ポリシーエージェントガイド

Sun™ ONE Identity Server

Version 6.0

817-1574-10
2002 年 12 月

目 次

本書について

ポリシーエージェントガイド では、Sun™ ONE Identity Server ポリシーエージェントの概要と、Web サーバ、プロキシサーバ、アプリケーションサーバに Sun ONE Identity Server ポリシーエージェントをインストールし、設定する方法について説明します。

ここでは、次の項目について説明します。

- お読みになる前に
- Identity Server のマニュアルセット
- マニュアルの内容
- 表記上の規則
- 関連情報

お読みになる前に

本書は、Sun ONE Identity Server 6.0 に付属する一連のマニュアルの中で、補助的なマニュアルです。このマニュアルでは、ディレクトリテクノロジについて理解し、Java および XML プログラミング言語の使用経験があることを前提にしています。ディレクトリサーバや LDAP (Lightweight Directory Access Protocol) に精通していれば、このマニュアルを最大限に活用できます。Sun ONE Directory Server のマニュアルを精読して、製品の使用方法に慣れておくことをお勧めします。

このマニュアルは、Sun ONE サーバおよびサービスを介したネットワークアクセスを管理する IT 技術者向けに書かれています。Sun ONE Identity Server に含まれる機能を利用すれば、全社的にユーザデータを管理し、アクセスポリシーを施行できます。

このマニュアルで説明する概念を理解するため、『Sun ONE Identity Server インストールおよび設定ガイド』および『Sun ONE Identity Server Programmer's Guide』を参照してください。

Identity Server のマニュアルセット

Sun ONE Identity Server のマニュアルセットには、次のマニュアルが含まれています。

- 『Product Brief』: Sun ONE Identity Server アプリケーションの概要と機能について説明します。
- 『インストールガイド』: Solaris™、Linux、Windows 2000 の各システムに Sun ONE Identity Server をインストールおよび配備する詳細な方法について説明します。
- 『Administration Guide』: Sun ONE Identity Server コンソールの使用方法と、コマンド行によるユーザ管理およびデータサービスの方法について説明します。
- 『Programmer's Guide』: 組織に合わせて Sun ONE Identity Server システムをカスタマイズする方法について説明します。また、公共の API を使ってアプリケーションに新しいサービスを付加する方法についても説明します。
- 『ポリシーエージェントガイド』(本書): リモートサーバに Sun ONE Identity Server ポリシーエージェントをインストールし、設定する方法について説明します。また、トラブルシューティングや、各エージェントに固有の情報についても説明します。
- 『Getting Started Guide』: Sun ONE Identity Server のさまざまな機能を利用して、ID、ポリシー、ロールが設定された簡単な組織を設定する方法について説明します。
- 『リリースノート』: このリリースの最新情報、インストールに関する最新の注意事項、既知の問題、制限事項、問題の報告方法などの各種情報を提供します。

注 リリースノートの更新およびマニュアルの改訂については、Sun ONE Identity Server マニュアルの Web サイト (`http://docs.sun.com/db/prod/s1idsrv#hic`) を確認してください。更新された文書には改訂日を記してあります。

マニュアルの内容

次の表には、このマニュアルで説明するすべてのエージェントが記載されています。

表 1 このマニュアルで説明するエージェント

エージェント	プラットフォーム
Sun ONE Web Server 6.0 SP*x*	Solaris 8
Sun ONE Web Server 4.1 SP8	Solaris 8
Sun ONE Web Proxy Server 3.6 (リバースプロキシモード)	Solaris 8
Microsoft IIS 5.0	Windows 2000

表 1　このマニュアルで説明するエージェント (続き)

エージェント	プラットフォーム
Sun ONE Web Server 6.0 SP*x*	Windows 2000
Microsoft IIS 4.0	Windows NT 4.0
Sun ONE Web Server 6.0 SP*x*	Solaris 9
Apache 1.3.26	Solaris 8
Apache 1.3.26	Solaris 9
Apache 1.3.26	Red Hat Linux 7.2
WebLogic 6.1 SP2	Solaris 8
WebLogic 6.1 SP2	Windows 2000
WebLogic 6.1 SP2	HP-UX 11

表記上の規則

このマニュアルを含む Sun ONE Identity Server 6.0 のマニュアルでは、説明を簡潔にし、内容をより理解しやすくするために、特定の表記および用語を使用します。これらの規則について次に説明します。

表記上の規則

このマニュアルでは、次の表記規則を適用します。

- イタリック体は、新出用語、強調語句、および文字通りの意味の語句を示すときに使用します。
- モノスペース (等倍) フォントは、サンプルコードとコードのリスト、API および言語の要素 (関数名、クラス名など)、ファイル名、パス名、ディレクトリ名、HTML タグ、画面に入力する必要のあるテキストを示すときに使用します。
- Serif フォントは、コードおよびコードフラグメント内の可変部分を示すときに使用します。たとえば、次のコマンドの場合、*filename* の位置には `gunzip` コマンドの引数が入ります。

  `gunzip -d` *filename*`.tar.gz`

用語

Sun ONE Identity Server ポリシーエージェントのマニュアルで共通に使用する用語を次に示します。

- *Agent_Install_Dir* は、Sun ONE Identity Server ポリシーエージェントをインストールしたディレクトリの可変部分を示します。
- *S1IS_Install_Dir* は、Sun ONE Identity Server 6.0 をインストールしたホームディレクトリの可変部分を示します。
- *WebLogic_Install_Dir* は、WebLogic Server をインストールしたホームディレクトリの可変部分を示します。

関連情報

Sun ONE Identity Server のマニュアルの他にも、参考になるマニュアルがあります。これらのマニュアルの入手先と関連情報を次に示します。

iPlanet Directory Server のマニュアルセット

iPlanet Directory Server 5.1 のマニュアルは、次の Web サイトから入手できます。
`http://docs.sun.com/db/coll/S1_ipDirectoryServer_51`

iPlanet/Sun ONE Web Server のマニュアルセット

iPlanet/Sun ONE Web Server のマニュアルは、次の Web サイトから入手できます。
`http://docs.sun.com/db/coll/S1_ipwebsrvree60_en`

Sun ONE Certificate Server のマニュアルセット

Sun ONE Certificate Server のマニュアルは、次の Web サイトから入手できます。
`http://docs.sun.com/db/coll/S1_s1CertificateServer_47`

iPlanet Proxy Server のマニュアルセット

iPlanet Proxy Server のマニュアルは、次の Web サイトから入手できます。
`http://docs.sun.com/db/coll/S1_ipwebproxysrvr36`

その他の iPlanet 製品のマニュアル

Sun ONE サーバおよびその関連技術に関するその他すべてのマニュアルは、次の Web サイトから入手できます。
`http://docs.sun.com/db/prod/sunone`

ダウンロードセンタ

Sun ONE/iPlanet ソフトウェアのダウンロードについては、次の Web サイトを参照してください。

`http://wwws.sun.com/software/download/`

Sun ONE テクニカルサポート

テクニカルサポートは、次の Web サイトから利用できます。

`http://www.sun.com/service/support/software/iplanet/index.html`

プロフェッショナルサービス

プロフェッショナルサービスは、次の Web サイトから利用できます。

`http://www.sun.com/service/sunps/iplanet/`

Sun エンタープライズサービスによる Solaris のパッチとサポート

Solaris のパッチとサポートは、次の Web サイトから利用できます。

`http://www.sun.com/service/`

開発者向け情報

Sun™ ONE Identity Server、LDAP、Sun ONE Directory Server、およびそれぞれの関連技術については、次の Web サイトを参照してください。

`http://developer.iplanet.com/tech/directory/`

第 1 部

Web エージェントとプロキシエージェント

第 1 章

ご使用にあたって

この章では、Web サーバおよび Web プロキシサーバ向けの Sun ONE Identity Server ポリシーエージェントについて、およびインストールプログラムに進む前に理解しておく必要のあるいくつかの概念について、その概要を説明します。この章にある情報は、Solaris、Windows、Linux の各オペレーティングシステムに共通です。

次のトピックがあります。

- ポリシーエージェントの動作
- サポートされるサーバ
- インストールを始める前に

ポリシーエージェントの動作

Sun ONE Identity Server のポリシーエージェントは、Web サーバと Web プロキシサーバを不正侵入から保護します。このエージェントは、管理者が設定したポリシーに基づいてサービスおよび Web リソースへのアクセスを制御します。

ポリシーエージェントの使用

ポリシーエージェントは、さまざまな目的で Web サーバにインストールされます。次に 3 つの例を示します。

- 人事部門のサーバ上のエージェントは、人事担当者以外の人が機密の給与情報やその他の機密データを見ることを防止します。
- 運用部門の Web サーバ上のエージェントは、ネットワーク管理者だけにネットワーク状態に関するポートの閲覧やネットワーク管理記録の変更を許可します。

- エンジニアリング部門の Web サーバ上のエージェントは、多くの社内組織の中で、認定を受けた人に研究開発情報の公開や共有を許可します。同時に、エージェントは外部の提携業者が社外秘情報にアクセスするのを制限します。

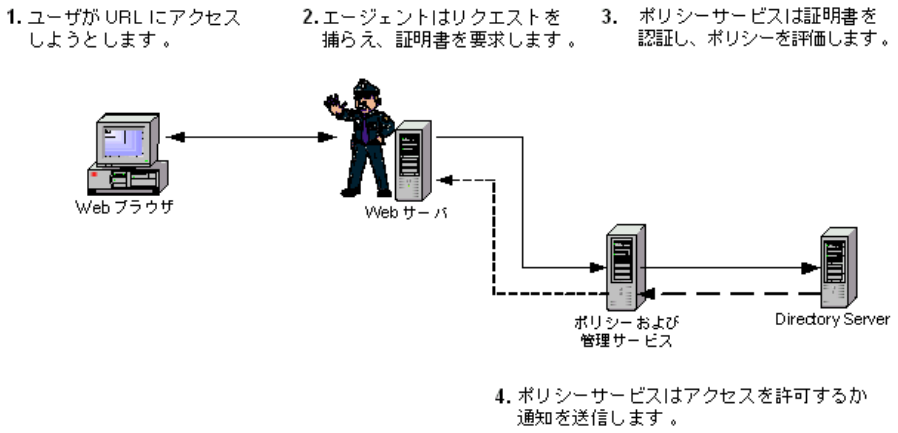
これらの状況ごとに、システム管理者は Web サーバ上のコンテンツへのユーザのアクセスを許可または拒否するポリシーを設定する必要があります。ポリシーの設定およびユーザへのロールとポリシーの割り当てについては、『Sun ONE Identity Server Administration Guide』を参照してください。

エージェントと Sun ONE Identity Server 6.0 の対話

図 1-1 は、リモート Web サーバにインストールされたポリシーエージェントがどのように Sun ONE Identity Server と対話するかを示しています。保護されている Web サーバ上の特定の URL にユーザがブラウザでアクセスを試みると、次の対話が行われます。

1. エージェントが要求を受け取ると、既存の認証資格と比較検証します。既存の認証レベルを満たしていない場合は、適切な Sun ONE Identity Server 認証サービスがログインページを表示します。ログインページでは、ユーザ名やパスワードなどの資格情報をユーザに要求します。
2. 認証サービスが、ユーザの資格が有効であるかを確認します。たとえば、デフォルトの LDAP 認証サービスの場合は、ユーザ名とパスワードが Sun ONE Directory Server に保存されているかを確認します。RADIUS モジュールや証明書モジュールなど、ほかの認証モジュールを使用することもできます。その場合、Directory Server は資格を検証しませんが、適切な認証モジュールが資格の検証を行います。
3. ユーザの資格が正しく認証されると、ポリシーエージェントは、ユーザに割り当てられているすべてのロールを調べます。
4. ユーザに割り当てられたすべてのポリシーの集合に基づいて、個々のユーザは URL へのアクセスを許可または拒否されます。

図 1-1　エージェントと Sun ONE Identity Server の対話



サポートされるサーバ

Sun ONE Identity Server ポリシーエージェントは、Solaris 8 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Sun ONE Web Server 6.0 SP*x*
- Sun ONE Web Server 4.1 SP8
- Sun ONE Web Proxy Server 3.6 (リバースプロキシモード)
- Apache 1.3.26

Sun ONE Identity Server ポリシーエージェントは、Solaris 9 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Sun ONE Web Server 6.0 SP*x*
- Apache 1.3.26

Sun ONE Identity Server ポリシーエージェントは、Red Hat Linux 7.2 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Apache 1.3.26

Sun ONE Identity Server ポリシーエージェントは、Windows 2000 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Microsoft IIS 5.0
- Sun ONE Web Server 6.0 SP*x*

Sun ONE Identity Server ポリシーエージェントは、Windows NT 4.0 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Microsoft IIS 4.0

インストールを始める前に

インストールプログラムを開始する前によく理解しておく必要のある問題と概念を次に示します。

- JRE (Java Runtime Environment) 1.3.1 要件
- Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ
- 同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定
- Sun ONE Identity Server エージェントのフェイルオーバ機能の提供
- エージェントキャッシュの更新
- グローバル不適用 URL リスト
- グローバル不適用 IP アドレスリスト
- ポリシーを適用しない認証だけの適用
- HTTP ヘッダーを介した LDAP ユーザ属性の転送
- `AMAgent.properties` ファイル
- 完全指定ドメイン名の設定
- インストールが正常に行われたことの確認

JRE (Java Runtime Environment) 1.3.1 要件

グラフィカルユーザインタフェース (GUI) バージョンのエージェントインストールプログラムを実行するには、JRE (Java Runtime Environment) 1.3.1 がインストールされているか、共有ファイルシステムで利用できる必要があります。現在、エージェントのインストールプログラムでの使用が認定されているのは、バージョン 1.3.1 以降の JRE です。詳細は、43 ページの「コマンド行によるインストール」を参照してください。

Windows オペレーティングシステムを実行している場合、JRE 1.3.1 が検出されない場合は、インストールプログラムによって JRE 1.3.1 がインストールされます。

Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ

インストールプログラムを使用して、Sun ONE Identity Server がインストールされている Web サーバ上にポリシーエージェントをインストールできます。Sun ONE のマニュアルでは、このようなサーバは「Sun ONE Identity Server 6.0 が稼働する Web サーバ」と表記されます。また、インストールプログラムを使って組織内のリモート Web サーバにポリシーエージェントを追加インストールすることもできます。Sun ONE Identity Server の配備では、リモート Web サーバとは Sun ONE Identity Server を実際に実行しているサーバ以外の Web サーバを指します。これは、Sun ONE Identity Server 専用の Web サーバに対して「リモート」な関係にあります。

同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定

1 つのコンピュータシステムに複数の Web サーバまたはプロキシサーバがインストールされている場合、サーバまたはサーバインスタンスごとに異なるエージェントをインストールすることができます。

詳細については、「複数の Web サーバインスタンス用のエージェント設定」を参照してください。

注 Microsoft IIS サーバのインスタンスはコンピュータシステムごとに 1 つしかインストールできないので、同じコンピュータシステムに複数の Microsoft IIS エージェントをインストールすることはできません。

Sun ONE Identity Server エージェントのフェイルオーバ機能の提供

ポリシーエージェントのインストール時に、Sun ONE Identity Server を実行する Web サーバのフェイルオーバ、つまりバックアップを指定できます。この機能は基本的に、可用性を上げるためのオプションです。この機能を利用すると、Sun ONE Identity Server を実行する Web サーバが使用不能になった場合でも、エージェントは、Sun ONE Identity Server を実行しているセカンダリ Web (フェイルオーバ) サーバを通じてアクセス要求を処理できるようになります。

ポリシーエージェントのフェイルオーバ機能を設定するには、最初に、2 つの Web サーバに 2 つの Sun ONE Identity Server インスタンスをそれぞれインストールする必要があります。詳細な方法について『Sun ONE Identity Server インストールおよび設定ガイド』を参照し、このマニュアルで後述する手順に従って適切なエージェントをインストールしてください。エージェントのインストールプログラムは、Sun ONE Identity Server と連携するように設定したフェイルオーバ Web サーバのホスト名とポート番号を要求します。フェイルオーバサーバの名前とポートは、`AMAgent.properties` ファイルの次のプロパティに設定されます。

```
com.sun.am.policy.am.loginURL= http://primary_Identity
_Server.siroe.com:58080/amserver/UI/Login http://failover_Identity
_Server.siroe.com:58080/amserver/UI/Login
```

フェイルオーバサーバの名前は、インストール時に設定した後で変更できます。これは、このプロパティの 2 つ目の項目です。最初の項目は Sun ONE Identity Server のログイン URL で、各項目は空白文字で区切られています。

注　　プライマリ Web サーバとフェイルオーバ Web サーバで同じプロトコル (たとえば、`http` または `https`) を使用する必要があります。

エージェントキャッシュの更新

各エージェントは、各ユーザのセッションに適用されるポリシーを格納しているキャッシュを保持しています。キャッシュは、キャッシュ有効期限メカニズムまたは通知メカニズムのいずれかで更新できます。

キャッシュの更新

エージェントは、アクティブなすべてのセッションのキャッシュを保持しています。キャッシュにエントリが追加されると、有効期間の間そのエントリは有効であり、その期日が過ぎると削除されます。

エントリがエージェントキャッシュに存在する期間は、`AMAgent.properties` ファイルの `com.sun.am.policy.am.cacheEntryLifeTime` プロパティで分単位で指定されます。このプロパティによって定められた期間を経過すると、エントリはキャッシュから削除されます。デフォルトでは、3 分に設定されています。

キャッシュのハイブリッド更新

キャッシュの有効期間はこのモードにも適用されます。さらに、エージェントはセッションの変更について Sun ONE Identity Server サービスから通知を受けます。セッションの変更には、セッションログアウトやセッションタイムアウトなどのイベントがあります。セッションまたはポリシーの変更通知を受けると、エージェントはキャッシュにある該当エントリを更新します。セッションの更新とは別に、エージェントはポリシー変更の更新についても通知を受け取ります。ポリシーの変更には、ポリシーの更新、削除、作成などのイベントがあります。

デフォルトでは、Sun ONE Identity Server ポリシーエージェントのハイブリッド更新モードはオンです。このモードは、`AMAgent.properties` ファイルの `com.sun.am.policy.am.notificationEnabled` プロパティを `true` に設定することで有効になります。このプロパティを `false` に設定すると、エージェントによるキャッシュの更新はエントリの有効期間メカニズムだけで行われます。

ファイアウォール、使用中の Web サーバのタイプなどの制限により、状況によっては通知が許可されないこともあります。このような場合、エージェントによるキャッシュの更新はエントリの有効期間メカニズムだけで行われます。

注　次の場合は、通知のサポートは利用できません。

- IIS 4.0 または IIS 5.0 が HTTPS を使用している場合
- プラットフォームに関係なく、Apache 1.3.26 エージェントを使用している場合

グローバル不適用 URL リスト

グローバル不適用 URL リストは、関連するポリシー (許可または拒否) を設定すべきでないリソースを指定します。

デフォルトでは、ポリシーエージェントは、ポリシーエージェントが保護する Web サーバの全リソースへのアクセスを拒否します。ただし、Web サーバ (Web サイトやアプリケーションなど) を通じて利用できるさまざまなリソースにはポリシーを適用する必要がない場合があります。そのようなリソースの一般的な例として、Web サイトのホームページに見られる HTML ページや `.gif` 画像があります。そのようなページは、ユーザが認証なしで閲覧できる必要があります。これらのリソースは、グローバル不適用 URL リストに載せる必要があります。この設定には、`com.sun.am.policy.agents.notenforcedList` プロパティを使います。URL パターンの指定には、ワイルドカードを利用できます。リスト内の URL は、空白文字で区切られます。

これと反対の状況として、Web サーバ上の一部の URL を除くすべてのリソースに誰でもアクセスできる場合が考えられます。このような場合は `com.sun.am.policy.agents.reverse_the_meaning_of_notenforcedList` プロパティを使用すると、`com.sun.am.policy.agents.notenforcedList` プロパティの意味を逆転させることができます。このプロパティの値を `true` に設定すると (デフォルト値は `false`)、グローバル不適用 URL リストはグローバル適用 URL リストとなります。

次に例をあげます。

シナリオ 1

```
com.sun.am.policy.agents.reverse_the_meaning_of_notenforcedList=
false
```

```
com.sun.am.policy.agents.notenforcedList =
http://mycomputer.siroe.com:80/welcome.html
http://mycomputer.siroe.com:80/banner.html
```

この場合、`notenforcedList` に含まれる 2 つの URL に対して認証とポリシーは適用されません。これ以外のすべてのリソースはエージェントによって保護されます。

シナリオ 2

```
com.sun.am.policy.agents.reverse_the_meaning_of_notenforcedList=
true
```

```
com.sun.am.policy.agents.notenforcedList =
http://mycomputer.siroe.com:80/welcome.html
http://mycomputer.siroe.com:80/banner.html
```

この場合、notenforcedList に含まれる 2 つの URL に対して、エージェントによって認証とポリシーが適用されます。どのユーザも、これ以外のすべてのリソースにはアクセスできます。

グローバル不適用 IP アドレスリスト

IP アドレスのリストを設定するには、com.sun.am.policy.agents.notenforced_client_IP_address_list プロパティを使います。指定したクライアント IP アドレスからの要求には、認証が不要になります。

言い換えれば、エージェントは Web サーバ上のリソースに対してリストに指定された IP アドレスからのアクセスを防ぐことはできません。

ポリシーを適用しない認証だけの適用

エージェントが保護する URL に対して認証だけを適用するときは、com.sun.am.policy.agents.do_sso_only プロパティを使います。このプロパティを true に設定すると (デフォルト値は false)、エージェントはポリシーを適用せずに認証だけを適用します。ユーザが Identity Server にログインすると、そのユーザおよびユーザがアクセスする URL に関連するポリシーをエージェントは確認しません。

HTTP ヘッダーを介した LDAP ユーザ属性の転送

Sun ONE Identity Server のポリシーエージェントには、HTTP ヘッダーを介して LDAP ユーザ属性値をエンド Web アプリケーションに転送する機能があります。LDAP ユーザ属性値は、Sun ONE Identity Server のサーバサイドから引き継ぎます。Sun ONE Identity Server のポリシーエージェントはブローカのように動作し、ユーザ属性値を取得して転送先のサーブレット、CGI スクリプト、または ASP ページに中継します。次に、これらのアプリケーションは、その属性値を使用してページコンテンツをパーソナライズすることができます。

この機能は、AMAgent.properties ファイルの 2 つのプロパティで設定されます。この機能のオン、オフを切り替えるときは、AMAgent.properties ファイルの次のプロパティを使います。

com.sun.am.policy.am.fetchHeaders

デフォルトでは、このプロパティは false に設定されており、機能はオフになっています。属性の転送をオンにするには、このプロパティを true に設定します。HTTP ヘッダーで転送される属性を設定するには、AMAgent.properties ファイルの次のプロパティを使います。

com.sun.am.policy.am.headerAttributes

次の例は、この機能の使い方を示す AMAgent.properties ファイルの一部です。

```
#
# The policy attributes to be added to the HTTP header.  The
# specification is of the format
# ldap_attribute_name|http_header_name[,...]. ldap_attribute_name
# is the attribute in data store to be fetched and
# http_header_name is the name of the header to which the value
# needs to be assigned.
#
# NOTE: In most cases, in a destination application where a
# "http_header_name" shows up as a request header, it will be
# prefixed by HTTP_, and all lower case letters will become upper
# case, and any - will become _; For example, "common-name" would
# become "HTTP_COMMON_NAME"
#
com.sun.am.policy.am.headerAttributes=cn|common-name,ou|organiza
tional-unit,o|organization,mail|email,employeenumber|employee-nu
mber,c|country
```

デフォルトでは、一部の LDAP ユーザ属性名および HTTP ヘッダー名は、サンプル値が設定されています。

適切な LDAP ユーザ属性名を見つけるには、Sun ONE Identity Server サーバがインストールされているマシンで次の XML ファイルを調べます。

S1IS_Install_Dir/SUNWam/config/xml/amUser.xml

このファイルに設定されている属性は、Sun ONE Identity Server の User 属性または Dynamic 属性です。この 2 種類のユーザ属性については、『Sun ONE Identity Server Administration Guide』を参照してください。

転送が必要な属性名および HTTP ヘッダー名は、エージェントが保護している Web サーバ上のエンドユーザアプリケーションが決定する必要があります。これは、これらのアプリケーションは転送されるヘッダー値のコンシューマであるためです。転送される情報は、Web ページのカスタマイズとパーソナライズに使用されます。

注　この機能は、Sun ONE Web Proxy Server エージェントには使用できません。

AMAgent.properties ファイル

AMAgent.properties ファイルは、ポリシーエージェントが使用する設定パラメータを格納します。このファイルのデフォルトのパラメータを変更する必要がある場合があります。たとえば、Sun ONE Identity Server を稼働する別のフェイルオーバ Web サーバを指定する場合などです。

AMAgent.properties ファイルには、次の設定を行うための情報が含まれています。

- デバッグ
- ポリシーエージェント
- FQDN マップ
- Sun ONE Identity Server サービス
- サービスおよびエージェントの配備記述子
- セッションのフェイルオーバ

また、AMAgent.properties ファイルには、HTTP ヘッダーによる LDAP ユーザ属性の転送、POST データの保存など、詳細な機能に関する設定情報も含まれています。AMAgent.properties ファイルには各プロパティの前にコメントがあるので、詳細はファイルを参照してください。

表 1-1 は、サポートされるさまざまなサーバの AMAgent.properties のデフォルトの場所を示しています。

表 1-1　プラットフォームごとの AMAgent.properties の場所

サーバ	場所
サポートされるすべての UNIX Web サーバ	/etc/opt/SUNWam/agents/*WebServer*/config/*_PathInstanceName*/ *WebServer* には次のものがあります。 •es6 •es4 •proxy •apache
Sun ONE Web Server 6.0 Windows 2000	¥*Agent_Install_Dir*¥Identity_Server¥Agents¥2.0¥es6¥config¥*_PathInstanceName*¥

表 1-1　プラットフォームごとの AMAgent.properties の場所 (続き)

サーバ	場所
Microsoft IIS 5.0 Windows 2000	¥*Agent_Install_Dir*¥Identity_Server¥Agents¥iis¥config¥*_PathInstance Name*¥
Microsoft IIS 4.0 Windows NT	¥*Agent_Install_Dir*¥Identity_Server¥Agents¥iis¥config¥*_PathInstance Name*¥
Apache 1.3.26 Red Hat Linux 7.2	/etc/opt/agents/apache/config/*_PathInstanceName*/
Apache 1.3.26 Solaris 8、9	/etc/opt/SUNWam/agents/apache/config/*_PathInstanceName*/

AMAgent.properties ファイルを変更すると、重大かつ広範囲に及ぶ影響が出る可能性があります。このファイルのプロパティの多くは、エージェントをインストールし直すだけで安全に変更できることを忘れないでください。ただし、手動で変更する必要がある場合は、次の点を念頭に置いてください。

- 変更を行う前にこのファイルのバックアップコピーを作成する
- 後続の空白文字には重要な意味があるので、慎重に使用する
- ディレクトリの区切りには、円記号 (¥) ではなくスラッシュ (/) を使用する。これは Windows システムにも当てはまる
- Windows ファイル名に空白文字を使用できる

注　AMAgent.properties ファイルに変更を加える場合は、Web サーバを再起動して変更内容を有効にする必要があります。

完全指定ドメイン名の設定

ユーザが適切な結果を得るには、ユーザが有効な URL を使ってエージェントによって保護されるリソースにアクセスする必要があります。保護されたリソースへのアクセスに、ユーザが有効な URL を使っているかどうかをエージェントが確認する上で必要な情報は、com.sun.am.policy.agents.fqdnDefault プロパティによって設定されます。要求に含まれる URL に有効なホスト名が指定されていないことを検出すると、エージェントは有効なホスト名が設定された URL にユーザをリダイレクトします。リダイレクト先の URL と、ユーザが最初に使っていた URL の違いはホスト名だけです。エージェントは、このプロパティの設定に基づいて、ホスト名を完全指定ドメイン名 (FQDN) に変換します。

この設定プロパティは必須です。これなしでは、Web サーバが正しく起動しないことがあります。このプロパティは、エージェントのインストール時に設定され、配備要件に対応する上でどうしても必要な場合を除き、変更する必要はありません。このプロパティに無効な値を設定すると、Web サーバを利用できなくなったり、リソースにアクセスできなくなることがあります。

