160

Sun Java System Web サーバ

obj.conf への変更 - 211

Sun Java System Web サーバがランタイムで失敗する - 197

Sun Java System サーバ構成に追加された設定 - 211

SuSE Linux 9 for zSeries 上の Apache エージェントに LD_ASSUME_KERNEL を設定する - 145

U

UNIX オペレーティング環境で HTML フォーム認証方式を有効にするための別名設定の設定 (オプション) - 155

UNIX オペレーティング環境で Oracle iPlanet Web サーバに JSP パスワード サービス用の ServletExec サブレット エンジンを設定する方法 - 174

UNIX システムで Web エージェントを r12.0 SP3 にアップグレードする - 80

UNIX システムへの Web エージェントのインストール - 53

UNIX 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定 (オプション) - 155

UNIX システムからの Web エージェントのアンインストール - 181

UNIX システムからのドキュメントのアンインストール - 183

UNIX システムでの Apache Web エージェントの設定方法 - 140

UNIX システムでの Domino Web エージェントの設定方法 - 153

UNIX システムに Domino Web エージェントを設定する方法 - 151

UNIX システムに Web エージェントをインストールする - 51

UNIX システムに設定されている Web サーバのライブラリパス - 219

UNIX システムへの Web エージェントのドキュメントのインストール - 51

UNIX 上で Web エージェント インストールに DISPLAY を設定する - 18

UNIX 上で進行中の無人インストールを停止する - 60

UNIX 上でトラステッド ホストとしてシステムを登録する - 63

UNIX での GUI モードインストールの実行 - 54

UNIX でのコンソール モードインストールの実行 - 57

UNIX での無人インストールの実行 - 60

UNIX での無人インストールの準備 - 59

UNIX の無人インストール - 58

UNIX プラットフォーム

Apache Web エージェントの設定 - 141

Web エージェントの再インストール - 63

インストール、Domino Web エージェント - 53

インストール、Sun Java System Web エージェント - 53

インストールに必要な条件 - 15

エージェント、Stronghold サーバ - 140

エージェントのインストール - 53

エージェントのインストールに必要なデータ - 22

UNIX プラットフォームの magnus.conf ファイルに追加されたコード - 212

UNIX プラットフォームへの obj.conf ファイルの追加 - 215

UNIX への Web エージェントの再インストール - 63

W

Web エージェント インストールのための

Windows システムを準備する方法 - 15

Web エージェントを r12.0 SP3 にアップグレードする - 75

WEB_SERVER_INFO 変数 - 206

-
- Web エージェント
 - Apache、設定、GUI モード - 141
 - Apache、設定、コンソール モード - 141
 - Apache の設定 - 137, 141
 - Domino/Windows、設定 - 149
 - httpd.conf の変更、Apache - 220
 - IBM HTTP Server、設定 - 140
 - IIS、設定 - 88
 - アンインストール、UNIX - 181
 - インストール、UNIX プラットフォーム - 53
 - 再インストール、UNIX - 63
 - 再インストール、Windows - 37
 - 再設定、Windows - 160
 - サポートされる UNIX プラットフォーム - 15
 - ドキュメントのアンインストール、Windows - 180
 - Web エージェント インストールによって追加または変更された環境変数 - 225
 - Web エージェント インストールのための Domino システムを準備する方法 - 20
 - Web エージェント インストールのための Linux システムを準備する方法 - 19
 - Web エージェント インストールのための UNIX システムを準備する方法 - 18
 - Web エージェント設定ウィザード
 - アクセス、Apache Web サーバ - 88, 127, 137
 - アクセス、Domino Web サーバ - 149
 - Web エージェントのアップグレードの準備をする方法 - 75
 - Web エージェントのアンインストール - 177
 - Web エージェントのアンインストールに関する注意事項 - 177
 - Web エージェントのインストール
 - Apache - 53
 - Domino/UNIX - 53
 - Sun Java System/UNIX - 53
 - UNIX 上 - 53
 - Web エージェントのインストールの準備をする方法 - 14
 - Web エージェントの開始とシャットダウンの問題 (IBM HTTP Server) - 188
 - Web エージェントの起動と停止 - 165
 - Web エージェントの再設定 - 160
 - Web エージェントのドキュメントのアンインストール
 - UNIX - 183
 - Windows - 180
 - Web エージェントの無効化 - 166
 - Web エージェントの有効化 - 165
 - Web エージェントをアンインストールする前に JRE を PATH 変数に設定する - 178
 - Web サーバが開始すると smget エラーメッセージが表示される - 196
 - Web サーバ設定
 - Windows の再起動、無人インストール - 209
 - 設定、無人インストール - 204
 - Web サーバは起動するのに Web エージェントが有効にならない - 195
 - Web サーバへの適切なエージェントのインストール - 23
 - Web サーバを再起動するように設定する (Windows のみ) - 209
 - Windows
 - Domino Web エージェント、設定 - 149
 - IIS Web エージェントの設定 - 88
 - Web エージェントの再インストール - 37
 - ドキュメントのアンインストール - 180
 - Windows および UNIX プラットフォームへの mime.types ファイルの追加 - 217
 - Windows システムからの Web エージェントのアンインストール - 179
 - Windows システムでの Web エージェントの r12.0 SP3 へのアップグレード - 78
 - Windows システムに Web エージェントをインストールする - 31
 - Windows での無人インストールの実行 - 36
 - Windows プラットフォームへの magnus.conf ファイルの追加 - 212
 - Windows プラットフォームへの obj.conf ファイルの追加 - 213
 - Windows 上で進行中の無人インストールを停止する方法 - 37
 - Windows 動作環境での CGI ディレクトリおよび CGI URL パス設定の設定 (オプション) - 151
-