ユーザがアクセスするために指定した URL に問題がある場合に、エージェントがそれを解決するには、com.sun.am.policy.agents.fqdnMap プロパティを使用することもできます。エージェントは、com.sun.am.policy.agents.fqdnDefault プロパティに定義されている値よりも優先してこのプロパティの値を適用します。このプロパティに設定されている項目と一致するホスト名がユーザ要求に含まれない場合は、エージェントは com.sun.am.policy.agents.fqdnDefault プロパティの値を適用します。

com.sun.am.policy.agents.fqdnMap プロパティを使って、複数のホスト名のマップを作成することができます。これは、このエージェントが保護する Web サーバに対して複数のホスト名でアクセスする場合などが対象となります。ただし、Web サーバ上のリソースにアクセスできなくなることもあるため、この機能の利用には注意が必要です。

このプロパティは、必要に応じてエージェントの動作を無効にする場合にも利用できます。たとえば、純粋な IP アドレスを使って Web サーバ上のリソースにアクセスするユーザに対しては、リダイレクトなどの対策を適用しない場合は、次のようなマップエントリを指定します。

com.sun.am.policy.agents.fqdnMap=*IP*|*IP*

com.sun.am.policy.agents.fqdnMap プロパティの書式は次のとおりです。

```
com.sun.am.policy.agents.fqdnMap =
[invalid_hostname|valid_hostname][,...]
```

変数の意味は次のとおりです。

invalid_hostname は、ホスト名の一部や IP アドレスなど、ユーザにより指定される可能性のある無効なホスト名です。

valid_hostname は、対応する有効なホスト名で、完全指定名で指定します。たとえば、xyz.domain1.com というホスト名については、次のような値の設定が考えられます。

```
com.sun.am.policy.agents.fqdnMap = xyz|xyz.domain1.com,
xyz.domain1|xyz.domain1.com
```

これにより、xyz および xyz.domain1 は、xyz.domain1.com という完全指定名にマッピングされます。

CDSSO の設定

CDSSO (Cross Domain Single Sign-On) 機能は、`AMAgent.properties` ファイルの 2 つのプロパティを使って設定されます。この機能のオン、オフを切り替えるときは、`AMAgent.properties` ファイルの次のプロパティを使います。

```
com.sun.am.policy.agents.cdsso-enabled=true
```

デフォルトでは、このプロパティは `false` に設定されており、機能はオフになっています。CDSSO 機能をオンにするには、このプロパティを `true` に設定します。CDSSO がインストールされている URL は、次のプロパティに指定します。

```
com.sun.am.policy.agents.loginURL =
http://mycomputer.domain:port/amcdsso/cdsso
```

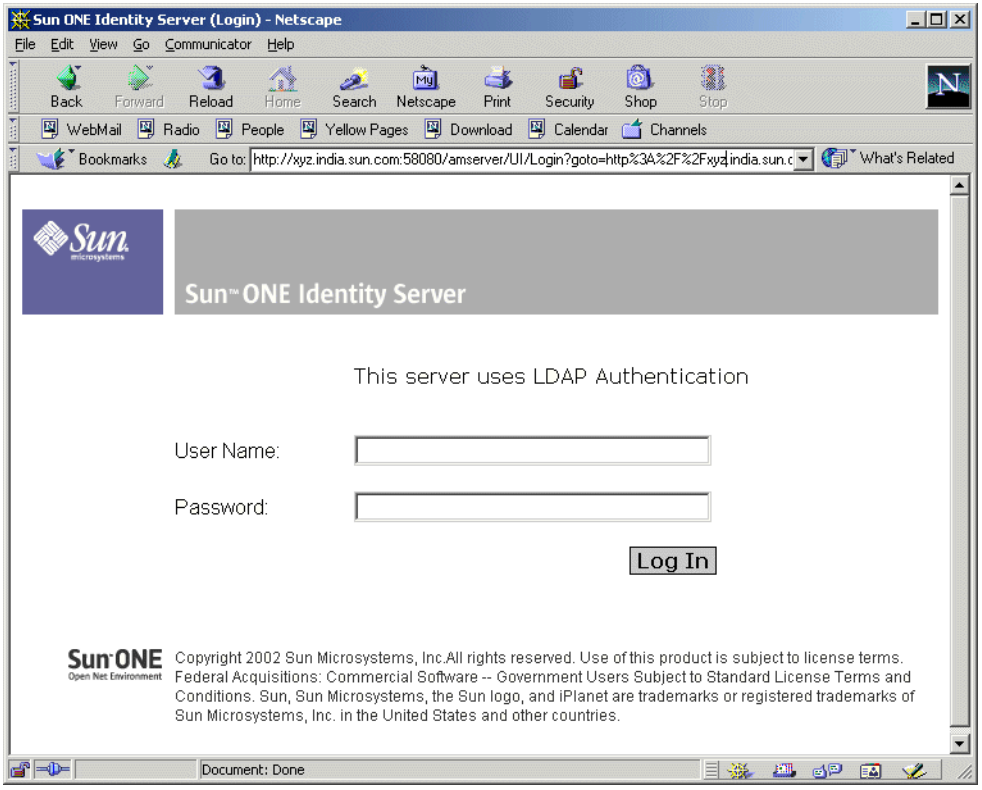
CDSSO コンポーネントの設定については、『Sun ONE Identity Server インストールおよび設定ガイド』を参照してください。

インストールが正常に行われたことの確認

ポリシーエージェントをインストールした後、エージェントが正常にインストールされ、期待通りに動作することを確認するようお勧めします。エージェントが正常にインストールされたことを確認できる点は 2 つあります。

1. エージェントがインストールされている Web サーバの Web コンテンツにアクセスしてみます。エージェントが正しくインストールされていれば、Sun ONE Identity Server のログインページが表示されます。図 1-2 は、LDAP 認証を使用している Sun ONE Identity Server のログインページの例です。
2. `AMAgent.properties` ファイルを調べます。各プロパティが適切に設定されていることを確認します。

図 1-2　Sun ONE Identity Server のログインページ



第 2 章

Solaris 8 および 9 のポリシーエージェント

Sun ONE Identity Server のポリシーエージェントは、Sun ONE Identity Server と連携して企業の Web サーバに対するユーザアクセスの可否を制御します。この章では、Solaris 8、Solaris 9 オペレーティングシステムで稼動するサーバでポリシーエージェントをインストールおよび設定する方法を説明します。

次のトピックがあります。

- 始める前に
- グラフィカルユーザインタフェースによるインストール
- コマンド行によるインストール
- 複数の Web サーバインスタンス用のエージェント設定
- SSL (Secure Sockets Layer) とエージェントの使用
- REMOTE_USER サーバ変数の設定
- クライアント IP アドレスの検証
- POST データの保存
- 共有シークレットの暗号化ユーティリティ
- Solaris エージェントのトラブルシューティング

始める前に

第 1 章「ご使用にあたって」で説明されている概念を良く理解しておいてください。この章には、次のトピックスに関する簡単ですが重要な情報があります。

- ポリシーエージェントの動作
- JRE (Java Runtime Environment) 1.3.1 要件
- Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ
- 同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定
- Sun ONE Identity Server エージェントのフェイルオーバ機能の提供
- エージェントキャッシュの更新
- グローバル不適用 URL リスト
- グローバル不適用 IP アドレスリスト
- ポリシーを適用しない認証だけの適用
- HTTP ヘッダーを介した LDAP ユーザ属性の転送
- `AMAgent.properties` ファイル
- 完全指定ドメイン名の設定
- CDSSO の設定

サポートされる Solaris Web サーバ

Sun ONE Identity Server のポリシーエージェントは、Solaris 8 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Sun ONE Web Server 6.0 SP*x*
- Sun ONE Web Server 4.1 SP8
- Sun ONE Web Proxy Server 3.6 (リバースプロキシモード)
- Apache 1.3.26

Sun ONE Identity Server Policy Agent は、Solaris 9 オペレーティングシステム上で稼動する次のサーバをサポートします。

- Sun ONE Web Server 6.0 SP*x*
- Apache 1.3.26

Solaris のパッチクラスタ

Solaris 8、9 プラットフォームで Apache 1.3.26 Web Server を稼働している場合は、推奨パッチ 109234-09 および 113146-01 をインストールしておく必要があります。これらのパッチは、`http://sunsolve.sun.com` からダウンロードできます。

グラフィカルユーザインタフェースによるインストール

Web サーバのポリシーエージェントのインストール

Solaris 8 オペレーティングシステムを使用する次のサーバにエージェントをインストールするには、次の手順を実行します。

- Sun ONE Web Server 4.1 SP8
- Sun ONE Web Server 6.0 SP*x*
- Apache Web Server 1.3.26

Solaris 9 オペレーティングシステムを使用する次のサーバにエージェントをインストールするには、次の手順を実行します。

- Sun ONE Web Server 6.0 SP*x*
- Apache Web Server 1.3.26

Web サーバのポリシーエージェントをインストールするには

エージェントのインストールプログラムを実行するには、root 権限が必要です。

1. 次のコマンドを使って製品のバイナリファイルを解凍します。

   Sun ONE Web Server 4.1

   ```
   # gunzip -dc agent_SunOS_es41.tar.gz| tar -xvof -
   ```

   Sun ONE Web Server 6.0 SP*x*

   ```
   # gunzip -dc agent_SunOS_es6.tar.gz| tar -xvof -
   ```

   Apache 1.3.26 Web Server

   ```
   # gunzip -dc agent_SunOS_apache.tar.gz| tar -xvof -
   ```

2. `setup` プログラムを実行します。このプログラムは、バイナリファイルを解凍したディレクトリにあります。コマンド行に次のコマンドを入力します。

   ```
   # ./setup
   ```

3. 開始ページで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。
5. プロンプトに従って、このエージェントが保護する Web サーバに関する次の情報を入力します。

   Install Sun ONE Identity Server Policy Agent in this directory: このエージェントをインストールするディレクトリの完全パスを入力して、「Next」をクリックします。

   Host Name: Web サーバがインストールされているマシンの完全指定のドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   Web Server Instance Directory: このプロンプトは、Sun ONE Web Server エージェントを選択した場合だけ表示されます。このエージェントが保護する Web サーバのインスタンスを指定します。Web サーバのインスタンスのあるディレクトリの完全パスを入力します。次に例を示します。

/*web_server_root*/https-mycomputer.siroe.com

Apache Configuration Directory: このプロンプトは、Apache Web サーバエージェントをインストールする場合だけ表示されます。httpd.conf ファイルが保存されている Apache サーバ設定ディレクトリを指定します。

Web Server Port: エージェントで保護する Web サーバのポート番号を入力します。

Web Server Protocol: Web サーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

Agent Deployment URI: ディレクトリ名を入力します。デフォルトの URI (Universal Resource Identifier) は /amagent です。

注 エージェントは、通知や POST データの保存など、一部の重要機能の実行に com.sun.am.policy.agents.agenturiprefix プロパティの値を使います。このプロパティに有効な URL を設定することが重要です。このプロパティのデフォルト値は次のとおりです。

http://*host.domain*:*port*/*agent_deployment_uri*

host、*domain*、*port* はエージェントがインストールされている Web サーバの完全指定ドメイン名とポート番号です。*agent_deployment_uri* は URI の接頭辞で、エージェントに関連する HTML ページの検索場所を Web サーバに指定します。デフォルト値は amagent です。

SSL Ready: このオプションは、Apache Web サーバエージェントをインストールするときだけ表示されます。Apache Web サーバが SSL をサポートしている場合は、このオプションを選択します。mod_ssl をサポートし、EAPI ルールを使ってソースがコンパイルされている Apache Web サーバは、SSL 対応と見なされます。

Apache Web サーバのコンパイルに EAPI フラグが使われているかどうかを調べるときは、Apache Web サーバの bin ディレクトリに移動して、次のコマンドを実行します。

```
# ./httpd -V
```

Apache Web サーバがコンパイルされた各種フラグが表示されます。この一覧に -D EAPI というフラグが含まれる場合は、使用中の Apache Web サーバは SSL をサポートしています。ただし、このフラグが見つからない場合でも、Apache Web サーバが mod_ssl に対応していることがあります。

Apache Web サーバの次の設定がサポートされています。

a. mod_ssl に対応していない Apache Web サーバ

b. mod_ssl に対応し、EAPI フラグが有効な Appache Web サーバ

注 Sun ONE Identity Server ポリシーエージェントは、mod_ssl に対応していても EAPI フラグが無効な Apache Web サーバをサポートしていません。

すべての情報を正しく入力したら、「Next」をクリックします。

6. Sun ONE Identity Server のポリシーおよび管理機能が稼動する Web サーバに関する情報を入力します。ポリシーエージェントはこのサーバに接続します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定のドメイン名を入力します。たとえば、myserver.siroe.com などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は /amserver です。

   Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は /amconsole です。

   Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定のドメイン名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は /amserver です。

   Failover Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は /amconsole です。

   Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

   Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

すべての情報を正しく入力したら、「Next」をクリックします。

7. 「Installation Summary」を見直して、入力した情報が正しいことを確認します。変更が必要な場合は、「Back」をクリックします。すべての情報を正しく入力したら、「Next」をクリックします。
8. 「Ready to Install」のページで、「Install Now」をクリックします。
9. インストールが終了したら、「Details」をクリックしてインストールの詳細を確認するか、「Exit」をクリックしてインストールプログラムを終了します。
10. インストールを完了するには、Sun ONE Web Server または Apache Web サーバを再起動する必要があります。

プロキシサーバのポリシーエージェントのインストール

Sun ONE Web Proxy Server 3.6 (リバースプロキシモード) を Solaris 8 オペレーティングシステムにインストールするには、次の手順を実行します。

プロキシサーバのポリシーエージェントをインストールするには

エージェントのインストールプログラムを実行するには、root 権限が必要です。

1. 次のコマンドを使って製品のバイナリファイルを解凍します。

   ```
   # gunzip -dc agent_SunOS_proxy.tar.gz | tar -xvof -
   ```

2. `setup` プログラムを実行します。このプログラムは、バイナリファイルを解凍したディレクトリにあります。コマンド行に次のコマンドを入力します。

   ```
   # ./setup
   ```

3. 開始ページで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。

5. プロンプトに従って、このエージェントをインストールする Web プロキシサーバに関する次の情報を入力します。

   Host Name: リモート Web サーバがインストールされているシステムの完全指定のドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   Proxy Server Instance Directory: Sun ONE Web Proxy Server のインスタンスがあるディレクトリの完全パスを入力します。次に例を示します。

   proxy_server_root_dir`/proxy-`*mycomputer*`-proxy`

   Proxy Server Port: プロキシサーバインスタンスのポート番号を入力します。

   Proxy Server Protocol: プロキシサーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

   Agent Deployment URI: ディレクトリ名を入力します。デフォルト URI は `/amagent` です。

   すべての情報を正しく入力したら、「Next」をクリックします。

6. プロンプトに従って、Sun ONE Identity Server を実行する Web サーバに関する次の情報を入力します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定のドメイン名を入力します。たとえば、`myserver.siroe.com` などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

   Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

   Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定のドメイン名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

Failover Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

すべての情報を正しく入力したら、「Next」をクリックします。

7. 「Installation Summary」を見直して、入力した情報が正しいことを確認します。変更が必要な場合は、「Back」をクリックします。すべての情報を正しく入力したら、「Next」をクリックします。
8. 「Ready to Install」のページで、「Install Now」をクリックします。
9. インストールが終了したら、「Details」をクリックしてインストールの詳細を確認するか、「Exit」をクリックしてインストールプログラムを終了します。
10. プロキシサーバを再起動します。

Web サーバのポリシーエージェントのアンインストール

エージェントをアンインストールするには、アンインストールプログラムを実行する必要があります。次の手順に従ってください。

1. エージェントがインストールされているディレクトリに移動して、コマンド行に次のコマンドを入力します。

   ```
   # ./uninstall_agent
   ```

2. 開始パネルで、「Next」をクリックします。
3. 「Ready to Uninstall Panel」パネルの「Uninstall Now」をクリックします。
4. アンインストールが終了したら、「Close」をクリックします。

製品のバイナリファイルを解凍した (およびインストールプログラムを起動した) ディレクトリには、`uninstall` という名前のアンインストールプログラムがもうひとつあります。`uninstall` プログラムを使えば、`setup` プログラムを使用して前にインストールしたすべてのエージェント、およびリモートマシンにインストールしたエージェントを検出してアンインストールできます。次のコマンドを使用して `uninstall` を起動します。

```
# uninstall
```

これに対して、`uninstall_agent` は、現在のディレクトリの `setup` プログラムを使って前にインストールしたエージェントだけをアンインストールします。

コマンド行によるインストール

グラフィカルユーザインタフェース (GUI) バージョンの代わりに、コマンド行バージョンのインストールプログラムを使用することもできます。

コマンド行を使って Web サーバのエージェントをインストールするには

1. バイナリファイルを解凍したディレクトリに移動して、コマンド行に次のコマンドを入力します。

   ```
   # setup -nodisplay
   ```

2. プロンプトが表示されたら、次の情報を入力します。

   Have you read, and do you accept, all of the terms of the preceding Software License Agreement?: 「yes」と入力します。

   Install Sun ONE Identity Server Agent in this directory: ポリシーエージェントを入力するディレクトリの完全パスを入力します。

3. このエージェントが保護する Web サーバに関する次の情報を入力します。

   - `Host Name`
   - `Port`
   - `Web Server Instance Directory`
   - `Web Server Protocol`
   - `Agent Deployment URI`
   - `SSL Ready` (Apache エージェントのインストール時のみ)

   これらの各項目の詳細は、「Web サーバのポリシーエージェントのインストール」を参照してください。

4. Sun ONE Identity Server を実行する Web サーバに関する次の情報を入力します。

   - `Primary Server Host`
   - `Primary Server Port`
   - `Primary Server Protocol`
   - `Primary Server Deployment URI`
   - `Primary Console Deployment URI`
   - `Failover Server Host`
   - `Failover Server Port`

- Failover Server Deployment URI
- Failover Console Deployment URI
- Agent-Identity Server Shared secret
- Re-enter Shared secret
- CDSSO Enabled
- CDSSO Component URL

これらの各項目の詳細は、「Web サーバのポリシーエージェントのインストール」を参照してください。

5. 次のメッセージが表示されます。

```
Ready to Install

1. Install Now
2. Start Over
3. Exit Installation
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

6. 次のメッセージが表示されます。

```
Product                          Result     More Information
1. Sun ONE Identity Server Agent  Installed  Available
2. Done
```

ログ情報を表示するときは、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

コマンド行を使って Web プロキシサーバのエージェントをインストールするには

1. バイナリファイルを解凍したディレクトリに移動して、コマンド行に次のコマンドを入力します。

   ```
   # setup -nodisplay
   ```

2. プロンプトが表示されたら、次の情報を入力します。

   Have you read, and do you accept, all of the terms of the preceding Software License Agreement?: 「yes」と入力します。

   Install Sun ONE Identity Server Agent in this directory: ポリシーエージェントを入力するディレクトリの完全パスを入力します。

3. このエージェントが保護する Web サーバに関する次の情報を入力します。

   - `Host Name`
   - `Proxy Server Instance Directory`
   - `Proxy Server Port`
   - `Proxy Server Protocol`
   - `Agent Deployment URI`

   これらの各項目の詳細は、「Web サーバのポリシーエージェントのインストール」を参照してください。

4. Sun ONE Identity Server を実行するプロキシ Web サーバに関する次の情報を入力します。

   - `Primary Server Host`
   - `Primary Server Port`
   - `Primary Server Protocol`
   - `Primary Server Deployment URI`
   - `Primary Console Deployment URI`
   - `Failover Server Host`
   - `Failover Server Port`
   - `Failover Server Deployment URI`
   - `Failover Console Deployment URI`
   - `Agent-Identity Server Shared secret`
   - `Re-enter Shared secret`
   - `CDSSO Enabled`

- CDSSO Component URL

これらの各項目の詳細は、「プロキシサーバのポリシーエージェントのインストール」を参照してください。

5. 次のメッセージが表示されます。

```
Ready to Install

1. Install Now
2. Start Over
3. Exit Installation
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

6. 次のメッセージが表示されます。

```
Product                          Result     More Information
1. Sun ONE Identity Server Agent  Installed  Available
2. Done
```

ログ情報を表示するときは、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

コマンド行を使ってエージェントをアンインストールするには

1. エージェントがインストールされているディレクトリから、コマンド行に次のコマンドを入力します。

   ```
   # ./uninstall_agent -nodisplay
   ```

   アンインストーラは、setup プログラムを使ってインストールされているエージェントを検出します。エージェントをアンインストールするには、1 を入力します。

2. 次のメッセージが表示されます。

```
Ready to Uninstall

1. Uninstall Now
2. Start Over
3. Exit Uninstallation
```

   「What next?」というプロンプトが表示されたら、1 を入力してアンインストールを開始します。

3. 次のメッセージが表示されます。

```
Product                           Result  More Information
1. Sun ONE Identity Server Agent  Full    Available
2. Done
```

   ログ情報を表示するには、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

複数の Web サーバインスタンス用のエージェント設定

1 台のコンピュータ上で稼働する複数の Web サーバインスタンス用にエージェントを設定するには、グラフィカルユーザインタフェース (GUI)、またはコマンド行でエージェントインストールプログラムを使って最初のエージェントをインストールします。最初のエージェントをインストールしたら、config スクリプトを使って残りのエージェントをインストールできます。このスクリプトは、次の節で説明するように、コマンド行から実行する必要があります。

今回のリリースでは、同じマシンに複数の種類のエージェントをインストールすることはできません。たとえば、Apache エージェントと Sun ONE Web Server エージェントは同じマシンにインストールできません。

同じコンピュータシステムに複数の Web サーバインスタンス用のエージェントを設定するには

システムにエージェントを 1 つインストールしたら、エージェントのインストール時にシステムにコピーされるスクリプトを使用して、残りのエージェントをそのシステムにインストールできます。次のディレクトリに、2 つのスクリプト、config と unconfig があります。

Agent_Install_Dir/SUNWam/agents/*WS_TYPE*/bin

WS_TYPE は、エージェントが保護している Web サーバの種類に応じて、es6、es4、proxy、または apache となります。

元のエージェントをインストールした後で追加のエージェントをシステムにインストールするには、次のコマンドを使って、bin ディレクトリから config スクリプトを実行します。

```
# ./config
```

プロンプトに従って、追加のエージェントをインストールします。各プロンプトについては、「Web サーバのポリシーエージェントのインストール」を参照してください。一般に、保護された Web サーバインスタンスと Sun ONE Identity Server サーバの両方の情報を入力する必要があります。次のテキストは実行例です。

```
# ./config
Enter the Web Server Instance Directory:
[/web_server_root/https-server_instance]
Enter the Local Hostname:[mycomputer.siroe.com]
Enter the Agent Web Server Port: [80]
Select Agent Web Server Protocol:[1] http [2] https-->[1]
```

```
Enter the Agent Deployment URI:[/amagent]
Select Identity Server Protocol:[1] http [2] https --> [1]
Enter the Identity Server Hostname:[mycomputer.siroe.com]
Enter the Identity Server Port: [58080]
Enter the Identity Server Deployment URI [/amserver]
Enter the Identity Server's Console Deployment URI [/amconsole]
Enter the Identity Server's Console Deployment URI [/amconsole]
Select Failover Identity Server Protocol: [1] http [2] https [3]
no failover --> []
Enter the Failover Identity Server Hostname:
[]mycomputer.siroe.com
Enter the Failover Identity Server Port: []
Enter the Identity Server Deployment URI [/amserver]
Enter the Identity Server's Console Deployment URI [/amconsole]
Enter Agent-Identity Server shared secret:
Re-enter Agent-Identity Server shared secret:
Configuring webserver ... Webserver version:  6.0
Done
```

注 config スクリプトでは、CDSSO の詳細を入力できません。これは、手動で設定する必要があります。詳細については、「CDSSO の設定」を参照してください。

config スクリプトによるサイレントインストール

config スクリプトを使用して、サイレントの非対話型エージェントインストールを実行することもできます。このスクリプトの使用方法については、config -h コマンドを使用して詳細を表示してください。

```
# ./config -h
Usage:config [ -r response_file | -R | -h ]
        -r specifies a response file.
        -R prints out the response file template.
        -h prints out this message.
```

サイレントインストールを実行するには、インストールするエージェントごとに応答ファイルを指定する必要があります。config -R コマンドは、応答ファイルで指定しなければならないフィールドを示します。このテキストファイルは、サイレントインストールを開始する前に用意しておく必要があります。

```
./config -R
Response file contains:
AGENT_PROTOCOL              # agent protocol:http|https
AGENT_HOST                  # agent hostname
AGENT_PORT                  # agent server port
AGENT_DEPLOY_URI            # agent deploy URI
FAILOVER_SERVER_HOST        # failover identity server name
FAILOVER_SERVER_PORT        # failover identity server port
FAILOVER_SERVER_DEPLOY_URI # failover identity server deploy URI
FAILOVER_CONSOLE_DEPLOY_URI # failover identity server console deploy URI
PRIMARY_SERVER_HOST         # primary identity server name
PRIMARY_SERVER_PORT         # primary identity server port
PRIMARY_SERVER_PROTO        # primary identity server protocol:http|https
PRIMARY_SERVER_DEPLOY_URI   # primary identity server deploy URI
PRIMARY_CONSOLE_DEPLOY_URI # primary identity server console deploy URI
SHARED_SECRET               # shared secret between agent and DSAME server
SERVER_INSTANCE             # web server instance directory
NOTIFICATION_ENABLE         # notification enabled
AGENT_URL_CASE_IGNORE       # url comparison case ignore
```

次に、response.iws60 という名前の応答ファイルの例を示します。

```
AGENT_PROTOCOL=http
AGENT_HOST=mycomputer.siroe.com
AGENT_PORT=80
AGENT_DEPLOY_URI=/amagent
FAILOVER_SERVER_HOST=failover_computer.siroe.com
FAILOVER_SERVER_PORT=58080
FAILOVER_SERVER_DEPLOY_URI=/amserver
FAILOVER_CONSOLE_DEPLOY_URI=/amconsole
PRIMARY_SERVER_HOST=primary_computer.siroe.com
PRIMARY_SERVER_PORT=58080
PRIMARY_SERVER_PROTO=http
PRIMARY_SERVER_DEPLOY_URI=/amserver
PRIMARY_CONSOLE_DEPLOY_URI=/amconsole
SHARED_SECRET=encrypted_shared_secret
SERVER_INSTANCE=/opt/iws6a/https-mycomputer.siroe.com
NOTIFICATION_ENABLE=true
AGENT_URL_CASE_IGNORE=true
```

次の例は、response.iws60 応答ファイルと組み合わせて config スクリプトを使って、サイレントインストールを実行する方法を示しています。

```
# ./config -r ./response.iws60
Configuring webserver ... Webserver version:  60
done
```

注 config スクリプトを使ってインストールしたエージェントは、必ず unconfig スクリプトを使用してアンインストールしてください。コマンド行からインストールしたエージェントのアンインストールに GUI インストールプログラムを使うことはできません。GUI のアンインストールプログラムは、コマンド行で unconfig スクリプトを使ってインストールしたすべてのエージェントをアンインストールしてから実行する必要があります。

unconfig スクリプトによるエージェントの削除

config スクリプトを使ってコマンド行からインストールしたエージェントを削除するには、unconfig スクリプトを使います。unconfig スクリプトは、次のディレクトリにあります。

Agent_Install_Dir/SUNWam/agents/*WS_TYPE*/bin

WS_TYPE は、エージェントが保護している Web サーバの種類に応じて、es6、es4、proxy、または apache となります。

次に、unconfig スクリプトの実行例を示します。

```
# ./unconfig /web_server_root/https-server_instance
Unconfiguring webserver ...
done.
```

SSL (Secure Sockets Layer) とエージェントの使用

インストール時に HTTPS プロトコルを選択すると、エージェントが自動的に設定されて、SSL を介して通信する用意ができます。

注

次の手順に進む前に、Web サーバに SSL が設定されていることを確認してください。

HTTPS プロトコルを介した通信を使用可能にするのに必要な SSL の概念およびセキュリティ証明書をよく理解しておく必要があります。Web サーバに付属のマニュアルを参照してください。Sun ONE Web Server を使用している場合は、次のサイトで参照できます。

`http://docs.sun.com/source/816-5682-10/esecurty.htm#1011961`

SSL モードで稼働している Web サーバまたは Web プロキシサーバ

Web サーバまたは Web プロキシサーバを SSL モードで稼働し、エージェントが通知モードの場合は、Web サーバまたは Web プロキシサーバのルート証明書を Sun ONE Identity Server にインストールする必要があります (インストールされていない場合のみ)。