Windows 認証方式の仮想ディレクトリの設定 (IIS 6.0) - 116

Windows 認証方式用の仮想ディレクトリの作成と設定 (IIS 7.5) - 96

Windows 認証方式用の仮想ディレクトリの設定 (IIS 7.0) - 104

Windows 認証役割サービスの IIS 7.x Web サーバへの追加 - 85, 99

Windows システムからのドキュメントのアンインストール - 180

Windows システムでの Apache Web エージェントの設定 - 137

Windows システムでの Domino Web エージェントの設定 - 147

Windows 上でトラステッド ホストとしてシステムを登録する - 39

Windows での GUI モードインストールの実行 - 31

Windows での設定ウィザードの実行 - 127

Windows での無人インストール - 34

Windows での無人インストールの準備 - 35

Windows に Web エージェントを再インストールします。 - 37

Windows の Domino Web エージェントでの設定ウィザードの実行 - 149

Windows プラットフォーム
Apache Web エージェントの設定 - 137

X

X.509 クライアント証明書および HTML フォーム認証方式 - 127, 130

X.509 クライアント証明書および基本認証方式 - 127, 130

X.509 クライアント証明書または HTML フォーム認証方式 - 127, 130

X.509 クライアント証明書または基本認証方式 - 127, 130

X509 クライアント証明書認証方式 - 127, 130

あ

アップグレード

LD_PRELOAD の設定 - 77

アップグレード前の問題 - 76

一般的な手順 - 76

カスタム ファイルのバックアップ - 76

設定ウィザードの実行、結果 - 76

パスワード サービス テンプレート - 76

フォーム テンプレート - 76

読み取り専用ファイルの置換 - 77

アップグレード中にエラー メッセージが表示される - 192

アップグレード手順の確認 - 76

アップグレードの後に設定ウィザードを実行した場合の結果 - 76

アップグレードの間のパスワード サービスとフォーム テンプレートの変更 - 76

アプリケーションプール識別情報に対する SiteMinder SmHost.conf ファイルおよびログ ディレクトリの許可 - 94

アンインストール
ドキュメント

UNIX - 183

一般情報
設定、無人インストール - 202

一般情報の変更 - 202

一般的なインストールの問題 - 191

インストール
ドキュメント、UNIX - 51

インストール エラー ログの場所 - 191

インストールおよび設定ログ ファイル - 43, 56

インストール後の Web エージェント環境変数の設定 - 61

インストールの前提条件
Web エージェント、UNIX - 15

インストール履歴ログ ファイル - 38, 62

エージェント インストールを完了するのに必要な情報の収集 - 22

エージェント設定オブジェクト
Domino の要件 - 24

IIS 要件 - 24

インストール要件 - 24

定義 - 24

エージェントのアップロード ファイルのサイズ制限の増加 - 111
エージェントのスタートアップおよびシャットダウンの問題(フレームワーク エージェントのみ) - 185
同じサーバ上の IIS 6.0 Web エージェントとサードパーティソフトウェア - 110
同じシステムでの複数のインストール中に 1 つのインストールがハングする - 191
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 - 144
オペレーティング システムのチューニング - 167

か

カスタマイズされたファイルのバックアップ - 76
環境変数の追加または変更 - 225
既存の読み取り専用ファイルの置換 - 77
起動時に Apache サーバに shmget エラーが表示される - 198
共有秘密キーのロールオーバーに対する SmHost.conf ファイル アクセス許可の確認 - 160
共有メモリ セグメント、調整 - 167
共有メモリ セグメントの調整 - 167
コントロール パネルの[プログラムの追加と削除]に Web エージェントが表示されない - 192