エージェントのデフォルトの信頼動作

デフォルトでは、リモートの Web サーバまたは Web プロキシサーバにインストールされたポリシーエージェントは、Sun ONE Identity Server を実行する Web サーバが SSL 上で提示したサーバ認証書を信頼します。エージェントはルートの認証局 (CA) 証明書をチェックしません。Sun ONE Identity Server を実行する Web サーバで SSL が有効になっていて、ポリシーエージェントで証明書をチェックしたい場合は、次の処理を行う必要があります。

1. エージェントのデフォルトの信頼動作を無効にします。
2. エージェントがインストールされているリモート Web サーバにルート CA 証明書をインストールします。ルート CA 証明書は、Sun ONE Identity Server サービスを実行する Web サーバにインストールされているものと同じでなければなりません。

エージェントのデフォルト信頼動作の無効化

次のプロパティは AMAgent.properites ファイルにあり、デフォルトでは true に設定されています。

```
com.sun.am.policy.agents.trustServerCerts=true
```

これは、エージェントが証明書のチェックを行わないということです。

デフォルトの動作を無効にするには

次のプロパティを false に設定する必要があります。

```
com.sun.am.policy.agents.trustServerCerts=false
```

リモート Web サーバへのルート CA 証明書のインストール

リモート Web サーバにインストールするルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

ルート CA 証明書を Sun ONE Web Server にインストールするには

Web サーバに付属のマニュアルにあるルート CA 証明書をインストールするための手順を参照してください。一般に、ルート CA 証明書は Web サーバの管理コンソールからインストールします。

Sun ONE Web Server 6.0 のマニュアルは、インターネットの次の URL からアクセスできます。

```
http://docs.sun.com/source/816-5682-10/esecurty.htm#1011961
```

ルート CA 証明書を Sun ONE Web Server 4.1 SP8 にインストールする手順も同様です。

ルート CA 証明書を Apache 1.3.26 にインストールするには

certutil プログラムを使って、ルート CA 証明書を Apache 1.3.26 にインストールできます。

1. C シェルで、コマンド行に次のコマンドを入力します (設定ファイルが保存されているディレクトリを /etc/apache とします)。

   ```
   # cd /etc/apache/cert
   ```

```
# setenv LD_LIBRARY_PATH
/Agent_Install_Dir/SUNWam/agents/apache/lib:/Agent_Install_Dir/SUNWam/agents/lib
```

2. 必要に応じて証明書データベースを作成します。

   ```
   # /Agent_Install_Dir/SUNWam/agents/apache/cert/certutil -N -d .
   ```

3. ルート CA 証明書をインストールします。

   ```
   # /Agent_Install_Dir/SUNWam/agents/apache/cert/certutil -A -n cert-name
   -t "C,C,C" -d cert-dir -i cert-file
   ```

   上のコマンドの各変数の意味は、次のとおりです。

   - *cert-name* には、このルート証明書の任意の名前を指定します。
   - *cert-dir* には、証明書関連のファイルが置かれているディレクトリを指定します。
   - *cert-file* には、Base64 で符号化されたルート証明書ファイルを指定します。

   `certutil` ユーティリティの詳細は、「`certutil -H`」と入力してオンラインヘルプを参照してください。

4. 証明書が正しくインストールされたことを確認するには、コマンド行に次のように入力します。

   ```
   # ./certutil -L -d .
   ```

   インストールしたルート CA 証明書の名前を含む信頼データベース情報が表示されます。次に例を示します。

```
Certificate Name                                Trust Attrubutes

cert-name                                       C,C,C

p    Valid peer
P    Trusted peer (implies c)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

ルート CA 証明書を Web プロキシサーバにインストールするには

certutil プログラムを使って、ルート CA 証明書を Web プロキシサーバにインストールできます。

1. C シェルで、コマンド行に次のコマンドを入力します。

   ```
   # mkdir Proxy_Server_Instance_Dir/cert
   ```

   ```
   # cd Proxy_Server_Instance_Dir/cert
   ```

   ```
   # setenv LD_LIBRARY_PATH
   /Agent_Install_Dir/SUNWam/agents/proxy/lib:/Agent_Install_Dir/SUNWam/agents/lib
   ```

2. 必要に応じて証明書データベースを作成します。

3. ルート CA 証明書をインストールします。

   ```
   # /Agent_Install_Dir/SUNWam/agents/proxy/cert/certutil  -N -d .
   ```

   ```
   # /Agent_Install_Dir/SUNWam/agents/proxy/cert/certutil -A -n cert-name
   -t "C,C,C" -d cert-dir -i cert-file
   ```

   上のコマンドの各変数の意味は、次のとおりです。

   - *cert-name* には、このルート証明書の任意の名前を指定します。
   - *cert-dir* には、証明書関連のファイルが置かれているディレクトリを指定します。
   - *cert-file* には、Base64 で符号化されたルート証明書ファイルを指定します。

   certutil ユーティリティの詳細は、「certutil -H」と入力してオンラインヘルプを参照してください。

4. 証明書が正しくインストールされたことを確認するには、コマンド行に次のように入力します。

   ```
   # ./certutil -L -d .
   ```

   インストールしたルート CA 証明書の名前を含む信頼データベース情報が表示されます。次に例を示します。

```
Certificate Name                                  Trust Attrubutes

cert-name                                         C,C,C

p    Valid peer
P    Trusted peer (implies c)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

REMOTE_USER サーバ変数の設定

REMOTE_USER サーバ環境変数は、Sun ONE Identity Server の認証ユーザまたは匿名ユーザに設定できます。この変数を特定のユーザに設定することによって、Web アプリケーション (CGI、サーブレット、ASP プログラムなど) をそのユーザが利用できるようになります。この機能によって、特定のユーザに表示される HTML ページのコンテンツをパーソナライズできます。

AMAgent.properties ファイルで指定されたグローバル不適用 URL (認証されていないユーザがアクセスできる URL) に対して REMOTE_USER 設定を有効にするには、AMAgent.properties ファイルの次のプロパティを TRUE (デフォルトでは、この値は FALSE) に設定する必要があります。

```
com.sun.am.policy.agents.anonRemoteUserEnabled=TRUE
```

このプロパティ値を TRUE に設定すると、REMOTE_USER の値は、AMAgent.properties ファイルの次のプロパティに含まれる値 (デフォルトでは、この値は anonymous) に設定されます 。

```
com.sun.am.policy.agents.unauthenticatedUser=anonymous
```

注	この機能は、Sun ONE Web Proxy Server エージェントには使用できません。

クライアント IP アドレスの検証

この機能を使用して、SSO トークンの盗難や「ハイジャック」を防ぎ、セキュリティを向上させることができます。

`AMAgent.properties` ファイルには `com.sun.am.policy.agents.client_ip_validation_enable` というプロパティが含まれており、デフォルトでは、このプロパティは `false` に設定されています。

このプロパティの値を `true` に設定すると、SSO トークンを含む各着信要求に対して、クライアント IP アドレスの検証が有効になります。要求の生成元の IP アドレスが SSO トークンの発行先の IP アドレスと一致しない場合、要求は拒否されます。これは基本的に、拒否ポリシーの適用と同じです。

ただし、クライアントブラウザが Web プロキシを使っている場合、またはエージェントが保護する Web サーバとクライアントブラウザとの間に負荷均衡アプリケーションがある場合は、この機能を使用しないでください。そのような場合、要求に現れる IP アドレスは、クライアントブラウザが稼動している実際の IP アドレスを反映しません。

POST データの保存

POST データの保存は、Sun ONE Web Server 6.0 SP*x* エージェントと Sun ONE Web Server 4.1 SP8 エージェントの両方でサポートされています。ユーザは、Identity サーバにログインする前に HTML 形式で Web サーバに送信される POST データを保存しておけます。このデータを含む HTML ページは、通常はグローバルリストに含まれ、適用リストには含まれません。デフォルトでは、この機能は無効に設定されています。

この機能は、`AMAgent.properties` ファイルの 2 つのプロパティで設定されます。この機能を有効にするときは、`AMAgent.properties` ファイルの次のプロパティの値を `true` から `false` に変更します。

```
com.sun.am.policy.agents.is_postdatapreserve_enabled = true
```

```
com.sun.am.policy.agents.postcacheentrylifetime = 10
```

2 番目のプロパティは、POST データが Web サーバのキャッシュに存在できる時間を決定します。一定の間隔で reaper スレッドが呼び出され、指定された時間を超過した POST データのキャッシュエントリをクリアします。次のプロパティは、管理者がこの間隔を設定するために使用されます。デフォルトでは、このプロパティは 10 分に設定されています。

注 この機能は、Sun ONE Web Proxy Server エージェントと Apache エージェントでは使用できません。

共有シークレットの暗号化ユーティリティ

ポリシーエージェントは、共有シークレットを `AMAgent.properties` ファイルに保存します。このパスワードのデフォルトは、Indentity Server の内部 LDAP 認証ユーザのパスワードです。これは、サーバ側で `AMConfig.Properties` ファイルを編集することで変更できます。

`AMAgent.properties` ファイルの `com.sun.am.policy.am.password` プロパティには、エージェントのインストール時に暗号化された共有シークレットを設定できます。

共有シークレットをリセットまたは変更するときは、次のユーティリティを使ってプロパティに値を設定します。

1. 次のディレクトリに移動します。

   Agent_Install_Dir`/bin`

2. コマンド行から次のスクリプトを実行します。

   `crypt_util` *shared_secret*

3. 手順 2 の出力をコピーして次のプロパティに貼り付けます。

   com.sun.am.policy.am.password

4. Web サーバを再起動し、エージェントが保護するリソースにアクセスしてみます。

Solaris エージェントのトラブルシューティング

古いインストールを削除したのにエージェントをインストールできない
エージェントのインストーラを実行すると、次のようなメッセージが表示されます。

```
"Sun ONE Identity Server Policy Agent 2.0 for Sun ONE Web Server
6.0 SPx is installed. Please refer to installation manual to
configure this agent for another web server instance. Or
uninstall it before installing another agent."
```

考えられる原因

- エージェントの既存のインストールが残っている
- 以前にエージェントをインストールしたが、そのエージェントのアンインストーラを使わずにエージェントをアンインストールした
- インストーラの productregistry ファイルが破損している

解決法

- エージェントのすべての既存インストールがアンインストールされていることを確認します。
- エージェントの既存インストールが見つからない場合は、productregistry ファイルが破損している可能性があります。このファイルは、インストーラがインストールされた製品を追跡するときに使われます。保存場所は /var/sadm/install ディレクトリです。

注　変更を行う前にこのファイルのバックアップコピーを作成してください。

このファイルからエージェント製品を削除します。この項目は、次の行から始まります。

```
<compid>SUNWamcom
        <compversion>2.0
            <uniquename>SUNWamcom</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent uninstall script
        <compversion>2.0
            <uniquename>Agent uninstall script</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent installer resource bundle
        <compversion>2.0
            <uniquename>Agent installer resource
bundle</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent Common Core and SDK
        <compversion>2.0
            <uniquename>Agent Common Core and SDK</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>SUNWames6
        <compversion>2.0
            <uniquename>SUNWames6</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent for ...
        <compversion>2.0
            <uniquename>Agent for ...</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Sun ONE Identity Server Policy Agent
        <compversion>2.0
            <uniquename>Sun ONE Identity Server Policy
Agent</uniquename>
    </compid>
```

既知の問題

Solaris 2.8 Patch # 109234-09 をインストールすると、Apache サーバが正常に起動しなくなったり、ハングアップしたりする

この問題は、パッチのインストール時に JServ をシステムにインストールしなかったために生じている可能性があります。この問題を解決するには、/etc/apache/ ディレクトリにある httpd.conf ファイルの次の行に変更を加えます。

LoadModule jserv_module /usr/apache/libexec/mod_jserv.so に /etc/apache/jserv.conf を追加します

起動時にエラーメッセージが表示される

エージェントをインストールすると、Apache サーバの起動時に次のエラーメッセージが表示されます。

```
Syntax error on line 1 of
/etc/opt/SUNWam/agents/apache/config/_usr_local_apache_conf/dsam
e.conf:

Invalid command 'LoadModule', perhaps mis-spelled or defined by a
module not included in the server configuration
./apachectl start: httpd could not be started
```

これは、Apache サーバの mod_so が無効で、ダイナミックな共有オブジェクトに対応できていないことを意味します。mod_so を有効にする方法については、http://httpd.apache.org/ に用意されている Apache サーバのマニュアルを参照してください。

Sun ONE Web Proxy Server エージェントでポリシーが機能しない

Sun ONE Web Proxy Server 3.6 の管理コンソールでは、リバースプロキシエージェントのポリシーを設定するリソースの名前は、クライアントからの URL を先頭に付けて設定します。

Sun ONE Web Server または Sun ONE Web Proxy Server の設定をそれぞれの管理コンソールで変更すると、エージェントは機能しなくなる

エージェントのインストール時にサーバの設定ファイルに加えた変更は、管理コンソール側で変更を保存すると上書きされてしまいます。

管理コンソールを使う場合は、最初にディスクからメモリに設定情報をロードして変更を加え、「適用」をクリックしてメモリからディスクに変更を保存します。

第 3 章

Windows 2000 のポリシーエージェント

Sun ONE Identity Server のポリシーエージェントは、Sun ONE Identity Server と連携して企業の Web サーバに対するユーザアクセスの可否を制御します。この章では、Windows 2000 オペレーティングシステムで稼動する各種の Web サーバでポリシーエージェントを設定する方法について説明します。

次のトピックがあります。

- 始める前に
- グラフィカルユーザインタフェースによるインストール
- コマンド行によるインストール
- SSL (Secure Sockets Layer) とエージェントの使用
- REMOTE_USER サーバ変数の設定
- クライアント IP アドレスの検証
- POST データの保存
- 共有シークレットの暗号化ユーティリティ
- IIS ポリシーエージェントのトラブルシューティング

始める前に

第 1 章「ご使用にあたって」で説明されている概念を良く理解しておいてください。この章には、次のトピックスに関する簡単ですが重要な情報があります。

- ポリシーエージェントの動作
- JRE (Java Runtime Environment) 1.3.1 要件
- Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ
- 同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定
- Sun ONE Identity Server エージェントのフェイルオーバ機能の提供
- エージェントキャッシュの更新
- グローバル不適用 URL リスト
- グローバル不適用 IP アドレスリスト
- ポリシーを適用しない認証だけの適用
- HTTP ヘッダーを介した LDAP ユーザ属性の転送
- `AMAgent.properties` ファイル
- 完全指定ドメイン名の設定
- CDSSO の設定

サポートされる Windows の Web サーバ

Sun ONE Identity Server のポリシーエージェントは、Windows 2000 オペレーティングシステム上で稼動する次の Web サーバをサポートします。

- Microsoft IIS 5.0
- Sun ONE Web Server 6.0 SP*x*

グラフィカルユーザインタフェースによるインストール

Microsoft IIS のポリシーエージェントのインストール

IIS エージェントは、Microsoft IIS (Internet Information Services) Web サーバへの URL によるアクセスに対してポリシーを適用します。エージェントは、IIS Web サービスレベルでインストールされる IIS ISAPI フィルタであるため、IIS Web サイトすべてにポリシーが適用されます。技術的な問題から、エージェントを Web サイトレベルでインストールすることはできません。

インストールの前に、エージェントをインストールするシステムのエントリにドメイン名が設定されていることを確認してください。Sun ONE Identity Server 6.0 を実行する Web サーバが別のシステムで稼動している場合は、そのサーバも DNS の照会リストに登録されていることを確認してください。

ポリシーエージェントを Microsoft IIS にインストールするには

インストールプログラムを実行するには、管理者特権が必要です。

1. 製品のバイナリファイルを解凍します。
2. `setup.exe` をダブルクリックして、インストールプログラムを実行します。
3. 開始ウィンドウで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。

5. このエージェントをインストールする Web サーバの情報を入力します。

   Host Name: エージェント Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   IIS Document Root: ドキュメントルートディレクトリを入力します。このディレクトリは、Web サーバルートの `w3svc` からアクセスできる必要があります。

   Server Port: エージェントで保護する Web サーバのポート番号を入力します。

   Server Protocol: Web サーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

   Agent Deployment URI: ディレクトリ名を入力します。デフォルトの URI (Universal Resource Identifier) は `/amagent` です。

注 エージェントは、通知や POST データの保存など、一部の重要機能の実行に `com.sun.am.policy.agents.agenturiprefix` プロパティの値を使います。このプロパティに有効な URL を設定することが重要です。このプロパティのデフォルト値は次のとおりです。

`http://`*host.domain*`:`*port*`/`*agent_deployment_uri*

host、*domain*、*port* はエージェントがインストールされているサーバの完全指定ドメイン名とポート番号です。`agent_deployment_uri` は URI の接頭辞で、エージェントに関連する HTML ページの検索場所を Web サーバに指定します。デフォルト値は `amagent` です。

すべての情報を正しく入力したら、「Next」をクリックします。

6. Sun ONE Identity Server を実行する Web サーバに関する次の情報を入力します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`myserver.siroe.com` などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定ドメイン名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

7. すべての情報を正しく入力したら、「Next」をクリックします。
8. 「Installation Summary」を見直して、入力した情報が正しいことを確認します。変更が必要な場合は、「Back」をクリックします。すべての情報を正しく入力したら、「Next」をクリックします。
9. 「Ready to Install」のページで、「Install Now」をクリックします。
10. インストールが終了したら、「Details」をクリックしてインストールの詳細を確認するか、「Close」をクリックしてインストールプログラムを終了します。
11. インストールを実行すると、エージェントのライブラリの場所がシステムパスに追加されます。変更を有効にし、エージェントを正しく機能させるには、コンピュータを再起動する必要があります。

注 マシンで以前に IIS 5.0 ポリシーエージェントをインストールおよびアンインストールした場合、同じ IIS 5.0 エージェントを同じディレクトリにインストールするときは、再起動する必要はありません。

Sun ONE Web Server のポリシーエージェントのインストール

インストールプログラムを実行するには、管理者特権が必要です。

Sun ONE Web Server にポリシーエージェントをインストールには

1. 製品のバイナリファイルを解凍します。
2. `setup.exe` をダブルクリックして、インストールプログラムを実行します。
3. 開始ウィンドウで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。
5. このエージェントをインストールする Web サーバに関する次の情報を入力します。

   Host Name: エージェント Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   Web Server Instance Directory: Sun ONE Web Server のインスタンスがあるディレクトリの完全パスを入力します。これが、エージェントが保護する Web サーバのインスタンスです。例を示します。

   /web_server_root`/https-mycomputer.siroe.com`

   Web Server Port: エージェントで保護する Web サーバのポート番号を入力します。

   Web Server Protocol: Web サーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

   Agent Deployment URI: ディレクトリ名を入力します。ポリシーエージェントのデフォルトの URI は `/amagent` です。
6. すべての情報を正しく入力したら、「Next」をクリックします。

7. Sun ONE Identity Server 6.0 を実行する Web サーバに関する情報を入力します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`myserver.siroe.com` などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

   Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

   Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定ドメイン名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

   Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

   Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

   CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

   CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

8. すべての情報を正しく入力したら、「Next」をクリックします。

9. 「Installation Summary」を見直して、入力した情報が正しいことを確認します。変更が必要な場合は、「Back」をクリックします。すべての情報を正しく入力したら、「Next」をクリックします。

10. 「Ready to Install」のページで、「Install Now」をクリックします。

11. インストールが終了したら、「Details」をクリックしてインストールの詳細を確認するか、「Close」をクリックしてインストールプログラムを終了します。
12. Sun ONE Web Server を再起動するとインストールが完了します。

ポリシーエージェントのアンインストールと無効化

ポリシーエージェントが必要ではなくなった場合は、アンインストールするか、または無効にすることができます。

ポリシーエージェントのアンインストール

1. Windows の「スタート」メニューから、「設定」>「コントロールパネル」を選択します。
2. 「コントロールパネル」で、「アプリケーションの追加と削除」を開きます。
3. 「アプリケーションの追加と削除」ウィンドウで「Sun ONE Identity Server Policy Agent」を選択します。
4. 「追加と削除」をクリックします。
5. 開始パネルで、「Next」をクリックします。
6. 「Uninstall Now」をクリックします。
7. アンインストールが終了したら、「Exit」をクリックします。

Microsoft IIS にインストールされたポリシーエージェントの無効化

次の手順を使用して、Microsoft IIS にインストールされているエージェントを無効にします。

1. インターネットサービスマネージャを起動します。
   - 「スタート」メニューで、「プログラム」>「管理ツール」>「インターネットサービスマネージャ」を選択します。
2. フィルタの状態を確認します。
   a. 「インターネット インフォメーション サービス」ウィンドウのツリーペインで、ホストコンピュータのプロパティを開きます。
   b. ツリー内で、「インターネット インフォメーション サービス」ルートの下にホストコンピュータ名が表示されます。
   c. 「インターネット インフォメーション サービス」タブの「マスタ プロパティ」セクションで、「編集」をクリックします。

d. 表示された「WWW サービス マスタ プロパティ」ダイアログで、「ISAPI フィルタ」タブを選択します。

e. 「Sun ONE Identity Server Agent」というフィルタを強調表示します。

「編集」をクリックして、フィルタ名と実行可能パスを表示できます。この情報は、エージェントを有効にし直すときに必要になります。「キャンセル」をクリックしてプログラムに戻ります。

f. 「削除」をクリックします。

g. 「適用」をクリックし、「WWW サービス マスタ プロパティ」ダイアログを閉じます。

h. Microsoft IIS を再起動します。

コマンド行によるインストール

グラフィカルユーザインタフェース (GUI) バージョンの代わりに、コマンド行バージョンのインストールプログラムを使用することもできます。

コマンド行によるエージェントのインストール

1. バイナリファイルを解凍したディレクトリに移動して、コマンド行に次のコマンドを入力します。

```
# setup.bat -nodisplay
```

2. プロンプトが表示されたら、次の情報を入力します。

Have you read, and do you accept, all of the terms of the preceding Software License Agreement?

Install Sun ONE Identity Server Policy Agent in this directory: エージェントをインストールするディレクトリを指定します。かっこ内に表示されているデフォルトのディレクトリを受け入れるには、Enter を押します。受け入れない場合は、完全パスを入力します。

3. プロンプトに従って、このエージェントが保護する Web サーバインスタンスに関する次の情報を入力します。

- `Host Name`
- `IIS Document Root`
- `Server Port`
- `Server Protocol`

- Agent Deployment URI

これらの各項目の詳細は「グラフィカルユーザインタフェースによるインストール」を参照してください。

4. プロンプトに従って、Sun ONE Identity Server サービスを実行する Web サーバに関する次の情報を入力します。
   - Primary Server Host
   - Primary Server Port
   - Primary Server Protocol
   - Primary Server Deployment URI
   - Primary Console Deployment URI
   - Failover Server Host
   - Failover Server Port
   - Failover Server Deployment URI
   - Secondary Console Deployment URI
   - Agent-Identity Server Shared Secret
   - Re-enter Shared secret
   - CDSSO feature enabled
   - CDSSO component URL

   これらの各項目の詳細は「グラフィカルユーザインタフェースによるインストール」を参照してください。
5. 表示されたら、指定したインストール情報の概要を確認します。Enter を押して操作を続行するか、あるいは感嘆符 (!) を入力してプログラムを終了します。
6. 次のメッセージが表示されます。

```
Ready to Install

1. Install Now
2. Start Over
3. Exit Installation

What would you like to do
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

7. 次のメッセージが表示されます。

```
Product                              Result     More Information
1. Sun ONE Identity Server Agent     Installed  Available
2. Done
```

ログ情報を表示するときは、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

コマンド行によるエージェントのアンインストール

1. *Agent_Install_Dir* ディレクトリで、コマンド行に次のコマンドを入力します。

   ```
   java uninstall_Sun_ONE_Identity_Server_Policy_Agent -nodisplay
   ```

2. 次のメッセージが表示されます。

```
1. Uninstall Now
2. Start Over
3. Exit Uninstallation
What would you like to do?
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

3. 次のメッセージが表示されます。

```
Product                              Result  More Information
1. Sun ONE Identity Server Agent     Full    Available
2. Done
```

ログ情報を表示するときは、1 を入力します。アンインストールプログラムを終了するときは、2 を入力します。

4. アンインストールを終了したら、システムを再起動する必要があります。

   アンインストールの詳細を参照する場合は、次の場所にログファイルが書き出されています。

   ```
   %TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall*
   ```

SSL (Secure Sockets Layer) とエージェントの使用

インストール時に HTTPS プロトコルを選択すると、SSL を介して通信するようにエージェントが自動的に設定されます。

注 次の手順に進む前に、Web サーバに SSL が設定されていることを確認してください。

HTTPS プロトコルを介した通信を使用可能にするのに必要な SSL の概念およびセキュリティ証明書をよく理解しておく必要があります。Web サーバに付属のマニュアルを参照してください。Sun ONE Web Server を使っている場合は、インターネット上の次のマニュアルにアクセスできます。

`http://docs.sun.com/source/816-5682-10/esecurty.htm#1011961`

エージェントのデフォルトの信頼動作

デフォルトでは、リモートの Sun ONE Web Server 6.0 または Microsoft IIS 5.0 にインストールされたポリシーエージェントは、Sun ONE Identity Server を実行する Web サーバが SSL 上で提示したサーバ認証書を信頼します。エージェントはルートの認証局 (CA) 証明書をチェックしません。Sun ONE Identity Server を実行する Web サーバで SSL が有効になっていて、ポリシーエージェントに証明書をチェックさせたい場合は、次の 2 つを行う必要があります。

1. エージェントのデフォルトの信頼動作を無効にします。
2. エージェントがインストールされているリモート Web サーバにルート CA 証明書をインストールします。ルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

エージェントのデフォルト信頼動作の無効化

次のプロパティは AMAgent.properites ファイルにあり、デフォルトでは true に設定されています。

```
com.sun.am.policy.agents.trustServerCerts=true
```

これは、エージェントが証明書のチェックを行わないということです。

デフォルトの動作を無効にするには

次のプロパティを false に設定する必要があります。

```
com.sun.am.policy.agents.trustServerCerts=false
```

リモート Web サーバへのルート CA 証明書のインストール

リモート Web サーバにインストールするルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

ルート CA 証明書を Sun ONE Web Server にインストールするには

Web サーバに付属のマニュアルにあるルート CA 証明書をインストールするための手順を参照してください。一般に、これは Web サーバの管理コンソールから実行します。Sun ONE Web Server 6.0 のマニュアルは、インターネットの次の URL からアクセスできます。
http://docs.sun.com/source/816-5682-10/esecurty.htm#1011961

ルート CA 証明書を Microsoft IIS にインストールするには

1. 次のディレクトリに移動します。

   Agent_Install_Dir¥iis¥cert

2. Sun ONE Identity Server を実行する Web サーバにインストールされているのと同じルート証明書を、既存の証明書データベースに追加します。コマンド行に次のコマンドを入力します。

   ¥*Agent_Install_Dir*¥bin¥certutil -A -n *cert-name* -t "C,C,C" -d *cert-dir* -i *cert-file*

   使用する変数は次のとおりです。

   - *cert-name* には、このルート証明書の任意の名前を指定します。
   - *cert-dir* には、証明書関連のファイルが置かれたディレクトリを指定します。Windows では、その場所は次のとおりです。

   Agent_Install_Dir¥bin

   - *cert-file* には、Base64 で符号化されたルート証明書ファイルを指定します。
   - certutil の詳細は、「certutil -H」と入力してヘルプを参照してください。

   ルート証明書が正しく証明書データベースにインストールされたことを確認するときは、次のコマンドを実行します。

   Agent_Install_Dir¥bin¥certutil -L -d .

   ルート証明書が追加されている場合は、コマンド出力のリストに名前が表示されます。

3. IIS を再起動します。

REMOTE_USER サーバ変数の設定

REMOTE_USER サーバ環境変数は、Sun ONE Identity Server の認証ユーザまたは匿名ユーザに設定できます。この変数を特定のユーザに設定することによって、Web アプリケーション (CGI、サーブレット、ASP プログラムなど) をそのユーザが利用できるようになります。この機能によって、特定のユーザに表示される HTML ページのコンテンツをパーソナライズできます。

IIS 5.0 エージェントに対して REMOTE_USER 機能を有効にするには、次の手順を実行します。

1. Windows の「スタート」メニューで、「プログラム」>「管理ツール」>「インターネットサービスマネージャ」を選択します。

   これで、インターネットインフォメーションサービスコンソール が起動します。

2. Sun ONE Identity Server エージェントで保護する Web サイトで、「プロパティ」を選択します。
3. 「ディレクトリセキュリティ」タブを選択します。
4. 「匿名アクセスおよび認証コントロール」セクションで、「編集」をクリックします。
5. 表示されたダイアログで、「匿名アクセス」および「基本認証」を選択してから、「統合 Windows 認証」の選択を解除します。

これらの手順を実行すると、許可された URL に合わせて REMOTE_USER が設定されます。

AMAgent.properties ファイルで指定されたグローバル不適用 URL (認証されていないユーザがアクセスできる URL) に対して REMOTE_USER 設定を有効にするには、AMAgent.properties ファイルの次のプロパティを TRUE (デフォルトでは、FALSE) に設定する必要があります。

```
com.sun.am.policy.agents.anonRemoteUserEnabled=TRUE
```

このプロパティ値を TRUE に設定すると、REMOTE_USER の値は、AMAgent.properties ファイルの次のプロパティに含まれる値 (デフォルトでは、anonymous) に設定されます。

```
com.sun.am.policy.agents.unauthenticatedUser=anonymous
```

クライアント IP アドレスの検証

この機能を使用して、SSO トークンの盗難や「ハイジャック」を防ぎ、セキュリティを向上させることができます。

`AMAgent.properties` ファイルには `com.sun.am.policy.agents.client_ip_validation_enable` というプロパティが含まれており、デフォルトでは、このプロパティは `false` に設定されています。

このプロパティの値を `true` に設定すると、SSO トークンを含む各着信要求に対して、クライアント IP アドレスの検証が有効になります。要求の生成元の IP アドレスが SSO トークンの発行先の IP アドレスと一致しない場合、要求は拒否されます。これは基本的に、拒否ポリシーの適用と同じです。

ただし、クライアントブラウザが Web プロキシを使っている場合、またはクライアントブラウザとエージェントが保護する Web サーバとの間に負荷均衡アプリケーションがある場合は、この機能を使用しないでください。そのような場合、要求に現れる IP アドレスは、クライアントブラウザが稼動している実際の IP アドレスを反映しません。

POST データの保存

Sun ONE Web Server 6.0 SP*x* エージェントは、POST データの保存をサポートしています。ユーザは、ユーザが Identity サーバにログインする前に HTML 形式で Web サーバに送信される POST データを保存しておけます。このデータを含む HTML ページは、通常はグローバルリストに含まれ、適用リストには含まれません。デフォルトでは、この機能は無効に設定されています。

この機能は、`AMAgent.properties` ファイルの 2 つのプロパティで設定されます。この機能を有効にするには、`AMAgent.properties` ファイルの次のプロパティの値を `true` から `false` に変更します。

```
com.sun.am.policy.agents.is_postdatapreserve_enabled=true
```

2 番目のプロパティは、POST データが Web サーバのキャッシュに存在できる時間を決定します。一定の間隔で reaper スレッドが呼び出され、指定された時間を超過した POST データのキャッシュエントリをクリアします。次のプロパティを使用して、管理者はこの間隔を設定できます。デフォルトでは、このプロパティは 10 分に設定されています。

```
com.sun.am.policy.agents.postcacheentrylifetime=10
```

注　この機能は、Windows 2000 の IIS 5.0 エージェントでは使用できません。

共有シークレットの暗号化ユーティリティ

ポリシーエージェントは、共有シークレットを AMAgent.properties ファイルに保存します。このパスワードのデフォルトは、Indentity Server の内部 LDAP 認証ユーザのパスワードです。これは、サーバ側で AMConfig.Properties ファイルを編集することで変更できます。

AMAgent.properties ファイルの com.sun.am.policy.am.password プロパティには、エージェントのインストール時に暗号化された共有シークレットを設定できます。

共有シークレットをリセットまたは変更するには、次のユーティリティを使ってプロパティに値を設定します。

1. 次のディレクトリに移動します。

   Agent_Install_Dir¥bin

2. コマンド行から次のスクリプトを実行します。

   cryptit *shared_secret*

3. 手順 2 の出力をコピーして次のプロパティに貼り付けます。

   com.sun.am.policy.am.password

4. Web サーバを再起動し、エージェントが保護するリソースにアクセスしてみます。

IIS ポリシーエージェントのトラブルシューティング

インストール時に問題が発生した場合は、次の手順を実行します。

- インストールのログファイルでエラーを確認します。

  %TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall.*nnnn*

- アンインストールを実行し、もう一度インストールを実行します。
- IIS にエージェントが読み込まれているかを確認します。

  a. インターネットサービスマネージャを起動します。

  b. 「スタート」メニューで、「プログラム」>「管理ツール」>「インターネットサービスマネージャ」を選択します。

  c. 「インターネット インフォメーション サービス」ウィンドウのツリーペインで、ホストコンピュータのプロパティを開きます。

d. ツリー内で、「インターネット インフォメーション サービス」ルートの下にホストコンピュータ名が表示されます。

e. 「インターネット インフォメーション サービス」タブの「マスタ プロパティ」セクションで、「編集」をクリックします。

f. 表示された「WWW サービス マスタ プロパティ」ダイアログで、「ISAPI フィルタ」タブを選択します。

g. 「Sun ONE Identity Server Agent」というフィルタを探します。

「Sun ONE Identity Server Agent」フィルタが見つからない場合は、インストールプログラムが実行され、インストール中にエラーが発生しなかったかを確認します。インストールのログは、次の場所に記録されています。

`%TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall.`*nnnn*

「Sun ONE Identity Server Agent」という名前の右にある「状態」列に緑色の上向き矢印が表示されている場合は、エージェントが IIS に正しく読み込まれています。赤い下向き矢印が表示している場合は、フィルタが正しく読み込まれていません。フィルタが読み込まれない (赤い矢印) 原因として可能性が高いのは、必要な dll ファイルが見つからない場合です。

h. システムパスをチェックして、次のディレクトリがあることを確認します。

Agent_Install_Dir`¥bin`

i. フィルタが正しく読み込まれていない場合は、次のことを確認します。

- 「Sun ONE Identity Server Agent」をクリックし、「編集」をクリックしてエージェントの DLL ファイルのパスを確認します。「実行ファイル」テキストボックスに示されたパスが有効であることを確認します。
- エージェントは、ほかにもいくつかの DLL ファイルを使用します。次のファイルが `Agents¥bin` ディレクトリにあることを確認します。

```
amsdk.dll

ames6.dll

libnspr4.dll

libplc4.dll

libplds4.dll

libxml2.dll

nss3.dll

ssl3.dll
```

j. このライブラリがシステムパスに含まれている場合は、システムを再起動してみます。

- フィルタの読み込みエラーのログは、システムイベントログに記録されます。イベントログを確認するには、次の手順を実行します。

  a. 「スタート」メニューから、「プログラム」>「管理ツール」>「サービス」を選択します。

  b. 「System Log」を選択します。

  c. 「ソース」が「W3SVC」であるエラーメッセージを確認します。

- エージェントは読み込まれているが、IIS Web サーバに対するすべての URL リクエストで「HTTP 500 Internal Server Error」が返される場合。

  この場合は、エージェントは読み込まれていますが、正しく初期化されていません。すべての HTTP リクエストで「HTTP 500 Internal Server Error」が返されるのは、エージェントが初期化できなかったときに URL リソースを保護するためのフェイルセーフ機能が働くためです。この原因として可能性が高いのは、Sun ONE Identity Server エージェントまたはサーバの設定に問題があるか、これらが使用不能である場合です。

- エージェントのデバッグログを確認します。

  このログは、デフォルトで *Agent_Install_Dir* ディレクトリに置かれます。初期設定やエージェントの動作に関する問題を解決するときは、デバッグ情報が最も役立ちます。ログファイルのディレクトリは、次のプロパティによって指定されます。

  次のディレクトリ内の `AMAgent.properties` ファイルに含まれる `com.sun.am.policy.am.logFile`

  Agent_Install_Dir`¥iis¥config¥`*_PathInstanceName*

  `com.sun.am.policy.am.loglevels` プロパティは、ログ情報の記録量を制御します。ログカテゴリごとのログレベルを設定します。

  この値の書式は次のとおりです。

  ModuleName `[:`*Level*`]` `,` *ModuleName* `[:`*Level*`]]*`*

  現在使用中のモジュール名は、AuthService、NamingService、PolicyService、SessionService、PolicyEngine、ServiceEngine、Notification、PolicyAgent、RemoteLog、all です。ログレベルの指定を省略すると、デフォルトのログレベルでログモジュールが作成されます。これは、「all」モジュールに設定されているログレベルです。

  all モジュールを使うことで、全モジュールのログレベルを設定することができます。また、後で作成されるすべてのモジュールにもこのデフォルトのログレベルが設定されます。次に、「Level」の値について説明します。

  `0` = 特定モジュールのログ記録を無効にする

  `1` = エラーメッセージを記録する

  `2` = 警告とエラーのメッセージを記録する

3 = 情報、警告、エラーのメッセージを記録する

4 = デバッグ、情報、警告、エラーのメッセージを記録する

5 = レベル 4 と同様だが、より多くのデバッグメッセージを記録する

- エージェントが AMAgent.properties 設定ファイルを探すことができることを確認します。

  エージェントはレジストリキー HKEY_LOCAL_MACHINE¥Software¥Sun Microsystems¥Identity Server IIS Agent を使って、AMAgent.properties ファイルの場所を特定します。AMAgent.properties ファイルは、次のディレクトリにあります。

  Agent_Install_Dir¥iis¥config¥*_PathInstanceName*

- エージェントは、AMAgent.properties に指定されたデバッグログファイルが開始される前に発生したエラーのログを、アプリケーションイベントログを使って記録します。

  a. 「スタート」メニューから、「プログラム」>「管理ツール」>「サービス」を選択します。

  b. 「Application Log」を選択します。

  c. 「ソース」が「Sun ONE Identity Server IIS Agent」であるエラーメッセージを確認します。

古いインストールを削除したのにエージェントをインストールできない

エージェントのインストーラを実行すると、次のようなメッセージが表示されます。

```
"Sun ONE Identity Server Policy Agent 2.0 for Sun ONE Web Server
6.0 SPx is installed. Please refer to installation manual to
configure this agent for another web server instance. Or
uninstall it before installing another agent."
```

考えられる原因

- エージェントの既存のインストールが残っている
- 以前にエージェントをインストールしたが、そのエージェントのアンインストーラを使わずにエージェントをアンインストールした
- インストーラの productregistry ファイルが破損している

解決法

- エージェントのすべての既存インストールがアンインストールされていることを確認します。
- エージェントの既存インストールが見つからない場合は、productregistry ファイルが破損している可能性があります。このファイルは、インストーラがインストール製品を追跡するときに使われます。このファイルは、C:¥WINNT¥system32 ディレクトリに保存されています。

注　変更を行う前にこのファイルのバックアップコピーを作成してください。

このファイルからエージェント製品を削除します。この項目は、次の行から始まります。

```
<compid>SUNWamcom
        <compversion>2.0
            <uniquename>SUNWamcom</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent uninstall script
        <compversion>2.0
            <uniquename>Agent uninstall script</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent installer resource bundle
        <compversion>2.0
            <uniquename>Agent installer resource
bundle</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent Common Core and SDK
        <compversion>2.0
            <uniquename>Agent Common Core and SDK</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>SUNWames6
        <compversion>2.0
<uniquename>SUNWames6</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent for ...
        <compversion>2.0
            <uniquename>Agent for ...</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Sun ONE Identity Server Policy Agent
```

```
<compid>SUNWamcom
        <compversion>2.0
            <uniquename>Sun ONE Identity Server Policy
Agent</uniquename>
    </compid>
```

Windows の「スタート」>「設定」>「コントロールパネル」>「アプリケーションの追加と削除」を使ってエージェントをアンインストールできない

考えられる原因 : Java のクラスパスがマシンに正しく設定されていません。

解決法 : 次の手順を実行してエージェントをアンインストールします。

1. コマンドプロンプトウィンドウを開きます。
2. *Agent_Install_Dir* に移動します。
3. 次のコマンドを実行します。

   ```
   java uninstall_Sun_ONE_Identity_Server_Policy_Agent
   ```

既知の問題

IIS 5.0 と終了の問題

IIS 5.0 にポリシーエージェントをインストールすると、各 Web サイトを終了したときに、メモリの衝突に関するメッセージが表示されることがあります。このメッセージは無視しても問題ないので、IIS サーバを再起動してください。

管理コンソールを使って Sun ONE Web Server の設定に変更を加えると、エージェントが機能しなくなる

エージェントのインストール時にサーバの設定ファイルに加えた変更は、管理コンソール側で変更を保存するときに上書きされてしまいます。

管理コンソールを使う場合は、最初にディスクからメモリに設定情報をロードして変更を加え、「適用」をクリックしてメモリからディスクに変更を保存します。

第 4 章

Windows NT のポリシーエージェント

Sun ONE Identity Server のポリシーエージェントは、Sun ONE Identity Server と連携して企業の Web サーバに対するユーザアクセスの可否を制御します。この章では、Windows NT オペレーティングシステム、バージョン 4.0、サービスパック 6 で稼動する IIS 4.0 で URL アクセスエージェントを設定する方法を説明します。

次のトピックがあります。

- 始める前に
- グラフィカルユーザインタフェースによるインストール
- コマンド行によるインストール
- SSL (Secure Sockets Layer) とエージェントの使用
- REMOTE_USER サーバ変数の設定
- クライアント IP アドレスの検証
- 共有シークレットの暗号化ユーティリティ
- IIS 4.0 ポリシーエージェントのトラブルシューティング
- 既知の問題

始める前に

第 1 章「ご使用にあたって」で説明されている概念を良く理解しておいてください。この章には、次のトピックスに関する簡単ですが重要な情報があります。

- ポリシーエージェントの動作
- JRE (Java Runtime Environment) 1.3.1 要件
- Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ
- 同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定
- Sun ONE Identity Server エージェントのフェイルオーバ機能の提供
- エージェントキャッシュの更新
- グローバル不適用 URL リスト
- グローバル不適用 IP アドレスリスト
- ポリシーを適用しない認証だけの適用
- HTTP ヘッダーを介した LDAP ユーザ属性の転送
- `AMAgent.properties` ファイル
- 完全指定ドメイン名の設定
- CDSSO の設定

サポートされる Windows NT の Web サーバ

Sun ONE Policy Agent は、Windows NT Server 4.0 SP6 オペレーティングシステム上で次の Web サーバをサポートします。

- Microsoft IIS 4.0

グラフィカルユーザインタフェースによるインストール

Microsoft IIS 4.0 のポリシーエージェントのインストール

IIS エージェントは、Microsoft の Internet Information Services (IIS4.0) Web サーバへの URL によるアクセスに対してポリシーを適用します。エージェントは、IIS Web サービスレベルでインストールされる IIS ISAPI フィルタであるため、IIS Web サイトすべてにポリシーが適用されます。技術的な問題から、エージェントを Web サイトレベルでインストールすることはできません。

インストールの前に、エージェントをインストールするシステムのエントリにドメイン名が設定されていることを確認してください。Sun ONE Identity Server を実行する Web サーバが別のシステムで稼動している場合は、そのサーバも DNS の照会リストに登録されていることを確認してください。

ポリシーエージェントを Microsoft IIS 4.0 に インストールするには

インストールプログラムを実行するには、管理者特権が必要です。

1. 製品のバイナリファイルを解凍します。
2. `setup.exe` をダブルクリックして、インストールプログラムを実行します。
3. 開始ウィンドウで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。

5. このエージェントをインストールする Web サーバの情報を入力します。

   Host Name: エージェント Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   IIS Document Root: ドキュメントルートディレクトリを入力します。このディレクトリは、Web サーバルートの `w3svc` からアクセスできる必要があります。

   Server Port: エージェントで保護する Web サーバのポート番号を入力します。

   Server Protocol: Web サーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

   Agent Deployment URI: ディレクトリ名を入力します。デフォルトの URI (Universal Resource Identifier) は `/amagent` です。

注 エージェントは、通知や POST データの保存など、一部の重要機能の実行に `com.sun.am.policy.agents.agenturiprefix` プロパティの値を使います。このプロパティに有効な URL を設定することが重要です。このプロパティのデフォルト値は次のとおりです。

`http://`*host.domain*`:`*port*`/`*agent_deployment_uri*

host、*domain*、*port* はエージェントがインストールされているサーバの完全指定ドメイン名とポート番号です。*agent_deployment_uri* は URI の接頭辞で、エージェントに関連する HTML ページの検索場所を Web サーバに指定します。デフォルト値は `amagent` です。

   すべての情報を正しく入力したら、「Next」をクリックします。

6. Sun ONE Identity Server を実行する Web サーバに関する次の情報を入力します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`myserver.siroe.com` などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

7. すべての情報を正しく入力したら、「Next」をクリックします。
8. 「Summary」パネルで選択項目を確認し、「Install Now」をクリックします。
9. インストールが終了したら、詳細を確認して、「Exit」をクリックします。
10. インストールを実行すると、エージェントのライブラリの場所がシステムパスに追加されます。変更を有効にし、エージェントを正しく機能させるには、コンピュータを再起動する必要があります。

注 マシンで以前に IIS 4.0 エージェントをインストールおよびアンインストールした場合、同じ IIS 4.0 エージェントを同じディレクトリにインストールするときは、再起動する必要はありません。

ポリシーエージェントのアンインストールと無効化

ポリシーエージェントが必要ではなくなった場合は、アンインストールするか、または無効にすることができます。

ポリシーエージェントのアンインストール

1. 「スタート」メニューから、「設定」>「コントロールパネル」を選択します。
2. 「コントロールパネル」で、「アプリケーションの追加と削除」を開きます。
3. 「アプリケーションの追加と削除」ウィンドウで「Sun ONE Identity Server Policy Agent」を選択します。
4. 「追加と削除」をクリックします。
5. 開始パネルで、「Next」をクリックします。
6. 「Uninstall Now」をクリックします。
7. アンインストールが終了したら、「Close」をクリックします。

Microsoft IIS 4.0 にインストールされたポリシーエージェントの無効化

1. インターネットサービスマネージャを起動します。
   a. Windows の「スタート」メニューで、「プログラム」>「Windows NT 4.0 Option Pack」>「Microsoft Internet Information Server」>「インターネットサービスマネージャ」を選択 します。
2. フィルタの状態を確認します。
   a. ウィンドウの左ペインで、コンピュータのホスト名を右クリックして、「プロパティ」を選択します。
   b. 「マスタ プロパティ」セクションで、「WWW サービス」を選択して、「編集」をクリックします。
   c. 表示された「WWW サービス マスタ プロパティ」ダイアログで、「ISAPI フィルタ」タブを選択します。
   d. 「Sun ONE Identity Server Policy Agent」というフィルタを強調表示します。

      「編集」をクリックして、フィルタ名と実行可能パスを表示します。この情報は、エージェントを有効にし直すときに必要になります。「キャンセル」をクリックしてプログラムに戻ります。
   e. 「削除」をクリックします。
   f. 「適用」をクリックし、「WWW サービス マスタ プロパティ」ダイアログを閉じます。

3. iisadmin を停止してから、iisadmin と w3csvc を起動して、Microsoft IIS 4.0 を再起動します。

コマンド行によるインストール

グラフィカルユーザインタフェース (GUI) バージョンの代わりに、コマンド行バージョンのインストールプログラムを使用することもできます。

コマンド行を使ってエージェントをインストールするには

1. バイナリファイルを解凍したディレクトリに移動して、コマンド行に次のコマンドを入力します。

   ```
   setup.bat -nodisplay
   ```

2. プロンプトが表示されたら、次の情報を入力します。

   Have you read, and do you accept, all of the terms of the preceding Software License Agreement?

   Install Sun One Policy Server Agent in this directory: エージェントをインストールするディレクトリを指定します。かっこ内に表示されているデフォルトのディレクトリを受け入れるには、Enter を押します。受け入れない場合は、完全パスを入力します。

3. プロンプトに従って、このエージェントが保護する Web サーバインスタンスに関する次の情報を入力します。

   - Host Name
   - Web Server Port
   - Web Server Protocol
   - Web Server Doocument Root
   - Agent Deployment URI

   これらの項目については、「グラフィカルユーザインタフェースによるインストール」を参照してください。

4. プロンプトに従って、Sun ONE Identity Server サービスを実行する Web サーバに関する次の情報を入力します。

   - Primary Server Host
   - Primary Server Port

- Primary Server Protocol
- Primary Server Deployment URI
- Primary Console Deployment URI
- Failover Server Host
- Failover Server Port
- Failover Server Deployment URI
- Failover Console Deployment URI
- Agent-Identity Server Shared secret
- Re-enter Shared secret
- CDSSO feature enabled
- CDSSO Component URL

これらの項目については、「グラフィカルユーザインタフェースによるインストール」を参照してください。

5. 表示されたら、指定したインストール情報の概要を確認します。Enter を押して操作を続行するか、あるいは感嘆符 (!) を入力してプログラムを終了します。
6. 次のメッセージが表示されます。

```
Ready to Install

1. Install Now
2. Start Over
3. Exit Installation

What would you like to do
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

7. 次のメッセージが表示されます。

```
Product                              Result     More Information
1. Sun ONE Identity Server Agent     Installed  Available
2. Done
```

ログ情報を表示するときは、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

8. インストールが終了したら、IIS 4.0 を再起動します。

コマンド行を使ってエージェントをアンインストールするには

1. *Agent_Install_Dir* ディレクトリで、コマンド行に次のコマンドを入力します。

   ```
   java uninstall_Sun_ONE_Identity_Server_Policy_Agent -nodisplay
   ```

2. プロンプトが表示されたら、次の情報を入力します。

   アンインストーラは、setup プログラムを使ってインストールされているエージェントを検出します。エージェントをアンインストールするときは、1 を入力します。

3. 次のメッセージが表示されます。

```
1. Uninstall Now
2. Start Over
3. Exit Uninstallation
What would you like to do?
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してアンインストールを開始します。

4. 次のメッセージが表示されます。

```
Product                          Result    More Information
1. Sun ONE Identity Server Agent Full      Available
2. Done
```

ログ情報を表示するときは、1 を入力します。アンインストールプログラムを終了するときは、2 を入力します。

5. アンインストールが完了したら、システムを再起動します。

アンインストールの詳細を参照する場合は、次の場所にログファイルが書き出されています。

```
%TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall.*
```

SSL (Secure Sockets Layer) とエージェントの使用

インストール時に HTTPS プロトコルを選択すると、SSL を介して通信するためにエージェントが自動的に設定されます。

注 次の手順に進む前に、Web サーバに SSL が設定されていることを確認してください。

エージェントのデフォルトの信頼動作

デフォルトでは、リモートの Microsoft IIS 4.0 にインストールされたポリシーエージェントは、Sun ONE Identity Server サービスを実行する Web サーバが SSL 上で提示したサーバ認証書を信頼します。エージェントはルートの認証局 (CA) 証明書をチェックしません。Sun ONE Identity Server を実行する Web サーバで SSL が有効になっていて、ポリシーエージェントで証明書をチェックしたい場合は、次の 2 つを行う必要があります。

1. エージェントのデフォルトの信頼動作を無効にします。
2. エージェントがインストールされているリモート Web サーバにルート CA 証明書をインストールします。ルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

エージェントのデフォルト信頼動作の無効化

次のプロパティは AMAgent.properites ファイルにあり、デフォルトでは true に設定されています。

```
com.sun.am.policy.agents.trustServerCerts=true
```

これは、エージェントが証明書のチェックを行わないということです。

デフォルトの動作を無効にするには

次のプロパティを false に設定する必要があります。

```
com.sun.am.policy.agents.trustServerCerts=false
```

リモート Web サーバへのルート CA 証明書のインストール

リモート Web サーバにインストールするルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

ルート CA 証明書を Microsoft IIS にインストールするには

1. 次のディレクトリに移動します。

   Agent_Install_Dir¥iis¥cert

2. Sun ONE Identity Server を実行する Web サーバにインストールされているのと同じルート証明書を、既存の証明書データベースに追加します。コマンド行に次のコマンドを入力します。

   ¥*Agent_Install_Dir*¥bin¥certutil -A -n *cert-name* -t "C,C,C" -d *cert-dir* -i *cert-file*

   使用する変数は次のとおりです。

   - *cert-name* には、このルート証明書の任意の名前を指定します。
   - *cert-dir* には、証明書関連のファイルが置かれたディレクトリを指定します。Windows NT では、その場所は次のとおりです。

     Agent_Install_Dir¥bin

   - *cert-file* には、Base64 で符号化されたルート証明書ファイルを指定します。

- certutil の詳細は、「certutil -H」と入力してヘルプを参照してください。

ルート証明書が正しく証明書データベースにインストールされたことを確認するときは、次のコマンドを実行します。

Agent_Install_Dir¥bin¥certutil -L -d .