さ

再インストール
Web エージェント、UNIX - 63
Web エージェント、Windows - 37
再設定
Web エージェント、Windows - 160
再設定された Web エージェントが動作しない - 196
サポートされているオペレーティング システムと Web サーバ - 15
サポートされるプラットフォーム
UNIX - 15
サンプルおよびエラー ファイル ディレクトリへの読み取り許可の割り当て - 108

準備 - 13
証明書認証エントリの追加 - 224
すべての Web エージェントに対する一般的な準備 - 22
すべての Web エージェントに利用可能な設定 - 157
すべてのエージェントで必要とされるエージェント設定パラメータ - 25
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 - 17
接続およびトラステッド ホスト登録の問題 - 188
設定
無人モード、Windows - 157
設定用の Web サーバの選択 - 204
その他の Web サーバの準備 - 21
その他の問題 - 193

た

ダイナミック ポリシー サーバクラスタ - 162
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 - 163
他のサードパーティフィルタの前にエージェント フィルタおよび拡張を配置する - 114
追加のエージェント コンポーネントをセットアップする方法 - 161
デフォルト以外の IIS Web サイトを使用する方法 - 16
デフォルトの Web サイトのディレクトリ場所と実行権限の設定 - 121, 126
登録ツールを使用したトラステッド ホストの再登録 (UNIX) - 70
登録ツールを使用したトラステッド ホストの再登録 (Windows) - 46
ドキュメント
UNIX システムでのアンインストール - 183
Windows システムでのアンインストール - 180
アンインストール
UNIX - 183
インストール、UNIX - 51
トラステッド ホスト

設定、無人インストール - 202, 209
定義 - 24, 63
登録、UNIX - 63
複数のホストの登録 - 49, 74
トラステッド ホストからポリシー サーバへの接続
がない - 190
トラステッド ホスト登録用のポリシー サーバの識
別 - 203
トラステッド ホストの登録 - 202
 UNIX プラットフォーム上 - 63
トラステッドホストの登録
 管理者権限 - 24
 複数のホストの登録 - 49
トラステッドホストの登録
 複数のホストの登録 - 74
トラステッド ホストの登録に失敗 - 189
トラステッド ホスト名およびホスト設定オブジェク
トの名前付け - 209
トラブルシューティング - 185

な

任意の WebAgentTrace.conf ファイル変更の保
存 - 22
認証方式
 SSL、設定 - 127, 130
 SSL を介した HTTP 基本 - 127, 130
 X.509 クライアント証明書および HTML
 フォーム - 127, 130
 X.509 クライアント証明書および基本 - 127,
 130
 X.509 クライアント証明書または HTML フォー
 ム - 127, 130
 X.509 クライアント証明書または基本 - 127,
 130
 X509 クライアント証明書 - 127, 130
 フォーム認証の使用 - 161

は

パスワード サービス - 171
 JSP のバージョンの設定、Windows - 171
 JSP バージョン - 171

パスワード サービスとフォーム ディレクトリ - 29
パスワード サービスの実装 - 171
パスワード サービスの準備 - 28
必要とされる Linux ツール - 20
必要とされる Linux パッチ - 19
必要とされる Linux ライブラリ - 19
必要とされる Solaris パッチ - 18
フォーム認証方式
 認証情報コレクション - 161
複数の bootstrap サーバ、設定 - 44, 68
別名エントリの追加 - 222
ホスト設定オブジェクト
 インストール要件 - 24
 定義 - 24
ホスト設定ファイル
 設定、無人インストール - 203
 変更、Windows - 44, 68
 用途 - 44, 63, 68
ホスト設定ファイルに書き込み許可がない - 188
ホスト設定ファイルの指定 - 203
ホストは登録されたが SMHost.conf ファイルが
削除されている - 190
ポリシー サーバ
 エージェントとの初期接続 - 63
 信頼されたホストの登録、UNIX - 63
 設定、無人インストール - 203
 設定の確認 - 24
ポリシー サーバ要件 - 24

ま

無人インストール - 201
 installer.properties、説明 - 35, 158
 installer.properties ファイル、説明 - 59
 UNIX - 58
 Windows - 34
 実行、UNIX - 60
 実行、Windows - 36, 158
 準備 - 35, 59, 158
無人設定
 Windows - 157
無人設定の実行 - 158
無人設定の準備 - 158

無人モードで任意の Web エージェントを設定
する方法 - 157

ら

ローカル設定モードの使用時に Domino エー
ジェントが初期化できない - 199