ルート証明書が追加されている場合は、コマンド出力のリストに名前が表示されます。

3. IIS を再起動します。

REMOTE_USER サーバ変数の設定

REMOTE_USER サーバ環境変数は、Sun ONE Identity Server の認証ユーザまたは匿名ユーザに設定できます。この変数を特定のユーザに設定することによって、Web アプリケーション (CGI、サーブレット、ASP プログラムなど) をそのユーザが利用できるようになります。この機能によって、特定のユーザに表示される HTML ページのコンテンツをパーソナライズできます。

REMOTE_USER 機能を有効にするには、次の手順を実行します。

1. Windows の「スタート」メニューで、「プログラム」>「Windows NT Option Pack」>「Microsoft Internet Information Server」>「インターネットサービスマネージャ」を選択します。

   これで、Microsoft Management Console が起動します。

2. Sun ONE Identity Server エージェントで保護する Web サイトで、「プロパティ」を選択します。
3. 「ディレクトリセキュリティ」タブを選択します。
4. 「匿名アクセスおよび認証コントロール」セクションで、「編集」をクリックし、「Allow Anonymous Access」(デフォルトで選択されている)、「基本認証」(デフォルトでは選択されていない) を選択して、「チャレンジ / レスポンス」(デフォルトで選択されている) の選択を解除します。

REMOTE_USER は、許可された URL へのアクセス中に設定されます。

AMAgent.properties ファイルで指定されたグローバル不適用 URL (認証されていないユーザがアクセスできる URL) に対して REMOTE_USER 設定を有効にするには、AMAgent.properties ファイルの次のプロパティを TRUE (デフォルトでは、FALSE) に設定する必要があります。

```
com.sun.am.policy.agents.anonRemoteUserEnabled=TRUE
```

このプロパティ値を TRUE に設定すると、REMOTE_USER の値は、AMAgent.properties ファイルの次のプロパティに含まれる値 (デフォルトでは、anonymous) に設定されます。

```
com.sun.am.policy.agents.unauthenticatedUser=anonymous
```

クライアント IP アドレスの検証

この機能を使用して、SSO トークンの盗難や「ハイジャック」を防ぎ、セキュリティを向上させることができます。

AMAgent.properties ファイルには com.sun.am.policy.agents.client_ip_validation_enable というプロパティが含まれており、デフォルトでは、このプロパティは false に設定されています。

このプロパティの値を true に設定すると、SSO トークンを含む各着信要求に対して、クライアント IP アドレスの検証が有効になります。要求の生成元の IP アドレスが SSO トークンの発行先の IP アドレスと一致しない場合、要求は拒否されます。これは基本的に、拒否ポリシーの適用と同じです。

ただし、クライアントブラウザが Web プロキシを使っている場合、またはクライアントブラウザとエージェントが保護する Web サーバとの間に負荷均衡アプリケーションがある場合は、この機能を使用しないでください。そのような場合、要求に現れる IP アドレスは、クライアントブラウザが稼動している実際の IP アドレスを反映しません。

共有シークレットの暗号化ユーティリティ

Sun ONE Identity Server のポリシーエージェントは、共有シークレットを AMAgent.properties ファイルに保存します。このパスワードのデフォルトは、Indentity Server の内部 LDAP 認証ユーザのパスワードです。これは、サーバ側で AMConfig.Properties ファイルを編集することで変更できます。

AMConfig.Properties ファイルの com.sun.am.policy.am.password プロパティには、エージェントのインストール時に暗号化された共有シークレットを設定できます。

共有シークレットをリセットまたは変更するときは、次のユーティリティを使ってプロパティに値を設定します。

1. 次のディレクトリに移動します。

   Agent_Install_Dir¥bin

2. コマンド行から次のスクリプトを実行します。

   cryptit *shared_secret*

3. 手順 2 の出力をコピーして次のプロパティに貼り付けます。

   com.sun.am.policy.am.password

4. Web サーバを再起動し、エージェントが保護するリソースにアクセスしてみます。エージェントが Identity Server にリダイレクトされるようであれば、上記手順は適切に実行されています。

IIS 4.0 ポリシーエージェントのトラブルシューティング

インストール時に問題が発生した場合は、次の手順を実行します。

- インストールのログファイルでエラーを確認します。

  `%TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall.`*nnnn*

- アンインストールを実行し、もう一度インストールを実行します。
- IIS にエージェントが読み込まれているかを確認します。

  a. インターネットサービスマネージャを起動します。

  b. Windows の「スタート」メニューで、「プログラム」>「Windows NT Option Pack」>「Microsoft Internet Information Server」>「インターネットサービスマネージャ」を選択します。

     これで、Microsoft Management Console が起動します。

  c. ウィンドウの左側で、コンピュータのホスト名を右クリックして、「プロパティ」を選択します。

  d. 「マスタ プロパティ」セクションで、「WWW サービス」を選択して、「編集」をクリックします。

  e. 表示された「WWW サービス マスタ プロパティ」ダイアログで、「ISAPI フィルタ」タブを選択します。

  f. 「Sun ONE Identity Server Agent」というフィルタを探します。

     「Sun ONE Identity Server Agent」フィルタが見つからない場合は、インストールプログラムが実行され、インストール中にエラーが発生しなかったかを確認します。インストールのログは、次の場所に記録されています。

     `%TEMP%¥Sun_ONE_Identity_Server_Policy_Agent_uninstall.`*nnnn*

     「Sun ONE Identity Server Agent」という名前の右にある「状態」列に緑色の上向き矢印が表示されている場合は、エージェントが IIS に正しく読み込まれています。赤い下向き矢印が表示している場合は、フィルタが正しく読み込まれていません。フィルタが読み込まれない (赤い矢印) 原因として可能性が高いのは、必要な dll ファイルが見つからない場合です。

  g. システムパスをチェックして、次のディレクトリがあることを確認します。

     `# ¥`*Agent_Install_Dir*`¥bin`

  h. フィルタが正しく読み込まれていない場合は、次のことを確認します。

     - 「Sun ONE Identity Server Agent」をクリックし、「編集」をクリックしてエージェントの DLL ファイルのパスを確認します。「実行ファイル」テキストボックスに示されたパスが有効であることを確認します。

- エージェントは、ほかにもいくつかの DLL ファイルを使用します。次のファイルが `lib` ディレクトリにあることを確認します。

  ```
  amsdk.dll
  ames6.dll
  libnspr4.dll
  libplc4.dll
  libplds4.dll
  libxml2.dll
  nss3.dll
  ssl3.dll
  ```

  このライブラリがシステムパスに含まれている場合は、システムを再起動してみます。

- フィルタの読み込みエラーのログは、システムイベントログに記録されます。イベントログを確認するには、次の手順を実行します。

  a. 「スタート」メニューから、「プログラム」>「管理ツール」>「サービス」を選択します。

  b. 「System Log」を選択します。

  c. 「ソース」が「W3SVC」であるエラーメッセージを確認します。

- エージェントは読み込まれているが、IIS Web サーバに対するすべての URL リクエストで「HTTP 500 Internal Server Error」が返される場合。この場合は、エージェントは読み込まれていますが、正しく初期化されていません。すべての HTTP リクエストで「HTTP 500 Internal Server Error」が返されるのは、エージェントが初期化できなかったときに URL リソースを保護するためのフェイルセーフ機能が働くためです。この原因として可能性が高いのは、Sun ONE Identity Server エージェントまたはサーバの設定に問題があるか、これらが使用不能である場合です。

- エージェントのデバッグログを確認します。

  このログは、デフォルトで *Agent_Install_Dir* ディレクトリに置かれます。初期設定やエージェントの動作に関する問題を解決するときは、デバッグ情報が最も役立ちます。ログファイルのディレクトリは、次のプロパティによって指定されます。
  `com.sun.am.policy.am.logFile`

  `AMAgent.properties` ファイルは、デフォルトで次のディレクトリに置かれます。

  Agent_Install_Dir¥`iis`¥`config`¥*_PathInstanceName*

  Agent_Install_Dir¥`iis`¥`config`¥*_PathInstanceName*

com.sun.am.policy.am.loglevels プロパティは、ログ情報の記録量を制御します。ログカテゴリごとのログレベルを設定します。

この値の書式は次のとおりです。

ModuleName [: *Level*] , *ModuleName* [: *Level*]] *

現在使用中のモジュール名は、AuthService、NamingService、PolicyService、SessionService、PolicyEngine、ServiceEngine、Notification、PolicyAgent、RemoteLog、all です。ログレベルの指定を省略すると、デフォルトのログレベルでログモジュールが作成されます。これは、「all」モジュールに設定されているログレベルです。

all モジュールを使うことで、全モジュールのログレベルを設定することができます。また、後で作成されるすべてのモジュールにもこのデフォルトのログレベルが設定されます。次に、「Level」の値について説明します。

com.sun.am.policy.am.loglevels プロパティは、ログ情報の記録量を制御します。次に、「Level」の値について説明します。

0 = 特定モジュールのログ記録を無効にする *

1 = エラーメッセージを記録する

2 = 警告とエラーのメッセージを記録する

3 = 情報、警告、エラーのメッセージを記録する

4 = デバッグ、情報、警告、エラーのメッセージを記録する

5 = レベル 4 と同様だが、より多くのデバッグメッセージを記録する

- エージェントが AMAgent.properties 設定ファイルを探すことができることを確認します。

  エージェントはレジストリキー HKEY_LOCAL_MACHINE¥Software¥Sun Microsystems¥Identity Server IIS Agent を使って、AMAgent.properties ファイルの場所を特定します。AMAgent.properties ファイルは、次のディレクトリにあります。

  Agent_Install_Dir¥iis¥config¥_*PathInstanceName*

- エージェントは、AMAgent.properties に指定されたデバッグログファイルが開始される前に発生したエラーのログを、アプリケーションイベントログを使って記録します。

  a. 「スタート」メニューから、「プログラム」>「管理ツール」>「サービス」を選択します。

  b. 「Application Log」を選択します。

  c. 「ソース」が「Sun ONE Identity Server IIS Agent」であるエラーメッセージを確認します。

古いインストールを削除したのにエージェントをインストールできない

エージェントのインストーラを実行すると、次のようなメッセージが表示されます。

```
"Sun ONE Identity Server Policy Agent 2.0 for Microsoft Internet
Information Services is installed. Please refer to installation
manual to configure this agent for another web server instance.
Or uninstall it before installing another agent."
```

考えられる原因

- エージェントの既存のインストールが残っている
- 以前にエージェントをインストールしたが、そのエージェントのアンインストーラを使わずにエージェントをアンインストールした
- インストーラの productregistry ファイルが破損している

解決法

- エージェントのすべての既存インストールがアンインストールされていることを確認します。
- エージェントの既存インストールが見つからない場合は、productregistry ファイルが破損している可能性があります。このファイルは、インストーラがインストール製品を追跡するときに使われます。このファイルは、C:¥WINT¥system32 ディレクトリに保存されています。

注 変更を行う前にこのファイルのバックアップコピーを作成してください。

このファイルからエージェント製品を削除します。この項目は、次の行から始まります。

```
<compid>SUNWamcom
        <compversion>2.0
            <uniquename>SUNWamcom</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent uninstall script
        <compversion>2.0
            <uniquename>Agent uninstall script</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent installer resource bundle
        <compversion>2.0
```

```
            <uniquename>Agent installer resource
bundle</uniquename>
            <vendor>Sun Microsystems, Inc.</vendor>
            ......
    </compid>
    <compid>Agent Common Core and SDK
        <compversion>2.0
            <uniquename>Agent Common Core and SDK</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>SUNWames6
        <compversion>2.0
<uniquename>SUNWames6</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Agent for ...
        <compversion>2.0
            <uniquename>Agent for ...</uniquename>
            <vendor></vendor>
            ......
    </compid>
    <compid>Sun ONE Identity Server Policy Agent
        <compversion>2.0
            <uniquename>Sun ONE Identity Server Policy
Agent</uniquename>
    </compid>
```

Windows の「スタート」>「設定」>「コントロールパネル」>「アプリケーションの追加と削除」を使ってエージェントをアンインストールできない

考えられる原因 : Java のクラスパスがマシンに正しく設定されていません。

解決法 : 次の手順を実行してエージェントをアンインストールします。

1. コマンドプロンプトウィンドウを開きます。
2. エージェントがインストールされているディレクトリに移動します。
3. `java uninstall_Sun_ONE_Identity_Server_Policy_Agent` を実行します。

既知の問題

個々の Web サイトをシャットダウンしようとすると、インターネットサービスがハングする。
回避策

Web サイトを個別にシャットダウンしないように強くお勧めします。シャットダウンは IIS Admin Service を停止して一括して行い、IIS Admin Service を再起動してから、個別に管理されているサービスを再起動してください。

1. IIS Admin Service を停止するには、コマンド行に次のコマンドを入力します。

   ```
   c:¥>net stop iisadmin /y
   ```

   別の方法として、「サービス」メニューから IIS Admin Service をシャットダウンすることもできます。

   a. 「スタート」メニューから、「設定」>「コントロールパネル」を選択します。

   b. 「サービス」をクリックします。

   c. 「IIS Admin Service」を選択します。

   d. 「停止」をクリックします。

   これにより、FTP サービス、WWW サービス、および SMTP サービスなど、`iisadmin` プロセスが管理するすべてのインターネットサービスがシャットダウンされます。

2. IIS Admin Service を再起動するには、コマンド行に次のコマンドを入力します。

   ```
   c:¥>net start iisadmin
   ```

   別の方法として、「サービス」メニューからサービスを再起動することもできます。

   a. 「スタート」メニューから、「設定」>「コントロールパネル」を選択します。

   b. 「サービス」をクリックします。

   c. 「IIS Admin Service」を選択します。

   d. 「開始」をクリックします。

3. 個々のサービスを再起動するには、コマンド行に次のコマンドを入力します。

   ```
   c:¥>net start w3svc
   ```

   別の方法として、「サービス」メニューから個々のサービスを再起動することもできます。

   a. 「スタート」メニューから、「設定」>「コントロールパネル」を選択します。

   b. 「サービス」をクリックします。

c. 「World Wide Web Publishing」を選択します。

d. 「開始」をクリックします。

IIS 4.0 と終了の問題

IIS 4.0 にポリシーエージェントをインストールすると、各 Web サイトを終了したときに、メモリの衝突に関するメッセージが表示されることがあります。このメッセージは無視しても問題ないので、IIS サーバを再起動してください。

第 5 章

Red Hat Linux 7.2 のポリシーエージェント

Sun ONE Identity Server のポリシーエージェントは、Sun ONE Identity Server と連携して企業の Web サーバに対するユーザアクセスの可否を制御します。この章では、Red Hat Linux 7.2 オペレーティングシステムで稼働する Apache 1.3.26 サーバに Sun ONE Identity Server のポリシーエージェントをインストールする方法について説明します。

次のトピックがあります。

- 始める前に
- Posix Threads による Apache Web サーバの設定
- グラフィカルユーザインタフェースによるインストール
- コマンド行によるインストール
- 複数の Web サーバインスタンス用のエージェント設定
- SSL (Secure Sockets Layer) とエージェントの使用
- REMOTE_USER サーバ変数の設定
- クライアント IP アドレスの検証
- 共有シークレットの暗号化ユーティリティ
- トラブルシューティング

始める前に

第 1 章「ご使用にあたって」で説明されている概念を良く理解しておいてください。この章には、次のトピックスに関する簡単ですが重要な情報があります。

- ポリシーエージェントの動作
- JRE (Java Runtime Environment) 1.3.1 要件
- Sun ONE Identity Server サービスを実行する Web サーバとリモート Web サーバ
- 同一コンピュータシステム上の複数の Web サーバインスタンスのためのエージェントの設定
- Sun ONE Identity Server エージェントのフェイルオーバ機能の提供
- エージェントキャッシュの更新
- グローバル不適用 URL リスト
- グローバル不適用 IP アドレスリスト
- ポリシーを適用しない認証だけの適用
- HTTP ヘッダーを介した LDAP ユーザ属性の転送
- `AMAgent.properties` ファイル
- 完全指定ドメイン名の設定
- CDSSO の設定

Posix Threads による Apache Web サーバの設定

エージェントをインストールする前に、次のタスクを次の順序で実行し、Posix Threads ライブラリによる Apache Web サーバの設定を完了しておく必要があります。これらのタスクを実行しない場合、アプリケーションを利用できなかったり、システム全体が不安定になる、または利用できなくなることがあります。

1. `http://httpd.apache.org/` から Apache ソースを取得します。
2. 次のディレクトリにある `Configure` ファイルを編集します。

   /Apache_root`/src`
3. `Configure` ファイルから `linux22` という項目を探します。
4. `LIBS` 変数の `-lm` の後に `-lpthread` を追加します。

```
*-linux22)
       # This handles linux 2.2 and above (2.4, ...)
       DEF_WANTHSREGEX=yes
       OS='Linux'
       CFLAGS="$CFLAGS -DLINUX=22"
       LIBS="$LIBS -lm -lpthread"
```

5. 変更内容を保存してファイルを閉じます。
6. 次のディレクトリにある Configure スクリプトを実行します。

   /Apache_root/
7. Apache Web サーバを再構築し、インストールします。
8. Apache エージェントをインストールします。

グラフィカルユーザインタフェースによるインストール

ポリシーエージェントのインストール

エージェントのインストールプログラムを実行するには、root 権限が必要です。

1. 次のコマンドを使って製品のバイナリファイルを解凍します。

   ```
   # gunzip -dc agent_Linux_apache.tar.gz | tar -xvof -
   ```
2. setup プログラムを実行します。このプログラムは、バイナリファイルを解凍したディレクトリにあります。コマンド行に次のコマンドを入力します。

   ```
   # ./setup
   ```
3. 開始ページで、「Next」をクリックします。
4. ライセンス契約書を確認してください。「Yes」をクリックして、ライセンスの条項に同意します。

   エージェントをインストールするディレクトリを選択する場合は、「Browse」をクリックします。デフォルトのディレクトリを使用する場合は、「Next」をクリックします。

5. プロンプトに従って、このエージェントが保護する Web サーバに関する次の情報を入力します。

   Install Sun ONE Identity Server Policy Agent in this directory: このエージェントをインストールするディレクトリの完全パスを入力して、「Next」をクリックします。

   Host Name: Web サーバがインストールされているマシンの完全指定のドメイン名を入力します。たとえば、`mycomputer.siroe.com` などとなります。

   Apache Configuration Directory: `httpd.conf` ファイルが保存されている Apache サーバ設定ディレクトリを指定します。

   Web Server Port: エージェントで保護する Web サーバのポート番号を入力します。

   Web Server Protocol: Web サーバを SSL を使用するように設定している場合は、「HTTPS」を選択します。SSL を使用していない場合は、「HTTP」を選択します。

   Agent Deployment URI: ディレクトリ名を入力します。デフォルトの URI (Universal Resource Identifier) は `/amagent` です。

   SSL Ready: Apache Web サーバが SSL をサポートしている場合は、このオプションを選択します。`mod_ssl` をサポートし、EAPI ルールを使ってソースがコンパイルされている Apache Web サーバは、SSL 対応と見なされます。

   Apache Web サーバのコンパイルに EAPI フラグが使われているかどうかを調べるときは、Apache Web サーバの `bin` ディレクトリに移動して、次のコマンドを実行します。

   ```
   # ./httpd -V
   ```

   Apache Web サーバがコンパイルされた各種フラグが表示されます。この一覧に `-D EAPI` というフラグが含まれる場合は、使用中の Apache Web サーバは SSL をサポートしています。ただし、このフラグが見つからない場合でも、Apache Web サーバが `mod_ssl` に対応していることがあります。

   Apache Web サーバの次の設定がサポートされています。

   a. `mod_ssl` に対応していない Apache Web サーバ

   b. `mod_ssl` に対応し、EAPI フラグが有効な Appache Web サーバ

注 Sun ONE Identity Server Policy Agent は、`mod_ssl` に対応していても EAPI フラグが無効な Apache Web サーバをサポートしていません。

すべての情報を正しく入力したら、「Next」をクリックします。

6. Sun ONE Identity Server のポリシーおよび管理機能が稼動する Web サーバに関する情報を入力します。ポリシーエージェントはこのサーバに接続します。

   Primary Server Host: Sun ONE Identity Server を実行するプライマリ Web サーバがインストールされているシステムの完全指定ドメイン名を入力します。たとえば、`myserver.siroe.com` などとなります。

   Primary Server Port: Sun ONE Identity Server を実行する Web サーバのポート番号を入力します。

   Primary Server Protocol: Sun ONE Identity Server を実行する Web サーバで SSL が有効になっている場合は、「HTTPS」を選択します。SSL が有効になっていない場合は、「HTTP」を選択します。

   Primary Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

   Primary Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

   Failover Server Host: プライマリ Web サーバが使用不能になった場合に Sun ONE Identity Server を実行するセカンダリ Web サーバの完全指定ドメイン名を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Port: Sun ONE Identity Server を実行するセカンダリ Web サーバのポート番号を入力します。フェイルオーバホストが存在しない場合は、このフィールドを空白のままにしておきます。

   Failover Server Deployment URI: Sun ONE Identity Server をインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amserver` です。

   Failover Console Deployment URI: Sun ONE Identity Server コンソールをインストールしたときに指定した場所を入力します。Sun ONE Identity Server のデフォルトの URI は `/amconsole` です。

   Agent Identity Server Shared Secret: Identity Server の内部 LDAP 認証ユーザのパスワードを入力します。

   Re-enter Shared secret: Identity Server の内部 LDAP 認証ユーザのパスワードをもう一度入力します。

   CDSSO Enabled: CDSSO 機能を有効にするときは、このボックスにチェックマークをつけます。

   CDSSO Component URL: CDSSO コンポーネントの URL を入力します。

   すべての情報を正しく入力したら、「Next」をクリックします。

7. 「Installation Summary」を見直して、入力した情報が正しいことを確認します。変更が必要な場合は、「Back」をクリックします。すべての情報を正しく入力したら、「Next」をクリックします。
8. 「Ready to Install」のページで、「Install Now」をクリックします。
9. インストールが終了したら、「Details」をクリックしてインストールの詳細を確認するか、「Close」をクリックしてインストールプログラムを終了します。
10. インストールを完了するには、Apache Web サーバを再起動する必要があります。

ポリシーエージェントのアンインストール

次の手順を実行してポリシーエージェントをアンインストールします。

1. エージェントがインストールされているディレクトリに移動して、コマンド行に次のコマンドを入力します。

   ```
   # ./uninstall_linux_apache_agent
   ```

2. 開始パネルで、「Next」をクリックします。
3. 「Uninstall Now」をクリックします。
4. アンインストールが終了したら、「Close」をクリックします。

コマンド行によるインストール

グラフィカルユーザインタフェース (GUI) バージョンの代わりに、コマンド行バージョンのインストールプログラムを使用することもできます。

ポリシーエージェントのインストール

エージェントのインストールプログラムを実行するには、root 権限が必要です。

1. 次のコマンドを実行して tar ファイルを解凍します。

   ```
   # gunzip -dc agent_Linux_apache.tar.gz | tar -xvof -
   ```

2. `setup` プログラムを実行します。このプログラムは、バイナリファイルを解凍したディレクトリにあります。コマンド行に次のコマンドを入力します。

   ```
   # ./setup -nodisplay
   ```

3. プロンプトが表示されたら、次の情報を入力します。

   Have you read, and do you accept, all of the terms of the preceding Software License Agreement?: 「yes」と入力します。

   Install Sun ONE Identity Server Agent in this directory: ポリシーエージェントを入力するディレクトリの完全パスを入力します。

4. このエージェントが保護する Web サーバに関する次の情報を入力します。

   - `Web Server Host Name`
   - `Apache Configuration Directory`
   - `Web Server Port`
   - `Web Server Protocol`
   - `Agent Deployment URI`
   - `SSL Ready`

   これらの各項目の詳細は、「ポリシーエージェントのインストール」を参照してください。

5. Sun ONE Identity Server を実行する Web サーバに関する次の情報を入力します。

   - `Primary Server Host`
   - `Primary Server Port`
   - `Primary Server Protocol`
   - `Primary Server Deployment URI`
   - `Primary Console Deployment URI`
   - `Failover Server Host`
   - `Failover Server Port`
   - `Failover Server Protocol`
   - `Failover Server Deployment URI`
   - `Failover Console Deployment URI`
   - `Agent-Identity Server Shared secret`
   - `Re-enter Shared secret`
   - `CDSSO Enabled`
   - `CDSSO Component URL`

   これらの各項目の詳細は、「ポリシーエージェントのインストール」を参照してください。

6. 次のメッセージが表示されます。

```
Ready to Install

1. Install Now
2. Start Over
3. Exit Installation
```

「What would you like to do?」というメッセージが表示されたら、1 を入力してインストールを開始します。

7. 次のメッセージが表示されます。

```
Product                              Result      More Information
1. Sun ONE Identity Server Agent     Installed   Available
2. Done
```

ログ情報を表示するときは、1 を入力します。インストールプログラムを終了するときは、2 を入力します。

ポリシーエージェントのアンインストール

1. *Agent_Install_Dir* ディレクトリで、コマンド行に次のコマンドを入力します。

   ```
   # ./uninstall_linux_apache_agent -nodisplay
   ```

2. 次のメッセージが表示されます。

```
The uninstaller has detected the following agents on this system:
1. Agent 2.0 for Apache 1.3.26 [/usr/local]
2. Exit
Please select an installed agent from the following list:
```

製品を削除するときは、1 を入力します。

3. 次のメッセージが表示されます。

```
Ready to Uninstall

1. Uninstall Now
2. Start Over
3. Exit Uninstallation
```

「What next?」というプロンプトが表示されたら、1 を入力してアンインストールを開始します。

4. 次のメッセージが表示されます。

```
Product                                    Result More  Information
1. Sun ONE Identity Server Policy Agent    Full         Available
2. Done
```

エージェントのログ情報を表示するときは、1 を入力します。アンインストールプログラムを終了するときは、2 を入力します。

複数の Web サーバインスタンス用のエージェント設定

1 台のコンピュータ上で稼働する複数の Web サーバインスタンスに合わせてエージェントを設定するには、グラフィカルユーザインタフェース (GUI) を使用するか、またはコマンド行でエージェントインストールプログラムを使って最初のエージェントをインストールします。最初のエージェントをインストールしたら、config スクリプトを使ってあとのエージェントをインストールできます。このスクリプトは、次の節で説明するように、コマンド行から実行する必要があります。

同じコンピュータシステムに複数の Web サーバインスタンス用のエージェントを設定するには

システムにエージェントを 1 つインストールしたら、エージェントのインストール時にシステムにコピーされるスクリプトを使用して、あとのエージェントをそのシステムにインストールできます。次のディレクトリに、2 つのスクリプト、config_linux と unconfig_linux があります。

Agent_Install_Dir/agents/apache/bin

元のエージェントをインストールした後で追加のエージェントをシステムにインストールするには、次のコマンドを使って、bin ディレクトリから config_linux スクリプトを実行します。

```
# ./config_linux
```

プロンプトに従って、追加のエージェントをインストールします。各プロンプトについては、「グラフィカルユーザインタフェースによるインストール」を参照してください。一般に、保護された Apache サーバインスタンスと Sun ONE Identity Server サーバの両方の情報を入力する必要があります。次のテキストは実行例です。

```
# ./config_linux
Enter the Apache Server Configuration Directory:
[/etc/httpd/conf]
SSL Ready: [true] false
Enter the Local Hostname:[mycomputer.siroe.com]
Enter the Agent Web Server Port: [80]
Select Agent Web Server Protocol:[1] http [2] https-->[1]
Enter the Agent Deployment URI:[/amagent]
Select Identity Server Protocol:[1] http [2] https --> [1]
Enter the Identity Server Hostname:[mycomputer.siroe.com]
Enter the Identity Server Port: [58080]
Enter the Identity Server Deployment URI [/amserver]
Enter the Identity Server's Console Deployment URI [/amconsole]
Enter the Identity Server's Console Deployment URI [/amconsole]
```

```
Select Failover Identity Server Protocol: [1] http [2] https [3]
no failover --> []
Enter the Failover Identity Server Hostname:
[]mycomputer.siroe.com
Enter the Failover Identity Server Port: []
Enter the Identity Server Deployment URI [/amserver]
Enter the Identity Server's Console Deployment URI [/amconsole]
Enter Agent-Identity Server shared secret:
Re-enter Agent-Identity Server shared secret:
Configuring Apache Web Server ...
Done
```

注 config スクリプトでは、CDSSO の詳細を入力できません。これは、手動で設定する必要があります。詳細については、「CDSSO の設定」を参照してください。

config スクリプトによるサイレントインストール

config_linux スクリプトを使用して、サイレントの非対話型エージェントインストールを実行することもできます。このスクリプトの使用方法については、config_linux -h コマンドを使用して詳細を表示してください。

```
# ./config_linux -h
Usage:config [ -r response_file | -R | -h ]
        -r specifies a response file.
        -R prints out the response file template.
        -h prints out this message.
```

サイレントインストールを実行するには、インストールするエージェントごとに応答ファイルを指定する必要があります。config_linux -R コマンドは、応答ファイルで指定しなければならないフィールドを示します。このテキストファイルは、サイレントインストールを開始する前に用意しておく必要があります。

```
./config_linux -R
Response file contains:
AGENT_PROTOCOL              # agent protocol:http|https
AGENT_HOST                  # agent hostname
AGENT_PORT                  # agent server port
```

```
AGENT_DEPLOY_URI            # agent deploy URI
FAILOVER_SERVER_HOST        # failover identity server name
FAILOVER_SERVER_PORT        # failover identity server port
FAILOVER_SERVER_DEPLOY_URI # failover identity server deploy URI
FAILOVER_CONSOLE_DEPLOY_URI # failover identity server console deploy URI
PRIMARY_SERVER_HOST         # primary identity server name
PRIMARY_SERVER_PORT         # primary identity server port
PRIMARY_SERVER_PROTO        # primary identity server protocol:http|https
PRIMARY_SERVER_DEPLOY_URI   # primary identity server deploy URI
PRIMARY_CONSOLE_DEPLOY_URI # primary identity server console deploy URI
SHARED_SECRET               # shared secret between agent and DSAME server
SERVER_INSTANCE             # web server instance directory
NOTIFICATION_ENABLE         # notification enabled
AGENT_URL_CASE_IGNORE       # url comparison case ignore
```

次に、response.apache という名前の応答ファイルの例を示します。

```
AGENT_PROTOCOL=http
AGENT_HOST=mycomputer.siroe.com
AGENT_PORT=80
AGENT_DEPLOY_URI=/amagent
FAILOVER_SERVER_HOST=failover_computer.siroe.com
FAILOVER_SERVER_PORT=58080
FAILOVER_SERVER_DEPLOY_URI=/amserver
FAILOVER_CONSOLE_DEPLOY_URI=/amconsole
PRIMARY_SERVER_HOST=primary_computer.siroe.com
PRIMARY_SERVER_PORT=58080
PRIMARY_SERVER_PROTO=http
PRIMARY_SERVER_DEPLOY_URI=/amserver
PRIMARY_CONSOLE_DEPLOY_URI=/amconsole
SHARED_SECRET=encrypted_shared_secret
SERVER_INSTANCE=/Agent_Install_Dir/apache26/conf
NOTIFICATION_ENABLE=true
AGENT_URL_CASE_IGNORE=true
```

次の例は、response.apache 応答ファイルと組み合わせて config_linux スクリプトを使って、サイレントインストールを実行する方法を示しています。

```
# ./config_linux -r response.apache
Configuring Apache Web Server ...
done
```

注 config_linux スクリプトを使ってインストールしたエージェントは、必ず unconfig_linux スクリプトを使用してアンインストールしてください。コマンド行からインストールしたエージェントのアンインストールに GUI インストールプログラムを使うことはできません。GUI のアンインストールプログラムは、コマンド行で unconfig スクリプトを使ってインストールしたすべてのエージェントをアンインストールしてから実行する必要があります。

unconfig スクリプトによるエージェントの削除

config_linux スクリプトを使ってコマンド行からインストールしたエージェントを削除するには、unconfig_linux スクリプトを使います。unconfig_linux スクリプトは、次のディレクトリにあります。

Agent_Install_Dir/agents/apache/bin

次に、unconfig_linux スクリプトの実行例を示します。

```
# ./unconfig_linux /web_server_root/httpd/conf
Unconfiguring webserver ... ¥c
done.
```

SSL (Secure Sockets Layer) とエージェントの使用

インストール時に HTTPS プロトコルを選択すると、SSL を介して通信するようにエージェントが自動的に設定されます。

注	次の手順に進む前に、Web サーバに SSL が設定されていることを確認してください。

エージェントのデフォルトの信頼動作

デフォルトでは、リモートの Apache Server 1.3.26 にインストールされたポリシーエージェントは、Sun ONE Identity Server を実行する Web サーバが SSL 上で提示したサーバ認証書を信頼します。エージェントはルートの認証局 (CA) 証明書をチェックしません。Sun ONE Identity Server を実行する Web サーバで SSL が有効になっていて、ポリシーエージェントで証明書をチェックしたい場合は、次の処理を行う必要があります。

1. エージェントのデフォルトの信頼動作を無効にします。
2. エージェントがインストールされているリモート Web サーバにルート CA 証明書をインストールします。ルート CA 証明書は、Sun ONE Identity Server サービスを実行する Web サーバにインストールされているものと同じでなければなりません。

エージェントのデフォルト信頼動作の無効化

次のプロパティは `AMAgent.properites` ファイルにあり、デフォルトでは `true` に設定されています。

```
com.sun.am.policy.amcpa.trustServerCerts=true
```

これは、エージェントが証明書のチェックを行わないということです。

デフォルトの動作を無効にするには

次のプロパティを `false` に設定する必要があります。

```
com.sun.am.policy.amcpa.trustServerCerts=false
```

リモート Web サーバへのルート CA 証明書のインストール

リモート Web サーバにインストールするルート CA 証明書は、Sun ONE Identity Server を実行する Web サーバにインストールされているものと同じでなければなりません。

ルート CA 証明書を Apache 1.3.26 にインストールするには

`certutil` プログラムを使って、ルート CA 証明書を Apache 1.3.26 にインストールできます。

1. C シェルで、コマンド行に次のコマンドを入力します (設定ファイルが保存されているディレクトリを `/etc/apache` とします)。

   ```
   # cd /etc/apache/cert

   # setenv LD_LIBRARY_PATH
   /Agent_Install_Dir/agents/apache/lib:/Agent_Install_Dir/agents/lib
   ```

2. 必要に応じて証明書データベースを作成します。

   ```
   # /Agent_Install_Dir/agents/apache/cert/certutil -N -d .
   ```

3. ルート CA 証明書をインストールします。

   ```
   # /Agent_Install_Dir/agents/apache/cert/certutil -A -n cert-name -t
   "C,C,C" -d cert-dir -i cert-file
   ```

   上のコマンドで、変数は次のものを表します。

   - *cert-name* には、このルート証明書の任意の名前を指定します。
   - *cert-dir* には、証明書関連のファイルが置かれたディレクトリを指定します。
   - *cert-file* には、Base64 で符号化されたルート証明書ファイルを指定します。

   `certutil` ユーティリティの詳細は、「`certutil -H`」と入力してオンラインヘルプを参照してください。

4. 証明書が正しくインストールされたことを確認するには、コマンド行に次のように入力します。

```
# ./certutil -L -d .
```

インストールしたルート CA 証明書の名前を含む信頼データベース情報が表示されます。次に例を示します。

```
Certificate Name                                    Trust Attrubutes

cert-name                                           C,C,C

p    Valid peer
P    Trusted peer (implies c)
c    Valid CA
T    Trusted CA to issue client certs (implies c)
C    Trusted CA to certs(only server certs for ssl) (implies c)
u    User cert
w    Send warning
```

REMOTE_USER サーバ変数の設定

REMOTE_USER サーバ環境変数は、Sun ONE Identity Server の認証ユーザまたは匿名ユーザに設定できます。この変数を特定のユーザに設定することによって、Web アプリケーション (CGI、サーブレット、ASP プログラムなど) をそのユーザが利用できるようになります。この機能によって、特定のユーザに表示される HTML ページのコンテンツをパーソナライズできます。

AMAgent.properties ファイルで指定されたグローバル不適用 URL (認証されていないユーザがアクセスできる URL) に対して REMOTE_USER 設定を有効にするには、AMAgent.properties ファイルの次のプロパティを TRUE (デフォルトでは、FALSE) に設定する必要があります。

```
com.sun.am.policy.agents.anonRemoteUserEnabled=TRUE
```

このプロパティ値を TRUE に設定すると、REMOTE_USER の値は、AMAgent.properties ファイルの次のプロパティに含まれる値 (デフォルトでは、anonymous) に設定されます。

```
com.sun.am.policy.agents.unauthenticatedUser=anonymous
```

クライアント IP アドレスの検証

この機能を使用して、SSO トークンの盗難や「ハイジャック」を防ぎ、セキュリティを向上させることができます。

`AMAgent.properties` ファイルには `com.sun.am.policy.agents.client_ip_validation_enable` というプロパティが含まれており、デフォルトでは、このプロパティは `false` に設定されています。

このプロパティの値を `true` に設定すると、SSO トークンを含む各着信要求に対して、クライアント IP アドレスの検証が有効になります。要求の生成元の IP アドレスが SSO トークンの発行先の IP アドレスと一致しない場合、要求は拒否されます。これは基本的に、拒否ポリシーの適用と同じです。

ただし、クライアントブラウザが Web プロキシを使っている場合、またはクライアントブラウザとエージェントが保護する Web サーバとの間に負荷均衡アプリケーションがある場合は、この機能を使用しないでください。そのような場合、要求に現れる IP アドレスは、クライアントブラウザが稼動している実際の IP アドレスを反映しません。

共有シークレットの暗号化ユーティリティ

ポリシーエージェントは、共有シークレットを `AMAgent.properties` ファイルに保存します。このパスワードのデフォルトは、Indentity Server の内部 LDAP 認証ユーザのパスワードです。これは、サーバ側で `AMConfig.Properties` ファイルを編集することで変更できます。

`AMConfig.Properties` ファイルの `com.sun.am.policy.am.password` プロパティには、エージェントのインストール時に暗号化された共有シークレットを設定できます。

共有シークレットをリセットまたは変更するときは、次のユーティリティを使ってプロパティに値を設定します。

1. 次のディレクトリに移動します。

   Agent_Install_Dir`/bin`

2. コマンド行から次のスクリプトを実行します。

   `crypt_util` *shared_secret*

3. 手順 2 の出力をコピーして次のプロパティに貼り付けます。

   `com.sun.am.policy.am.password`

4. Web サーバを再起動し、エージェントが保護するリソースにアクセスしてみます。エージェントが Sun ONE Identity Server にリダイレクトされるようであれば、上記手順は適切に実行されています。

トラブルシューティング

起動時にエラーメッセージが表示される

エージェントをインストールすると、Apache サーバの起動時に次のエラーメッセージが表示されます。

```
Syntax error on line 1 of

/etc/opt/SUNWam/agents/apache/config/_usr_local_apache_conf/dsam
e.conf:

Invalid command 'LoadModule', perhaps mis-spelled or defined by a
module not included in the server configuration

./apachectl start: httpd could not be started
```

解決法: これは、Apache サーバの `mod_so` が無効で、ダイナミックな共有オブジェクトに対応できていないことを意味します。`mod_so` を有効にする方法については、`http://httpd.apache.org/` に用意されている Apache サーバのマニュアルを参照してください。

第 2 部

J2EE エージェント

第 6 章「ご使用にあたって」

第 7 章「WebLogic 6.1 SP2 のポリシーエージェント」

付録 A 「インストーラが実行する設定タスク」

付録 B 「WebLogic ポリシーエージェントのデバッグエンジンの使用」

付録 C 「ロールと主体のマッピングに関するサンプルシナリオ」

第 6 章

ご使用にあたって

Sun ONE Identity Server ポリシーエージェントは、Sun ONE Identity Server サービスを使って認証と承認をアプリケーションサーバに適用します。そのため、ホストしている J2EE アプリケーションへのクライアントからのアクセスが安全に行われ、配備したアプリケーションの配備記述子に定義されている J2EE セキュリティポリシーを適用することができます。

この章では、アプリケーションサーバ向けの Sun ONE Identity Server ポリシーエージェントの概要と、インストールプログラムに進む前に理解しておく必要のあるいくつかの概念について説明します。

次のトピックがあります。

- アプリケーションサーバ向けのポリシーエージェントの動作
- アプリケーションサーバ向けポリシーエージェントの使用
- サポートされるサーバ

アプリケーションサーバ向けのポリシーエージェントの動作

アプリケーションサーバ向けの Sun ONE Identity Server ポリシーエージェントは、アプリケーションサーバの操作と、保護されているアプリケーションの動作に影響を与える次の 2 つの主要コンポーネントから構成されます。

- **エージェントレルム** : エージェントレルムコンポーネントは、アプリケーションサーバが Identity Server のユーザとロールの情報にアクセスする機能を提供します。これはエージェントの中心的機能であり、エージェントが機能するには、正しく設定する必要があります。

- **エージェントフィルタ**：エージェントフィルタコンポーネントは、ホストしているアプリケーションが Sun ONE Identity Server に基づく認証を適用する機能を提供します。また、ログオンしているユーザに関連するセキュリティ主体の作成にも利用されます。エージェントによって保護されるすべてのアプリケーションでは、エージェントフィルタコンポーネントを使用するように、配備記述子の設定を変更する必要があります。この設定がされていないとアプリケーションはエージェントによって保護されず、エージェントレルムコンポーネントがインストールされたアプリケーションサーバに配備された場合、正常に機能しなかったり、利用できなくなることがあります。

エージェントレルムコンポーネントとエージェントフィルタコンポーネントは Identity Server と併用され、保護されている J2EE アプリケーションにアクセスしようとするクライアントに認証と承認を適用します。

アプリケーションサーバ向けポリシーエージェントの使用

アプリケーションサーバ向けの Sun ONE Identity Server ポリシーエージェントをインストールして、ホストするさまざまな J2EE アプリケーション (各種セキュリティポリシーのセットの実装を必要とする場合もあります) を保護することができます。J2EE のセキュリティインフラストラクチャは、宣言およびプログラムによりセキュリティを提供します。これらのセキュリティは、プラットフォームに依存せず、すべての J2EE 互換アプリケーションサーバによりサポートされます。J2EE プラットフォームの宣言およびプログラムによるセキュリティの使用方法については、`http://java.sun.com/j2ee` にある J2EE のマニュアルを参照してください。

エージェントは、保護される J2EE アプリケーションと Sun ONE Identity Server 主体用にロールから主体へのマッピング機能を提供します。このため、実行時に J2EE ポリシーが評価される場合、この評価は Sun ONE Identity Server 側の情報に対して実行されます。この機能を利用することで、ホストする J2EE アプリケーションがエージェントによって保護されるように設定し、本来の意味でのセキュリティサービスと、シングルサインオンなどの重要機能を提供することができます。

例

商用アプリケーション

商用アプリケーションでは、クライアントにさまざまなサービスを提供するために、数多くの専用 Enterprise JavaBean が利用されています。たとえば、注文書の作成機能を提供する専用のコンポーネントなどがあげられます。同様に、注文書の承認機能を提供する専用のコンポーネントも考えられます。これらのコンポーネントはアプリケーションが機能するための基本的なビジネスサービスを提供しますが、これらのサービスを適切に利用するためにはセキュリティポリシーが必要です。

配備記述子を使うことで、アプリケーションのベンダまたは開発者は、抽象的なセキュリティロール名を使ってこのようなコンポーネントを保護することができます。たとえば、「Buyer」という名前のロールを使って、注文書の作成機能を提供するコンポーネントを保護できます。また、「Approver」という名前のロールを使って、注文書の承認機能を提供するコンポーネントを保護できます。これらのロールは、セキュリティポリシーの適用というアプリケーションベンダまたは開発者の意図を内包していますが、Identity Server に存在する実際のユーザやロールなど、本来の主体とこれらの抽象的なロール名がマッピングされていない限り機能することはありません。

エージェントは、抽象的なセキュリティロールと実際の主体を実行時に結びつける機能をコンテナに提供します。エージェントをインストールして設定すると、アプリケーションのセキュリティロールを実際の主体にマップできます。たとえば、「Buyer」というロールを「Staff」という Identity Server ロールにマッピングできます。このため、「Arvind」というユーザがアプリケーション側の保護されたリソースにアクセスを試みた場合に、エージェントは、マッピングされた「Staff」というロールに「Arvind」という実際のユーザが含まれている場合にだけこのアクセスを許可します。

イントラネットの従業員ポータル

イントラネットの従業員ポータルでは、給与情報やオンラインの利益管理などのサービスが提供されます。このようなサービスは、一般の従業員には読み取り専用で提供され、管理者には関連するデータを変更するための特別なアクセス権が付与されるかもしれません。たとえば、給与情報を読み取るサービスと、給与情報を変更するサービスの 2 つを提供する専用の Enterprise JavaBean コンポーネントの使用が考えられます。エージェントを使ってこのアプリケーションを保護することで、管理者には給与情報の変更に必要なアクセス権を付与し、一般の従業員には読み取り専用のアクセス権だけを許可することもできます。

コンテンツベースの Web アプリケーション

コンテンツベースの Web アプリケーションでは、ペイパービュー方式のサービスを提供できます。アプリケーションは、匿名ユーザによるアクセスも許可されるパブリックドメインと、このサービスの利用者だけがアクセスを許可されるプライベートドメインに分割されます。エージェントを使うことで、認証、承認されたユーザだけにアプリケーションのプライベートドメインへのアクセスを許可し、すべてのユーザにパブリックドメインへのアクセスを許可することができます。アプリケーションの機能を提供する特定のサーブレットと JSP は、関連するセキュリティロールと実際の Identity Server 主体をマッピングすることで、エージェントによって保護されます。

サポートされるサーバ

Sun ONE Identity Server のポリシーエージェントは、次のアプリケーションサーバをサポートしています。

- Solaris 8、Windows 2000 Server、HP-UX 11 の各オペレーティングシステムで稼働する WebLogic 6.1 SP2

第 7 章

WebLogic 6.1 SP2 のポリシーエージェント

この章では、WebLogic 6.1 SP2 アプリケーションサーバ向け Sun ONE Identity Server ポリシーエージェントをインストールおよび設定する方法について説明します。次のトピックがあります。

- サポートされるプラットフォーム
- ガイドライン
- エージェントのインストール
- WebLogic Server の設定
- アプリケーションの設定
- エージェントの設定
- エージェントと Sun ONE Identity Server SDK API の使用
- エージェントのアンインストール

サポートされるプラットフォーム

WebLogic 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントは、次のプラットフォームをサポートしています。

- Solaris 8
- Windows 2000 Server
- HP-UX 11

ガイドライン

WebLogic 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントを最も有効に利用するには、次のガイドラインに従うことをお勧めします。

- **エージェントベースの認証の使用**

  エージェントをインストールし、エージェントフィルタコンポーネントを使用するようにアプリケーションを設定すると、保護されているアプリケーションの対象部分に対するすべての Web ベースのアクセスについて、エージェントフィルタコンポーネントは認証を適用します。エージェントレルムコンポーネントと共に使用することで、エージェントフィルタは、保護されているアプリケーションに定義された J2EE ポリシーがロールから主体へのマッピングに基づいて確実に正しく評価されるようにします。同時にシングルサインオン (SSO) などの重要なサービスを提供します。このため、アプリケーションの動作時にエージェントフィルタコンポーネントを使用せずに、独自の認証メカニズムや、コンテナベースのその他の認証メカニズムを保護されたアプリケーションが使用することがないように設定することをお勧めします。

- **セキュリティを認識する拡張アプリケーションの作成**

  エージェントは、Identity Server SDK ライブラリに含まれる豊富な API を提供します。これらの API は、保護されたアプリケーションで利用できます。これらの API を使うことで、アプリケーション設計者は Identity Server が提供するセキュリティフレームワークの範囲内でセキュリティを認識する拡張アプリケーションを自由に作成できます。Sun ONE Identity Server SDK の詳細な使用方法については、『Sun ONE Identity Server Programmer's Guide』を参照してください。

エージェントのインストール

WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントをインストールできるプラットフォームは、Solaris 8、Windows 2000 Server、または HP-UX 11 です。WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントのインストールプログラムを起動する場合、使用するプラットフォームに応じて次の手順に従う必要があります。インストールプログラムが正常に起動した場合は、エージェントのインストールに必要な詳細な手順について説明している次の各項を参照する必要はありません。

インストール前のタスク

WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントをインストールする前に、次のタスクを実行する必要があります。

1. WebLogic Server 6.1 SP2 をインストールします。

   詳細については、WebLogic Server のマニュアルを参照してください。サーバのインストールが完了したら、付属するサンプルアプリケーションを使ってサーバが正しくインストールされていることを確認します。

2. 保護するアプリケーションの配備をテストします。

   エージェントをインストールする前に、保護するアプリケーションを配備し、簡単な機能を実行してテストすることが重要です。環境が整い、アプリケーションを正しく配備できたら、エージェントのインストールを開始できます。

Solaris 8 でのインストールプログラムの起動

Solaris プラットフォーム用の WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントのバイナリファイルは、tar-gzip 形式で圧縮されています。WebLogic Server をインストールしたマシンにこの圧縮ファイルをコピーし、次の手順を実行してインストールプログラムを起動します。

1. root としてログインします。
2. 次のコマンドを使ってバイナリファイルを解凍します。

   ```
   # gzip -dc
   j2eeagents-2.0-domestic-us.sparc-sun-solaris2.8.tar.gz | tar xvf
   -
   ```

3. `JAVA_HOME` 環境変数をバージョン 1.3.1 以降の JDK に設定します。必要なバージョンの JDK がシステムに用意されていない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用します。JDK は、次の場所に保存されています。

   WebLogic_Install_Dir`/bea/jdk131`

   インストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。ほとんどの場合、エージェントのインストールには GUI 形式のインストールプログラムを使います。ただし、リモートサーバの telnet セッションでインストールを行う場合にウィンドウ機能を利用できないときは、コマンド行形式のインストールプログラムを使ってエージェントをインストールします。このインストールプログラムを起動するには、次のコマンドを実行します。

   ```
   # ./setup -nodisplay
   ```

ただし、GUI 形式のインストールプログラムを利用する場合は、GUI インストールプログラムウィンドウが正しいコンソールに表示されるように DISPLAY 環境変数を設定する必要があります。

注　-nodisplay オプションを指定してコマンド行形式のインストールプログラムを使用する場合は、次の手順を省略し、138 ページの「GUI によるエージェントのインストール」に進み、詳細なインストール手順を参照してください。

---

4. setup スクリプトを呼び出して GUI 形式のインストールプログラムを起動します。

   ```
   # ./setup
   ```

   - インストールプログラムを利用するには、手順 3 で説明した方法で JAVA_HOME 変数を正しく設定する必要があります。JAVA_HOME 変数が正しく設定されていない状態で setup スクリプトを実行すると、JAVA_HOME の正しい値を入力するように求められます。

     ```
     Enter JAVA_HOME location (Enter "." to abort):
     ```

     インストールプログラムの起動に使う JDK インストールディレクトリの完全パスを入力します。インストールを中止するときは、ピリオド (.) を入力します。

   - GUI 形式のインストールプログラムをコンソールに表示するには、シェルの DISPLAY 環境変数を正しく設定する必要があります。DISPLAY 環境変数が正しく設定されていない状態で setup スクリプトを実行すると、DISPLAY の正しい値を入力するように求められます。

     ```
     Please enter the value of DISPLAY variable (Enter "." to
     abort):
     ```

     上記プロンプトに DISPLAY 変数の正しい値を入力します。インストールを中止するときは、ピリオド (.) を入力します。

注　agent_SunOS.class ファイルを使ってエージェントをインストールすることもできます。このファイルは、バイナリファイルを解凍したディレクトリにあります。

Windows 2000 Server でのインストールプログラムの起動

Windows プラットフォーム用の WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントのバイナリファイルは、zip 形式で圧縮されています。WebLogic Server をインストールしたマシンにこの圧縮ファイルをコピーし、次の手順を実行してインストールプログラムを起動します。

1. インストールプログラムを実行するには、管理者特権が必要です。管理者特権がない場合は、管理者としてログオンするか、マシンまたはドメインのシステム管理者に依頼して、使用アカウントに適切な権限を付与してもらいます。
2. Zip ユーティリティを使ってエージェントのバイナリファイルを適切な場所に解凍します。これにより、インストールプログラムの起動に使われる 2 種類の実行可能ファイル、`setup.bat` および `setup.exe` が生成されます。それぞれのファイルは、異なる方法でインストールプログラムを起動します。インストール要件に合わせていずれかのファイルを使用することができます。

注 `agent_WINNT.class` ファイルを使ってエージェントをインストールすることもできます。このファイルは、バイナリファイルを解凍したディレクトリにあります。

setup.bat の使用

`setup.bat` ファイルを使ってインストールプログラムを起動するには、バージョン 1.3.1 以降の JDK をシステムパスから利用できる必要があります。これは、コマンドプロンプトウィンドウで次のコマンドを実行することで確認できます。

```
C:¥> java -version

java version 1.3.1_02

Java(TM) 2 Runtime Environment, Standard Edition (build
1.3.1_02-b02)

Java HotSpot(TM) Client VM (build 1.3.1_02-b02, mixed mode)
```

必要なバージョンの JDK をシステムパスから利用できない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用することができます。これは次の場所にあります。

WebLogic_Install_Dir¥bea¥jdk131

`setup.bat` を実行するときは、コマンドプロンプトウィンドウでファイルが保存されている場所まで移動してからファイル名を入力するか、Windows Explorer 上でファイルをダブルクリックします。たとえば、`C:¥>setup.bat` と入力します。

インストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。GUI 形式のインストールプログラムを起動するときは、上で説明したように、コマンドプロンプトウィンドウで `setup.bat` ファイルを呼び出すか、Windows Explorer 上でファイルをダブルクリックします。コマンド行形式のインストールプログラムを起動するときは、次のように、`setup.bat` スクリプトに `-nodisplay` という引数を指定して実行します。

```
C:¥>setup.bat -nodisplay
```

setup.exe の使用

`setup.exe` を使う場合は、適切なバージョンの JDK を環境パスに含める操作は必要ありません。このプログラムは、最初に適切なバージョンの JDK をシステムから検索し、見つかった JDK を使用します。適切なバージョンが見つからなかった場合は、必要なランタイムをインストールした上でインストールプログラムを起動します。

`setup.exe` を実行するときは、コマンドプロンプトから呼び出すか、Windows Explorer 上でファイルをダブルクリックします。`setup.exe` を使って起動できるのは、GUI 形式のインストールプログラムだけです。

注 `setup.exe` を使ってコマンド行形式のインストールプログラムを起動することはできないので、この形式のインストールプログラムが必要な場合は、`setup.bat` を使うことをお勧めします。

HP-UX 11 でのインストールプログラムの起動

HP-UX 11 プラットフォーム用の WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントのバイナリファイルは、tar-gzip 形式で圧縮されています。このファイルを WebLogic Server がインストールされているマシンにコピーします。次の手順を実行して、インストールプログラムを起動します。

1. root としてログインします。
2. 次のコマンドを使ってバイナリファイルを解凍します。

   ```
   # gzip -dc
   j2eeagents-2.0-domestic-us.hppa1.0-hp-hpux11.00.tar.gz | tar xvf
   -
   ```

3. `JAVA_HOME` 環境変数をバージョン 1.3.1 以降の JDK に設定します。必要なバージョンの JDK がシステムに用意されていない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用します。JDK は、次の場所に保存されています。

   WebLogic_Install_Dir`/bea/jdk131`

4. インストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。ほとんどの場合、エージェントのインストールには GUI 形式のインストールプログラムを使います。ただし、リモートサーバの telnet セッションでインストールを行う場合にウィンドウ機能を利用できないときは、コマンド行形式のインストールプログラムを使ってエージェントをインストールすることをお勧めします。このインストールプログラムを起動するには、次のコマンドを実行します。

   ```
   # ./setup -nodisplay
   ```

   ただし、GUI 形式のインストールプログラムを利用する場合は、GUI プログラムウィンドウが正しいコンソールに表示されるように DISPLAY 環境変数を設定する必要があります。

注 -nodisplay オプションを指定してコマンド行形式のインストールプログラムを使用する場合は、次の手順を省略し、インストール手順を詳しく説明している次の項に進んでください。

5. setup スクリプトを呼び出して GUI 形式のインストールプログラムを起動します。

   ```
   # ./setup
   ```

   - インストールプログラムを利用するには、手順 3 で説明した方法で JAVA_HOME 変数を正しく設定する必要があります。JAVA_HOME 変数が正しく設定されていない状態で setup スクリプトを実行すると、JAVA_HOME の正しい値を入力するように求められます。

     ```
     Enter JAVA_HOME location (Enter "." to abort):
     ```

     インストールプログラムの起動に使う JDK インストールディレクトリの完全パスを入力します。インストールを中止するときは、ピリオド (.) を入力します。

   - GUI 形式のインストールプログラムをコンソールに表示するには、シェルの DISPLAY 環境変数を正しく設定する必要があります。DISPLAY 環境変数が正しく設定されていない状態で setup スクリプトを実行すると、DISPLAY の正しい値を入力するように求められます。

     ```
     Please enter the value of DISPLAY variable (Enter "." to
     abort):
     ```

     上記プロンプトに DISPLAY 変数の正しい値を入力します。インストールを中止するときは、ピリオド (.) を入力します。

GUI によるエージェントのインストール

インストールプログラムを起動すると、開始画面が表示されます。質問に答えて、「Next」をクリックして画面を切り替えます。

1. ライセンス契約書を確認してください。インストールを持続するときは、「Yes (Accept License)」をクリックします。
2. 「Select Installation Directory」画面では、インストール先のパスを入力します。

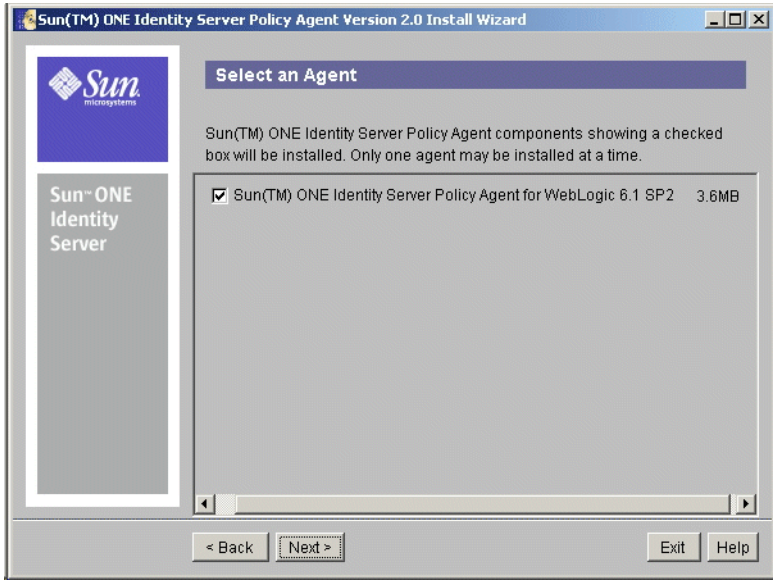
   デフォルトとは異なるディレクトリにエージェントをインストールする場合は、「Browse」ボタンをクリックしてディレクトリを選択します。適切なディレクトリを選択したら、「Next」をクリックして次の画面に進みます。

注　システムに存在しないディレクトリを指定すると、新しいディレクトリを作成するかどうか確認メッセージが表示されます。「Create Directory」ボタンをクリックして新しいディレクトリを作成するか、「Choose another Directory」ボタンをクリックして別のディレクトリを選択します。

---

3. 「Select an Agent」画面では、コンポーネント名のチェックボックスを選択してインストールするコンポーネントを指定します。利用できるコンポーネントは、「Identity Server Policy Agent for WebLogic Server 6.1 SP2」だけです。これはデフォルトで選択されています。

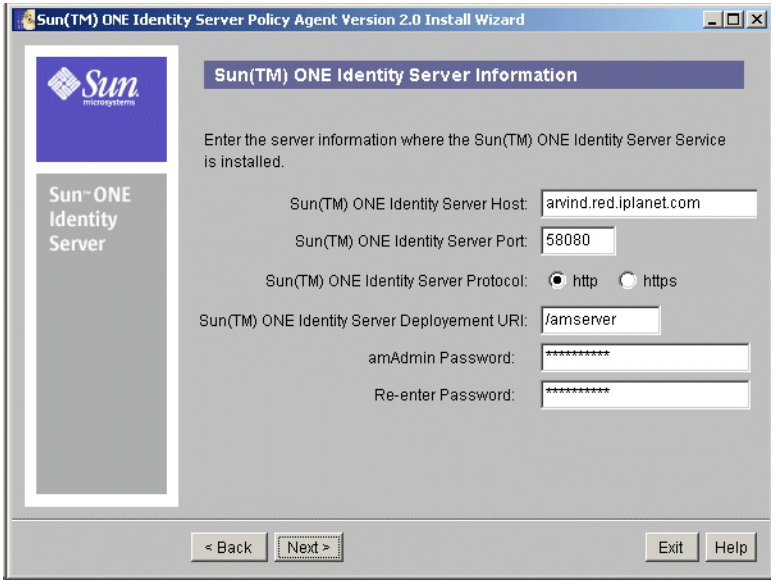
図 7-1　コンポーネント選択画面



注 1 つのシステムで一度にインストールできる WebLogic Server 6.1 SP2 向け Sun ONE Identity Server ポリシーエージェントは 1 つだけです。以前にインストールしたエージェントがシステムから完全に削除されていない場合は、チェックボックスが無効になり、選択できません。「Exit」をクリックしてインストールプログラムを終了し、古いバージョンを完全に削除してからインストールを再開することをお勧めします。

4. 「Sun ONE Identity Server Information」画面では、Sun ONE Identity Server に関する次の情報を入力し、「Next」をクリックします。

図 7-2 「Sun ONE Identity Server Information」画面



Sun ONE Identity Server Host: Sun ONE Identity Server がインストールされているシステムの完全指定ホスト名を入力します。

Sun ONE Identity Server Port: Sun ONE Identity Server サービスを実行する Web サーバのポート番号を入力します。

Sun ONE Identity Server プロトコル : エージェントが Sun ONE Identity Server サービスとの通信に使うプロトコルを選択します。これは、HTTP または HTTPS のいずれかです。

Sun ONE Identity Server Deployment URI: Sun ONE Identity Server サービスへのアクセスに使う URI を入力します。

amAdmin Password: `amAdmin` ユーザのパスワードを入力します。

Re-enter Password: 確認のために amAdmin ユーザのパスワードをもう一度入力します。

注

インストール時に入力したパスワードは、安全な方法で記録されます。ただし、Sun ONE Identity Server でパスワードを変更した場合、エージェントのパスワードも変更する必要があります。パスワード変更には、エージェントに付属する agentadmin ツールを使います。エージェントをシステムにインストールすると、次の場所にある agentadmin ツールが起動されます。

Agent_Install_Dir/SUNWam/wlAgent/bin/agentadmin

Windows プラットフォームでは、agentadmin ツールは agentadmin.bat を使って起動できます。

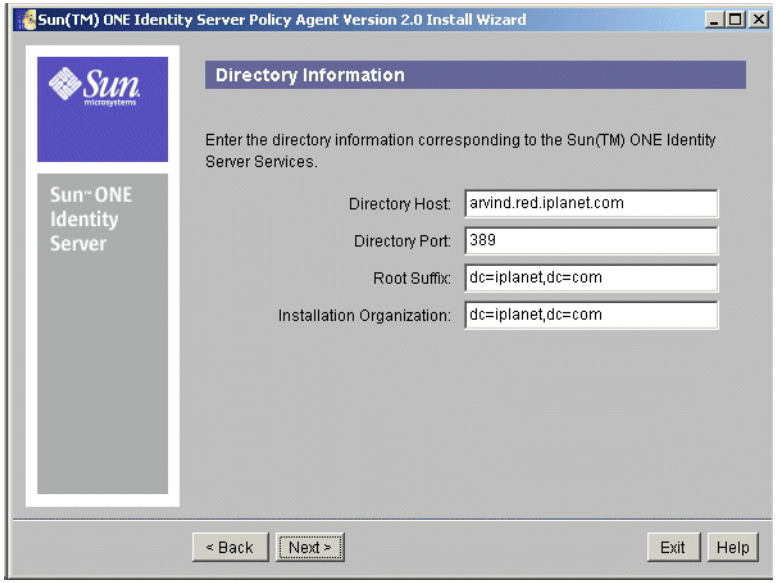
パスワードを変更するには、次の方法でこのツールを呼び出します。

#./agentadmin -password *oldpassword newpassword*

---

5. 「Directory Server Information」画面では、Sun ONE Identity Server サービスと関連付けられたディレクトリサーバに関する次の情報を入力します。

図 7-3 「Directory Information」画面



Directory Host: ディレクトリサーバをインストールしたシステムの完全指定ドメイン名を入力します。

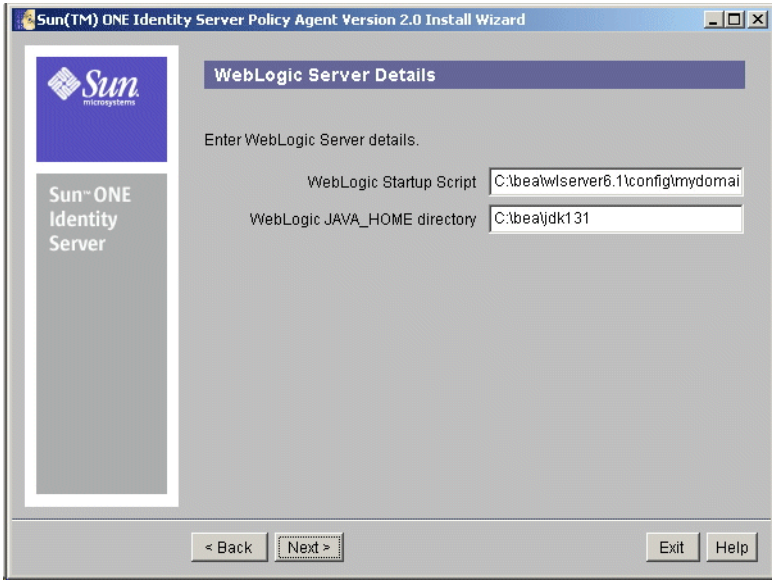
Directory Port: ディレクトリサーバが使うポートの番号を入力します。

Root Suffix: ディレクトリサーバに適用されるルートの接尾辞を入力します。

Installation Organization: Sun ONE Identity Server のインストール時に指定した組織名を入力します。

6. 「WebLogic Server Details」画面では、エージェントをインストールする WebLogic Server に関する次の情報を入力します。

図 7-4　「WebLogic Server Details」画面



WebLogic Startup Script: WebLogic Server の起動に使うスクリプトの場所を完全パスで指定します。Solaris プラットフォームの WebLogic Server 起動スクリプトは、WebLogic Server の起動に使うシェルスクリプトです。Windows プラットフォームでは、CMD スクリプトが使用されます。このスクリプトは、次のディレクトリにあります。

/WebLogic_Install_Dir/bea/wlserver6.1/config/*server-domain-name*/

WebLogic JAVA_HOME directory: WebLogic Server が使う JDK をインストールするホームディレクトリを完全パスで指定します。WebLogic JAVA_HOME ディレクトリは、WebLogic Server が使う JDK のインストールを参照します。通常は、次のディレクトリを示す完全パスが使われます。

/WebLogic_Install_Dir/bea/jdk131

このディレクトリの場所がわからない場合は、WebLogic Server の起動に使われる WebLogic 起動スクリプトを開き、このファイルに指定されている JAVA_HOME 変数の値を確認します。

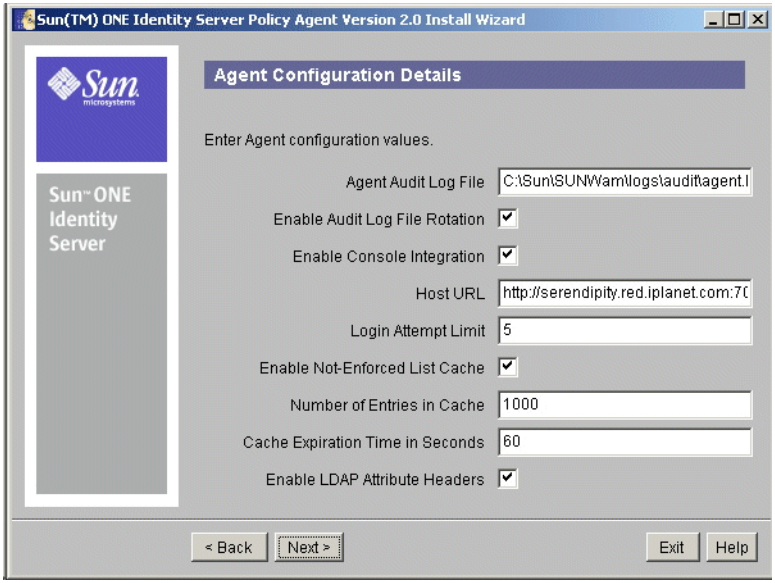
警告

- インストールプログラムは WebLogic Server の起動スクリプトを変更し、WebLogic CLASSPATH に特定のライブラリを指定します。また、Java 仮想マシン上の WebLogic Server の起動クラスをロードするコマンドに必要なパラメータを追加します。WebLogic Server の起動スクリプトに誤った値を指定すると、必要なクラスとパラメータが追加されず、エージェントが正しく動作しないため、WebLogic Server を利用できなくなることがあります。このような問題を避けるため、WebLogic Server の起動スクリプトには正確な値を指定してください。詳細は、「付録 A」を参照してください。
- インストールプログラムは、WebLogic Server が使用する JDK に拡張機能を追加します。これは、エージェントの実行に必要な機能です。この拡張機能は、前述の WebLogic JAVA_HOME の値が示すディレクトリにインストールされます。値が誤っていたり、WebLogic Server が使わない JDK が指定されている場合は、エージェントが正しく動作しないため、WebLogic Server を利用できなくなることがあります。このような問題を避けるため、WebLogic JAVA_HOME には正確な値を指定してください。

7. 「Agent Configuration Details」画面では、エージェントが正しく機能するために必要な設定情報を入力します。

注 この手順を実行する前に、「重要なヒント」をお読みください。

図 7-5 「Agent Configuration Details」画面



Audit Log File: エージェントが監査メッセージの記録に使うログファイルの完全パスを入力します。

Enable Audit log file rotation: 監査ログファイルのローテーションを有効にする場合に選択します。

Enable Console Integration: Sun ONE Identity Server と WebLogic Server の管理コンソールをコンソールレベルで統合する場合に選択します。

Host URL: 必要に応じてユーザをリダイレクトするためにエージェントがベース URL として使う有効な URL を入力します。この値を空白のまま残すことはできません。エージェントが有効なサーバの完全指定のドメイン名を入力する必要があります。たとえば、`http://www.mycompany.com:80/` という URL でアクセスできる WebLogic Server にエージェントがインストールされている場合は、「Host URL」の値も `http://www.mycompany.com:80/` となります。

Login Attempt Limit: ユーザが特定の URL へのアクセスに連続して失敗した場合、セキュリティ上の目的から何回続けて失敗したらそのユーザのアクセスを一時的に拒否するかを指定します。この機能を無効にするには、`0` を指定します。

Enable Not-Enforced List Cache: 不適用リストの評価結果をキャッシュする場合に選択します。

Number of Entries in Cache: 特定のインスタンスについて、キャッシュに保持できるエントリ数を指定します。

Cache Expiration Time: 不適用リストのキャッシュに追加されたエントリがキャッシュ内に存在できる最大時間を秒単位で指定します。

Enable LDAP Attribute Headers: 現在のユーザに関連する LDAP 属性を HTTP ヘッダーに渡す場合に選択します。

重要なヒント

- 監査ログファイルは、エージェントに不可欠なものです。システムに存在しないファイルの名前を監査ファイルとして指定することもできます。この場合、エージェントは最初の使用時にこのファイルと必要なディレクトリを作成します。反対に、エージェントが利用できる既存のファイルの名前を指定することもできます。ただし、エージェントは WebLogic Server と同じプロセスで実行されるので、WebLogic Server プロセスには適切な書き込みアクセス権が設定されている必要があります。この設定に誤りがあると、エージェントが正しく動作しないため、WebLogic Server を利用できなくなることがあります。

- コンソールを統合すると、Identity Server 側のユーザ、ロール、およびロールに関連づけられているユーザに関する情報を WebLogic Server の管理コンソール側でも確認できるようになります。ユーザとロールに関する情報が WebLogic Server のコンソールにアクセスできる管理者にも開示されることになるので、この機能を有効にするときは注意が必要です。

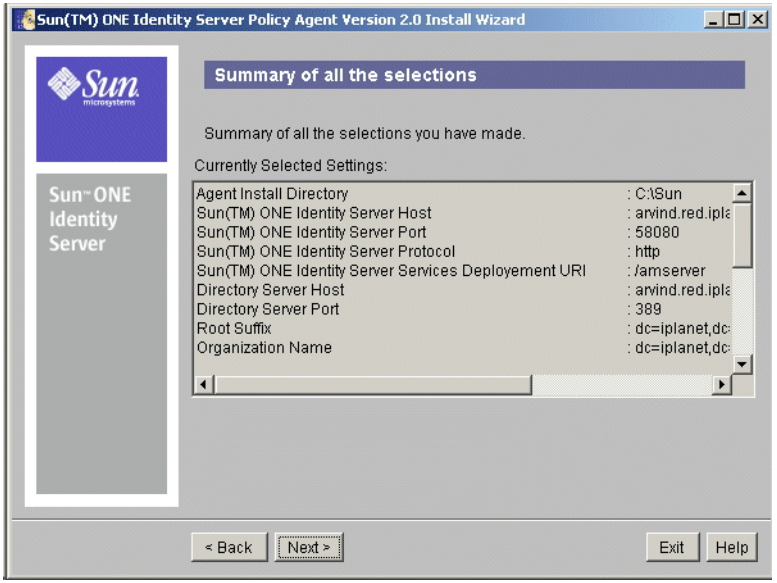
- 適切な証明情報のない要求を受け取ると、エージェントはそのユーザを Sun ONE Identity Server の認証サービスにリダイレクトします。このリダイレクトと同時に、エージェントは元の要求に関する情報も認証サービスに渡します。この情報は、ユーザを元の要求先に再リダイレクトするときに使用されます。この要求には、ユーザがアクセスしようとしていた Web コンテナまたは Web サーバを識別するホスト URL が含まれます。このホスト URL は、この画面の「Host URL」の値として設定できます。ホスト URL は、エージェントが必要に応じてデフォルトの FQDN を指定する場合にも使用されます。たとえば、ユーザが `http://mycompany/SomeApp/SomeModule` という URL を入力し、ホスト URL に `http://www.mycompany.com:80/` が指定されている場合は、エージェントは処理を進める前に、ユーザを `http://www.mycompany.com:80/SomeApp/SomeModule` にリダイレクトします。これにより、エージェントがユーザの特定に使用するドメイン固有の SSO Cookie を確実に利用できるようになります。ホスト URL の設定は、ユーザが最初にアクセスを試みる Web コンテナまたは Web サーバに関係なく、認証に成功した後にユーザを特定のホスト URL にリダイレクトするためにも利用されています。この設定はデフォルトの動作に優先して適用されますが、誤った値を指定した場合はアプリケーションにアクセスできなくなります。このため、別の値への変更が必要となる配備の場合以外は、インストールプログラムが設定するデフォルト値のまま残しておくことをお勧めします。

- インストールプログラムの画面でホスト URL を指定する場合は、画面に表示されるプロトコル、完全指定ホスト名、ポートが有効で、実際の配備に適していることを確認してください。たとえば、エージェントを使って SSL モードの WebLogic Server を保護するには、ホスト URL のプロトコルが「https」に設定されている必要があります。

- Login Attempt Limit 機能は、エンドユーザが認証要求を繰り返すことでアプリケーションサーバをオーバロード状態にするサービス拒否の攻撃から、ホストしているアプリケーションを守るために使われます。この機能を無効にすると、このような攻撃に対してはシステムのセキュリティは弱くなります。このため、この機能を無効にする具体的な必要性がない限り、この機能を有効にしておく必要があります。
- 設定されている不適用パターンルールのリストが非常に長い場合、エージェントは要求を認証なしに許可するかどうかについて、すべての要求に対してすべてのルールを評価する必要が生じます。ユーザ負荷が大きくなると、ルールの評価に要する時間も長くなり、システム全体のパフォーマンスも低下します。このような問題を避けるために、不適用リストのキャッシュを有効にすることをお勧めします。
- 不適用リストのキャッシュを有効にしても、キャッシュに残すエントリ数とキャッシュエントリの有効期限が適切に設定されていない場合はパフォーマンスが低下します。キャッシュの有効期限が必要以上に長い場合は、キャッシュがすぐにいっぱいになり、新しい要求は指定されるすべてのパターンルールと照合されるため、システムのパフォーマンス改善には結びつきません。キャッシュに残すエントリ数にかなり大きな値を設定すると、システムメモリの消費が大きくなり、パフォーマンスが低下します。これらの値を設定するときは事前に配備環境を慎重に検討し、システムの利用状況に応じて値を変更する必要があります。制御された環境で 2 つのパラメータにさまざまな値を設定してシステムをテストし、最適な値を見つけてから運用環境に配備することをお勧めします。
- エージェントは、メモリに 2 種類のキャッシュを保持します。1 つには適用と評価された URL が記録され、もう 1 つには不適用として評価された URL が記録されます。キャッシュに残すエントリ数とキャッシュの有効期限に指定した値は、どちらのキャッシュにも適用されます。キャッシュのサイズと有効期限を設定する場合は、この点に注意する必要があります。
- LDAP 属性をヘッダーに渡す機能を有効にすると、エージェントは、すべての要求について認証されたユーザに関連する LDAP 属性を検索し、それを要求のヘッダー情報に追加する必要があります。この機能は、配備したアプリケーションがビジネスロジックを実装する上で、これらのヘッダー情報の値を必要とする場合にだけ有効にしてください。特に必要がない場合にこの機能を有効にすると、システムのパフォーマンスが低下し、目的の機能を果たせなくなります。
- インストール時に設定したパラメータは、`AMAgent.properties` ファイルを編集することで後から変更できます。「エージェントの設定」を参照してください。

8. 「Summary of all the selections」画面では、エージェントに設定したインストールオプションを確認できます。変更が必要な場合は、「Back」をクリックします。変更の必要がない場合は、「Next」をクリックして処理を続行します。

図 7-6　「Summary of all the selections」画面



9. 「Ready to Install」画面では、「Install Now」ボタンをクリックしてインストールを開始します。

10. インストールプログラムがシステムに変更を加え始めると、進捗状況が「Install Progress」画面に表示されます。必要に応じて「Stop」ボタンをクリックすることで、このプロセスを中断できます。

注　インストールを中断すると製品が部分的にインストールされ、アンインストールできなくなることがあり、WebLogic Server を利用できなくなることがあります。このプロセスを中断しないことを強くお勧めします。どうしても必要な場合にだけプロセスを中断してください。

---

11. 「Installation Summary」画面で、「Details」をクリックすると、インストール中に処理された設定情報の詳細を参照できます。「Exit」をクリックしてプログラムを終了します。

注　インストールの状態が「Failed」と表示されるときは、「Details」ボタンをクリックしてインストールログファイルの内容を確認し、問題のあったタスクを特定します。この場合、エージェントをアンインストールし、失敗の原因を解決した上で再度インストールを実行できます。

インストールが完了したら、WebLogic Server と配備されるアプリケーションを適切に設定します。詳細については、次の節で説明します。

WebLogic Server の設定

WebLogic Server 6.1 SP2 用の Sun ONE Identity Server ポリシーエージェントをシステムにインストールしたら、エージェントの一部として提供されるエージェントレルムを使って WebLogic Server を設定します。

エージェントレルムのインストール

エージェントレルムは、WebLogic Server 管理コンソールを使って WebLogic Server に追加されるカスタムセキュリティレルムです。ここでは、エージェントレルムを WebLogic Server に追加する手順について説明します。ここに示す情報は、エージェントレルムのインストールだけを目的としており、WebLogic Server のマニュアルに記載されている情報に代わるものではありません。WebLogic カスタムレルムの詳細については、次の Web サイトにある WebLogic Server のマニュアルを参照してください。`http://www.bea.com`

エージェントレルムをインストールするには、次の手順を実行する必要があります。

1. エージェントのカスタムレルムの作成
2. エージェントレルムのキャッシングレルムの作成
3. ファイルレルムの設定

エージェントのカスタムレルムの作成

WebLogic Server にエージェントレルムをインストールするために、カスタムレルムを作成するには、次の手順に従います。

1. WebLogic Server の管理コンソールにログオンします。コンソールにログオンするには、設定されているシステムユーザ名とパスワードを入力します。
2. 管理コンソールの左ペインで、「+」記号をクリックして「Security」ノードを展開します。
3. 左ペインの「Security」ノードの下に表示される「Realms」をクリックします。システムで利用できるレルムのリストが右ペインに表示されます。
4. 右ペインで、「Configure a new Custom Realm」というリンクをクリックします。新規作成するカスタムレルムに関する情報を入力するための書式が表示されます。

5. この書式に次の情報を入力し、「Create」をクリックします。

   Name: Agent Realm

   Realm Class Name: `com.iplanet.amagent.weblogic.realm.AgentRealm`

6. 新しいレルムを作成したら、WebLogic Server を再起動します。

WebLogic Server を再起動したら、管理コンソールから「Security」> 「Realms」ノードに移動します。右ペインのレルムのリストには、新たに作成したエージェントレルムが表示されます。

エージェントレルムのキャッシングレルムの作成

WebLogic Server にエージェントレルムをインストールするために、キャッシングレルムを作成するには、次の手順に従います。

1. WebLogic Server の管理コンソールにログオンします。コンソールにログオンするには、設定されているシステムユーザ名とパスワードを入力します。
2. 管理コンソールの左ペインで、「+」記号をクリックして「Security」ノードを展開します。
3. 左ペインの「Security」ノードの下に表示される「Caching Realms」をクリックします。システムで利用できるキャッシングレルムのリストが右ペインに表示されます。
4. 右ペインで、「Configure a new Caching Realm」というリンクをクリックします。新規作成するキャッシングレルムに関する情報を入力するための書式が表示されます。
5. この書式に次の情報を入力します。

   Name: Agent Caching Realm

   Basic Realm: プルダウンメニューからエージェントレルムを選択します

6. 「Create」ボタンをクリックします。右ペインの表示が更新され、新しいキャッシングレルムが作成されます。右ペインには、新たに作成したキャッシングレルムの設定が表示されます。
7. 右ペインで、すべてのキャッシング属性を無効にします。次の手順を行います。

   c. 「ACL」タブをクリックします。ACL キャッシング属性が表示されます。

   d. 「Enable ACL Cache」の隣のチェックボックスからチェックマークを外します。

   e. 「Apply」ボタンをクリックします。

   f. 「Authentication」、「Groups」、「Users」、「Permissions」タブでもこの手順を繰り返します。「Enable Cache」のオプションからチェックマークを外し、「Apply」ボタンをクリックします。

8. WebLogic Server を再起動します。

WebLogic Server を再起動したら、管理コンソールから「Security」>「Caching Realms」ノードに移動します。右ペインのキャッシングレルムのリストには、新たに作成したエージェントキャッシングレルムが表示されます。

ファイルレルムの設定

エージェントキャッシングレルムを作成したら、WebLogic Server がこのキャッシングレルムを使うように設定します。この処理は、ファイルレルムを設定することで行います。次の手順は、ファイルレルムの設定方法を示しています。

1. WebLogic Server の管理コンソールにログオンします。
2. 左ペインの「Security」ノードをクリックします。これにより、コンソールの右ペインには WebLogic Server のセキュリティ設定が表示されます。
3. 右ペインの「Filerealm」タブをクリックします。現在のファイルレルムの詳細がコンソールに表示されます。
4. 右ペインに表示される書式で、「Caching Realm」の下のプルダウンメニューから「Agent Caching Realm」を選択します。
5. 「Apply」ボタンをクリックします。
6. WebLogic Server を再起動します。

ファイルレルムが設定され WebLogic Server が再起動すると、エージェントレルムのインストールは完了です。

注 エージェントレルムを正しく設定したら、WebLogic Server の `config.xml` ファイルをバックアップしておくことをお勧めします。このファイルに `config.xml.withAgent` という名前をつけておき、次にエージェントをインストールする場合、既存の `config.xml` ファイルにこのファイルをコピーするだけで、エージェントレルムのインストールに必要な手順を省略できます。

インストールのトラブルシューティング

ファイルレルムを設定した後で、WebLogic Server が正しく起動しなくなった場合は、次の理由が考えられます。

- **エージェントのインストール時に WebLogic Server の起動ファイルが正しく変更されなかった**

  この問題は、エージェントのインストール前に WebLogic Server の起動ファイルを修正した場合、またはこのファイルの変更に必要な権限がインストールプログラムに設定されていなかった場合に生じます。どちらの場合も、付録 A に記載されている情報を参照して起動ファイルを手動で変更することで問題を解決できます。

- **エージェントのインストールプログラムが、WebLogic Server の使用に必要な JDK の拡張機能をインストールできなかった**

  これを確認するには、まず、WebLogic Server の起動ファイルを確認し、WebLogic Server が使う JDK の場所を特定します。これは、WebLogic Server 起動ファイルの `JAVA_HOME` の値から確認できます。この値が示すディレクトリに必要な拡張機能がインストールされているかどうかを調べます。必要に応じて、付録 A に記載されている手順に従って拡張機能を手動でインストールします。

WebLogic Server が正しく起動しない理由が上記の原因では説明できない場合は、重大なインストールエラーが発生した可能性があります。この問題を解決するには、システムからエージェントをアンインストールします。これにより、WebLogic Server の状態はエージェントをインストールする前の状態に戻ります。

アプリケーションの設定

WebLogic Server 6.1 SP2 用 Sun ONE Identity Server ポリシーエージェントのエージェントレルムコンポーネントは、Sun ONE Identity Server の各種主体との実行時マッピングを提供します。ホストしているアプリケーションでは、現在認証されているユーザが特定リソースへのアクセスを認証されているかどうか、または特定のロールのメンバーに含まれているかどうかを決定するために、セキュリティロールの抽象的な名前が使われます。この実行時評価は、Identity Server の認証サービスによってユーザが Identity Server 主体として認証された場合にだけ行われます。ユーザが適切に認証されない場合は、エージェントレルムが行うこの評価の結果はすべてネガティブなものとなり、要求するリソースに対するユーザのアクセスは拒否されます。

エージェントフィルタは、特定のアプリケーションリソースにアクセスしようとするユーザに認証を適用し、必要に応じて主体とのマッピングを正しく評価するエージェントレルムコンポーネントを有効にします。

WebLogic Server のコアとしてインストールされるエージェントレルムとは異なり、エージェントフィルタは Identity Server によって保護される配備済みアプリケーションにインストールされます。WebLogic Server では、すべてのアプリケーションがエージェントによる保護を必要とします。エージェントによって保護されていないアプリケーションを、エージェントレルムがインストールされている WebLogic Server に配備しないことをお勧めします。これは、このようなアプリケーションが独自のセキュリティ要件を個別に適用することを防ぐためです。このようなアプリケーションが実行するセキュリティ評価がエージェントレルムに影響するため、動作不良の原因となることがあります。

アプリケーションへのエージェントフィルタコンポーネントのインストール

エージェントフィルタは、保護が必要なアプリケーションの配備記述子を変更するだけでインストールできます。次の手順は、特定のアプリケーションにエージェントフィルタコンポーネントをインストールする方法を説明しています。

1. アプリケーションが WebLogic Server に配備されている場合は、WebLogic Server の管理コンソールまたは WebLogic Server の配備ツールを使って削除する必要があります。
2. このアプリケーションにエージェントフィルタをインストールするときに内容が変更されるので、配備記述子のバックアップを作成しておくことをお勧めします。
3. アプリケーションの `web.xml` 配備記述子を編集します。フィルタはサーブレットの仕様 2.3 で取り上げられた概念であるため、配備記述子がサーブレット 2.3 互換の配備記述子を認識できるように、`web.xml` の `DOCTYPE` 要素を変更する必要があります。`DOCTYPE` 要素を次のように変更します。

```
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web
Application 2.3//EN" "http://java.sun.com/dtd/web-app_2_3.dtd">
```

4. DOCTYPE 要素を変更したら、配備記述子にフィルタ要素を追加します。web.xml 配備記述子の web-app 要素の description 要素の直後にフィルタ要素 (filter) とフィルタマッピング要素 (filter-mapping) を追加します。次に、フィルタ要素とフィルタマッピング要素を追加した web.xml の例を示します。

```
<web-app>
  <display-name>...</display-name>
  <description>...</description>

  <filter>
  <filter-name>Agent</filter-name>
  <display-name>Agent</display-name>
  <description>SunTM ONE Idenitity Server Policy Agent for
WebLogic 6.1 SP2</description>
  <filter-class>com.iplanet.amagent.weblogic.filter.AgentFilter<
/filter-class>
  </filter>
  <filter-mapping>
  <filter-name>Agent</filter-name>
  <url-pattern>/*</url-pattern>
  </filter-mapping>
  ...
  ...
</web-app>
```

5. web.xml 配備記述子を変更して、新しい DOCTYPE 要素とフィルタ要素を反映させると、エージェントフィルタはアプリケーションに追加されます。

ロールと主体のマッピングの作成

アプリケーションを設定してエージェントフィルタをインストールすると、エージェントフィルタが認証を適用するようになり、ロールと主体のマッピングがエージェントレルムによって正しく解決されるようになります。ただし、ホストするアプリケーションが実行時に利用できるように、このマッピングを事前に作成しておく必要があります。

このマッピングの作成方法には、次の 2 つがあります。

- WebLogic Server 固有の配備記述子を編集する

  WebLogic Server 固有の配備記述子を編集して、ロールと主体のマッピングを作成できます。この配備記述子は、`weblogic.xml` ファイルと `weblogic-ejb-jar.xml` ファイルにあります。これらの記述子を編集してロールと主体のマッピングを作成する方法について詳細は、WebLogic Server のマニュアルを参照してください。このようなマッピングを作成する記述子の例については、付録 C を参照してください。

- WebLogic Server の管理コンソールを使用する

  WebLogic Server の管理コンソールを使って、配備済みアプリケーションの配備記述子を編集し、ロールと主体のマッピングを作成することができます。この機能を利用するには、アプリケーションが配備され、WebLogic Server が稼働している必要があります。配備したアプリケーションのマッピングを管理コンソールを使って作成する方法については、WebLogic Server のマニュアルを参照してください。

アプリケーション固有のエージェントの設定

配備したアプリケーションを、パブリックな部分と、各種アクセス制限によって保護される部分に区分することがよくあります。ほとんどの場合は、アプリケーションのパブリック部分にはあらゆるユーザがアクセスでき、保護された部分には登録ユーザだけがアクセスできます。Identity Server の認証サービスによる認証を必要とせずに、アプリケーションのパブリック部分に対してあらゆるユーザがアクセスできるようにエージェントを設定することができます。この情報は、次の場所に保存されているエージェントの設定プロパティファイルに記録されます。

Agent_Install_Dir`/wlAgent/amAgent/config/AMAgent.properties`

このファイルを編集して、エージェントの一般的な設定と、各アプリケーションに固有の設定を行うことができます。

注

- AMAgent.properties ファイルに設定されるプロパティは、エージェントが正しく機能する上で不可欠な情報です。このファイルに誤った値を設定すると、エージェントが正しく機能しなくなったり、アプリケーションにアクセスできなくなったり、またはシステム全体が利用不可能になったりすることがあります。このファイルの値を変更するときは細心の注意を払い、いつでも変更を破棄してシステムを元の状態に戻せるように、バックアップを作成しておくことをお勧めします。
- AMAgent.properties ファイルに設定したプロパティは、WebLogic Server の起動時にロードされます。WebLogic Server の実行中にこのファイルに加えた変更は、サーバを再起動するまで適用されません。

アプリケーション固有の不適用リストの設定

アプリケーションの一部に対してあらゆるユーザがアクセスできるようにするには、特定のアプリケーション専用のエントリを AMAgent.properties ファイルに追加する必要があります。このプロパティを設定するエントリはアプリケーション不適用リストと呼ばれ、次の文字列によって識別されます。

com.sun.am.policy.config.filter.*AppName*.notEnforcedList[*index*]=*pattern*

エージェントが実行時にこのプロパティを使うには、正しい書式で設定する必要があります。次のように、この文字列の中でイタリック体で表示されている項目は、適切な値に置き換える必要があります。

AppName: この文字列には、配備されたアプリケーションのコンテキストパスから最初のスラッシュ (/) を除いた部分を入力します。コンテキストパスは、ユーザがアクセスしようとしているアプリケーションの識別に使われる URI の最初の部分です。たとえば、ユーザが次の URL を使ってアプリケーションにアクセスしているとします。

http://myserver.mydomain.com/SomeApp/index.html、または

http://myserver.mydomain.com/SomeApp/SomeModule/doSometing.jsp

どちらの場合も、*AppName* は SomeApp となります。

index: これは配備するアプリケーションごとに割り当てられる 0 以上の整数で、アプリケーション不適用リスト内の他のエントリと同じ値にすることはできません。たとえば、コンテキストパスが /SomeApp のアプリケーションのアプリケーション不適用リストでは、次の 2 つのエントリが考えられます。

```
com.sun.am.policy.config.filter.SomeApp.notEnforcedList[0]=/SomeApp
/public/*
```

```
com.sun.am.policy.config.filter.SomeApp.notEnforcedList[1]=/SomeApp
/images/*
```

pattern: これは、認証を適用せずにアクセスを許可するかどうかを評価するときに、要求と照合されるパターン文字列です。パターン文字列には、`/SomeApp/public/RegistrationServlet` のような具体的な URI だけでなく、ワイルドカード文字「`*`」を使って、要求 URI 中の 0 文字以上の一致を表わす汎用パターンも指定できます。たとえば、`/SomeApp/public/*` を指定すると、`/SomeApp/public/` から始まるすべての URI と一致します。

このプロパティを使うことで、エージェントが不適用として扱うパターン文字列と URI を数多く設定できます。また、設定しないこともできます。言い換えれば、これらのパターンと一致するユーザ要求は、認証の適用なしにアクセスが許可されます。

アプリケーション固有のアクセス拒否 URI の設定

「Login Attempt Limit」オプションが有効な場合 (この機能の設定方法については「エージェントの設定」を参照)、特定の状況でエージェントはユーザからのアクセスをブロックします。この場合のエージェントのデフォルトの動作は、「HTTP Status Code 403 Forbidden」の送信です。このような状況では、Web コンテナは事前に設定されている「アクセス禁止」画面を表示するか、単に状態コードを返します。コードが返された場合は、ユーザのブラウザにはブラウザ独自のエラーメッセージが表示されます。これはエージェントのデフォルトの動作ですが、アクセスが拒否された場合に、指定した URI にあるアプリケーション固有のエラー画面が表示されるように変更することができます。

この処理は、`AMAgent.properties` ファイルの次のプロパティを設定して行います。

`com.sun.am.policy.config.filter.`*AppName*`.accessDeniedURI=/`*URI to use*

エージェントが実行時にこのプロパティを使うには、正しい書式で設定する必要があります。次のように、この文字列の中でイタリック体で表示されている項目は、適切な値に置き換える必要があります。

AppName: この文字列には、配備されたアプリケーションのコンテキストパスから最初のスラッシュ (/) を除いた部分を入力します。コンテキストパスは、ユーザがアクセスしようとしているアプリケーションの識別に使われる URI の最初の部分です。たとえば、ユーザが次の URL を使ってアプリケーションにアクセスしているとします。

`http://myserver.mydomain.com/SomeApp/index.html`、または

`http://myserver.mydomain.com/SomeApp/SomeModule/doSometing.jsp`

どちらの場合も、*AppName* は `SomeApp` となります。

URI to use: これは、ユーザからの要求をブロックしたときに表示するページの場所を示す、アプリケーション固有の URI です。この URI には、スタティックな HTML ページ、JSP、またはサーブレットを指定できます。ただし、この URI はアプリケーションの一部である必要があります。言い換えれば、この URI は次の文字列から始まる必要があります。

/AppName/rest of the URI

特殊なケース：デフォルトの Web アプリケーション

WebLogic Server のデフォルトの Web アプリケーションは、要求される URI にコンテキストパスが含まれていなくてもアクセスできます。たとえば、次のような URI でもデフォルトの Web アプリケーションにアクセスできます。

```
http://myserver.mydomain.com/index.html
```

この URL には関連するコンテキストパスが含まれていません。

このようなアプリケーション用に、エージェントには、エントリがデフォルトの Web アプリケーションに固有のものであるかどうかを識別する方法が用意されています。識別するには、次の手順を実行します。

1. 次のプロパティに、デフォルトの Web アプリケーションを示す名前を設定します。

   ```
   com.sun.am.policy.config.filter.defaultWebAppName= DefaultWebApp
   ```

2. 次に、この名前を使って、次に示すようにアプリケーション不適用リストとアプリケーションのアクセス拒否 URI を指定します。

   ```
   com.sun.am.policy.config.filter.DefaultWebApp.notEnforcedList[0]
   =/index.html

   com.sun.am.policy.config.filter.DefaultWebApp.notEnforcedList[1]
   =/about.html

   com.sun.am.policy.config.filter.DefaultWebApp.accessDeniedURI=/U
   RLAccessDenied.html
   ```

この方法により、関連するコンテキストパスを持たないデフォルトの Web アプリケーションも、コンテキストパスを持つ通常のアプリケーションと同様に設定することができます。また、デフォルトの Web アプリケーションにも、通常のアプリケーションと同様に不適用リストのエントリや、アクセス拒否 URI を指定できます。ただし、デフォルト Web アプリケーションのアクセス拒否 URI と不適用リストのエントリには、`/DefaultWebApp/` から始まるパスを設定できません。これは、このようなパスは現実にはアプリケーションサーバに存在しないためです。この場合、実際のコンテキストパスの設定は空白となるため、*AppName* を指定します。これは、デフォルトの Web アプリケーションのプロパティを設定できるように用意されているプロパティであり、値の指定には使用できません。

エージェントのグローバル設定

グローバル不適用リストを設定するには、AMAgent.properties ファイルを使います。この設定は、サーバに配備され、保護されているすべてのアプリケーションに適用されます。このリストは、次のプロパティを使って設定されます。

com.sun.am.policy.config.filter.global.notEnforcedList[*index*]= *pattern*

pattern には、具体的な URI、または要求 URI 中の 0 文字以上の一致を検索するように、ワイルドカード文字「*」を使ったパターンを指定できます。

index は 0 以上の整数値で、他のエントリの値と同じ値にすることはできません。

不適用リストの使用について

アプリケーションをパブリックドメインと保護ドメインに分割する上で、不適用リストを利用するととても便利です。しかし、適切に使用しない場合は悪影響が生じることがあります。

たとえば、サーブレットにアクセスするために必要な URI が不適用リストに設定されている一部のパターンと一致する場合、エージェントフィルタは、そのサーブレットにアクセスしようとしているユーザに認証を適用しません。しかし、ロールと主体のマッピングを使って、このサーブレットがエージェントによって保護されている Enterprise JavaBean コンポーネントにアクセスすることも考えられます。この場合、ユーザは認証されていないため、保護されているコンポーネントへのアクセスによって、アプリケーションサーバはセキュリティ違反の例外を生成します。このため、不適用リストにエントリを追加する前に、そのエントリがいかなる場合も保護されているリソースを含まないこと、および保護されているリソースにアクセスしないことを確認する必要があります。

不適用リストの使用でもう一つ特徴的なのは、画像の取り扱いです。一般に、Web ページにはボタン、プレースホルダ、バナー、ロゴなど、さまざまな目的で多数の画像が使われています。ユーザがこのページにアクセスするたびに、ブラウザはアプリケーションサーバに要求を発行し、そのページの画像を取得します。このような要求は、それぞれがクライアントからの個別の要求として扱われ、その他の要求と同様に、認証の評価メカニズムと不適用リストのチェックの対象となります。これにより、1 つのページを表示するために、1 つのクライアントからサーバに対して複数の呼び出しが行われます。このような要求に個別に認証を適用するオーバヘッドを考えると、システム全体に与える影響も大きなものとなります。この問題を解決するには、すべての画像と一致するグローバル不適用リストエントリを作成します。

次に例を示します。

com.sun.am.policy.config.filter.global.notEnforcedList[0]=*.gif

```
com.sun.am.policy.config.filter.global.notEnforcedList[1]=
/images/*
```

これは、.gif で終わるすべての要求 URI、および /images/ から始まるすべての URI に対して認証が適用されないことを意味します。ユーザ負荷が大きな環境では、こうすることでシステムのパフォーマンスを大幅に改善できます。

エージェントの設定

WebLogic Server 6.1 SP2 用 Sun ONE Identity Server ポリシーエージェントの中心的な設定は、次のディレクトリに保存されている AMAgent.properties ファイルを使って行われます。

Agent_Install_Dir/wlAgent/amAgent/config

このプロパティファイルには数多くの設定情報が含まれ、これを変更することで、配備環境に合わせてエージェントの動作をカスタマイズすることができます。

注 先に進む前に、このファイルおよびこのファイルに記録されている情報は、エージェントの動作に大きな影響を与えるという事実に注意してください。このファイルを変更するときは、常に事前にバックアップを作成しておくことを強くお勧めします。また、どうしても必要な場合以外はこのファイルを変更しないことも強くお勧めします。このファイルに無効なデータ項目があると、エージェントや配備したアプリケーションが正しく動作しなくなり、システム全体が利用できなくなることがあります。

このファイルで設定できる内容は、次のカテゴリに分類されます。

- 共通の設定
- 監査の設定
- レルムの設定
- グローバルフィルタの設定
- アプリケーションフィルタの設定
- デバッグエンジンの設定

次に、各カテゴリについて設定の詳細を説明します。

共通の設定

このカテゴリの設定は、エージェントの動作全体に適用される一般設定です。

組織名

キー：com.sun.am.policy.config.org

説明：このプロパティは、Sun ONE Identity Server から主体を検索するときに使われる組織名を指定します。

有効な値：Sun ONE Identity Server で組織名を表す文字列です。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例：com.sun.am.policy.config.org=iplanet.com

ルートの接尾辞

キー：com.sun.am.policy.config.rootsuffix

説明：このプロパティは、Sun ONE Identity Server から主体を検索するときに使われるルートの接尾辞を指定します。

有効な値：Sun ONE Identity Server でルートの接尾辞を表す文字列です。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例：com.sun.am.policy.config.rootsuffix=o=isp

ピープルコンテナレベル

キー：com.sun.am.policy.config.realm.peopleContainerLevel

説明：このプロパティは、Sun ONE Identity Server から主体を検索するときに使われるピープルコンテナレベルを指定します。

有効な値：Sun ONE Identity Server でピープルコンテナレベルを表す 0 以外の記号なし整数です。この値は、主体の検索に使われます。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例：com.sun.am.policy.config.realm.peopleContainerLevel=1

監査の設定

このカテゴリの設定は、エージェントが使用する監査エンジンの設定だけに適用されます。

言語コード

キー：com.sun.am.policy.config.audit.localeLanguageCode

説明：このプロパティは、監査ログメッセージのロケールを指定します。

有効な値：localeLanguageCode には、有効な ISO 言語コードを設定する必要があります。このプロパティのデフォルト値は en です。

注	詳細については、次の Web サイトで ISO 639 仕様を参照してください。 http://www.ics.uci.edu/pub/ietf/http/related/iso639.txt

例：com.sun.am.policy.config.audit.localeLanguageCode=en

国コード

キー：com.sun.am.policy.config.audit.localeCountryCode

説明：このプロパティは、監査ログメッセージのロケールを指定します。

有効な値：localeCountryCode には、有効な ISO 国別コードを設定する必要があります。このプロパティのデフォルト値は US です。

注	詳細については、次の Web サイトで ISO 3166 仕様を参照してください。 http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html

例：com.sun.am.policy.config.audit.localeCountryCode=US

監査ログファイル

キー：com.sun.am.policy.config.audit.logfile.name

説明：このプロパティは、監査メッセージの記録に使われる監査ログファイルを指定します。

有効な値：エージェントが監査メッセージの記録に使うログファイルの完全パスです。

注

- このファイルに対する適切な書き込みアクセス権が WebLogic Server のプロセスに設定されていることを確認してください。
- このプロパティに誤った値を設定すると、システムが正しく起動しないことがあります。

例 : `com.sun.am.policy.config.audit.logfile.name=/audit/agent.log`

監査ログファイルのローテーションフラグ

キー : `com.sun.am.policy.config.audit.logfile.rotate`

説明 : このプロパティは、エージェントが監査ログファイルをローテーションするかどうかを指定します。

有効な値 : `true` または `false`。このプロパティのデフォルト値は `false` です。必要に応じて変更してください。

例 : `com.sun.am.policy.config.audit.logfile.rotate=false`

監査ログファイルローテーションサイズ

キー : `com.sun.am.policy.config.audit.logfile.rotate.size`

説明 : このプロパティは、監査ログファイルのおよそのサイズをバイト数で指定します。ログファイルのローテーションの必要性を評価するときに使われます。

有効な値 : ローテーションが必要なログファイルのサイズをバイト数で表す 0 以外の符号なし整数値です。このプロパティのデフォルト値は `52428800` バイト (50M バイト) です。必要に応じて変更してください。

例 : `com.sun.am.policy.config.audit.logfile.rotate.size=52428800`

注

監査ログファイルのローテーションフラグが `false` に設定されている場合は、このプロパティは使われません。

レルムの設定

このカテゴリの設定は、エージェントレルムコンポーネントの設定に使われます。

コンソールの統合を許可するフラグ

キー：`com.sun.am.policy.config.weblogic.allowConsoleIntegration`

説明：このプロパティは、Sun ONE Identity Server と WebLogic Server のコンソールレベルの統合をエージェントレルムが許可するかどうかを指定します。

有効な値：`true` または `false`

注

- 有効に設定すると、コンソールレベルの統合により WebLogic Server の管理者が Sun ONE Identity Server の主体を WebLogic Server 側のコンソールで確認できるようになります。
- 有効に設定すると、WebLogic Server の管理者が Sun ONE Identity Server の主体を WebLogic Server 側のコンソールで確認できるようになるので、注意が必要です。
- このプロパティのデフォルト値は `false` です。必要に応じて変更してください。

例：`com.sun.am.policy.config.weblogic.allowConsoleIntegration=true`

SSO キャッシュのクリーンアップサイズ

キー：`com.sun.am.policy.config.realm.ssoCacheCleanupSize`

説明：このプロパティは、エージェントレルムが SSO キャッシュに維持できるエントリの最大数を指定します。この値を超えるとクリーンアップが開始されます。

有効な値：任意の正の整数です。エージェントレルムの SSO キャッシュに保持されるエントリ数を表します。この値を超えると、システムのメモリ消費を最小限に抑え、メモリを適切に使用するようにクリーンアップが開始されます。

注

- 有効ではあるが、配備環境に適さない値を設定すると、システムのパフォーマンスが低下します。また、システムの利用可能な SSO トークンが少なくなることもあります。
- システムのパフォーマンスに最適な値は、配備するアプリケーションの種類、アプリケーションのピーク時のアクティブユーザセッション数、ユーザセッションの平均時間によって異なります。
- このプロパティの最適値を決定するには、アプリケーションを運用環境に配備する前に、制御されたテスト環境で運用テストする必要があります。
- このプロパティのデフォルト値は 1000 です。

例 : `com.sun.am.policy.config.realm.ssoCacheCleanupSize = 1000`

SSO キャッシュクリーンアップのロック時間

キー : `com.sun.am.policy.config.realm.ssoCacheCleanupLockTime`

説明 : このプロパティは、直前のクリーンアッププロセスでメモリをまったく解放できなかった場合に、エージェントレルムが SSO キャッシュの次回のクリーンアッププロセスを開始するまでの待ち時間を秒単位で指定します。

有効な値 : 任意の正の整数値です。直前のクリーンアッププロセスによって適切な量のメモリを解放できなかった場合に、次回のクリーンアッププロセスを開始するまでの秒数を示します。

注

- 有効ではあるが、配備環境に適さない値を設定すると、システムのパフォーマンスが低下することがあります。
- システムのパフォーマンスに最適な値は、配備するアプリケーションの種類、アプリケーションを使用する平均的なユーザの代表的なセッション時間によって異なります。
- このプロパティの最適値を決定するには、アプリケーションを運用環境に配備する前に、制御されたテスト環境で運用テストする必要があります。
- このプロパティのデフォルト値は 1200 です。

例 : `com.sun.am.policy.config.realm.ssoCacheCleanupLockTime = 1200`

SSO キャッシュクリーンの制限サイズ

キー : `com.sun.am.policy.config.realm.ssoCacheCleanupBoundSize`

説明 : このプロパティは、クリーンアッププロセスの実行時に確認される、エージェントレルム SSO キャッシュの最大エントリ数を指定します。このプロパティに適切な値を設定すると、キャッシュクリーンアップの実行時にシステムの応答時間が全体的に向上します。

有効な値 : 任意の正の整数値です。クリーンアップの実行時にクリーンアッププロセスが調べる SSO キャッシュ内のエントリ数を表します。この値は、`com.sun.am.policy.config.realm.ssoCacheCleanupSize` プロパティの値に関係なく設定できます。

注

- 有効ではあるが、配備環境に適さない値を設定すると、システムのパフォーマンスが低下することがあります。
- システムのパフォーマンスに最適な値は、配備するアプリケーションの種類、アプリケーションを使用する平均的なユーザの代表的なセッション時間によって異なります。
- このプロパティの最適値を決定するには、アプリケーションを運用環境に配備する前に、制御されたテスト環境で運用テストする必要があります。
- このプロパティのデフォルト値は 50 です。

例 :

```
com.sun.am.policy.config.realm.ssoCacheCleanupBoundSize = 50
```

グローバルフィルタの設定

このカテゴリの設定は、エージェントフィルタコンポーネントの設定に使われます。

SSO トークン名

キー : `com.sun.am.policy.config.filter.ssoTokenName`

説明 : このプロパティは、SSO トークンを表す Cookie の名前を指定します。

有効な値 : Sun ONE Identity Server の認証サービスが発行する SSO トークン Cookie の名前を表す文字列です。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例 : `com.sun.am.policy.config.filter.ssoTokenName=iPlanetDirectoryPro`

FQDN マップ

キー :

`com.sun.am.policy.config.filter[`*invalid-name*`]`

説明 : FQDN マップは、ホスト名の一部だけを指定したり、IP アドレスを入力した場合など、ユーザが保護されたリソースにアクセスするときに誤った URL を入力した場合にエージェントが対応するための簡単なマップです。

有効な値 : 有効な値は、無効な FQDN 値を対応する有効な値にマッピングするこのプロパティの構文に準拠している必要があります。

このプロパティを指定する書式は次のとおりです。

`com.sun.am.policy.config.filter.fqdnMap[`*invalid-name*`]=`*valid-name*

invalid-name は、ユーザが入力すると考えられる無効な FQDN ホスト名、*valid-name* はフィルタがユーザをリダイレクトする FQDN ホスト名です。

注

- 無効な同じ FQDN 名に異なる値を設定しないでください。このように設定した場合、アプリケーションにアクセスできなくなることがあります。
- このプロパティに無効な値を設定すると、アプリケーションにアクセスできなくなることがあります。
- このプロパティを使って、複数のホスト名のマップを作成することができます。このサーバがホストするアプリケーションに対して、複数のホスト名を使ってアクセスできる場合がこれに該当します。ただし、アプリケーションにアクセスできなくなることもあるため、この機能の利用には注意が必要です。
- エージェントは、`com.sun.am.policy.config.filter.hostURL` プロパティに優先してこのプロパティの値を適用します。
- このプロパティを使うことで、特定のホスト名の URL に対してエージェントが対応手段を取らないように設定することができます。たとえば、raw IP アドレスを使ってアプリケーションリソースにアクセスするユーザに対しては、リダイレクトなどの対策を適用しない場合は、次のようなマップエントリを指定します。

  `com.sun.am.policy.config.filter[`*IP*`]=`*IP*

- 上記の要件を満たしている限り、設定できるプロパティの数 (マップの数) に制限はありません。

例 :

```
com.sun.am.policy.config.filter[myserver]=myserver.mydomain.com
```

com.sun.am.policy.config.filter[myserver.mydomain]=myserver.mydomain.com

com.sun.am.policy.config.filter[*IP*]=myserver.mydomain.com

com.sun.am.policy.config.filter[*invalid-name*]=*valid-name*

ログイン URL

キー：com.sun.am.policy.config.filter.loginURL

説明：このプロパティは、適切な証明情報を持たないユーザをエージェントが Sun ONE Identity Server の認証サービスにリダイレクトするときに使われるログイン URL を指定します。

有効な値：完全な URL を指定します。適切な証明情報を持たないユーザを Sun ONE Identity Server の認証サービスにリダイレクトする場合のリダイレクト先 URL として使われます。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例：
com.sun.am.policy.config.filter.loginURL=http://myserver.mydomain.com:58080/amserver/login

ホスト URL

キー：com.sun.am.policy.config.filter.hostURL

説明：このプロパティは、ブラウザから発行された要求をエージェントが再構築するときに使われるホスト URL を指定します。この値は、認証後に Sun ONE Identity Server の認証サービスがユーザを元の要求先にリダイレクトするために使われます。

有効な値：エージェントが介入できるように、ユーザのアクセス先となるホストの URL を指定します。この値は、このプロパティの書式に準拠する必要があります。このプロパティの値の書式は次のとおりです。

protocol://*hostname*.*optional-sub-domain*.*domain*:*port*

protocol には http または https を指定できます。

hostname.optional-sub-domain.domain には、エージェントが介入できるように、ユーザのアクセス先となるホストの名前を完全指定名で指定します。

port は、受信側 Web サーバが待機するポートの番号です。

指定しない場合は、エージェントは要求に含まれるホスト URL を使って再構築を試みます。

例：
com.sun.am.policy.config.filter.hostURL=http://www.iplanet.com:80

goto パラメータ

キー：com.sun.am.policy.config.filter.gotoParameter

説明：このプロパティは、エージェントがユーザを適切な認証サービスにリダイレクトするときに使う goto パラメータの値を指定します。このパラメータの値は、認証サービスがユーザを元の要求先にリダイレクトするために使われます。

有効な値：認証サービスが認識できる goto パラメータの名前を示す文字列です。

例：com.sun.am.policy.config.filter.gotoParameter=goto

ログイン試行制限

キー：com.sun.am.policy.config.filter.loginAttemptLimit

説明：このプロパティは、1 回のブラウザセッションでユーザが実行できるログイン試行回数を指定します。

有効な値：0 を含む符号なし整数値です。これは、保護されているリソースにアクセスしようとするユーザが、ログインを何回試みることができるかを表します。

注

- 値を 0 に設定すると、このオプションは無効になります。
- このプロパティのデフォルト値は 5 です。

例：com.sun.am.policy.config.filter.loginAttemptLimit=5

ログインカウンタの Cookie 名

キー：com.sun.am.policy.config.filter.loginCounterCookieName

説明：このプロパティは、ユーザが試行し、ログインに失敗した回数の追跡に使われる Cookie の名前を指定します。

有効な値：ユーザが試行し、ログインに失敗した回数を追跡するためにエージェントが発行する Cookie の名前を示す文字列です。このプロパティは、エージェントのインストール時に設定され、特別に必要な場合を除いて変更の必要はありません。

例：
com.sun.am.policy.config.filter.loginCounterCookieName=iPlanetLoginAttemptID

不適用リストキャッシュの有効化フラグ

キー：com.sun.am.policy.config.filter.notEnforcedList.cache

説明：このプロパティは、システムのパフォーマンスを向上させるために、適用または不適用と評価された要求 URI をキャッシュするかどうかを指定します。

有効な値：`true` または `false`。このプロパティのデフォルト値は `true` です。

例：`com.sun.am.policy.config.filter.notEnforcedList.cache=true`

不適用リストのキャッシュサイズ

キー：`com.sun.am.policy.config.filter.notEnforcedList.cacheSize`

説明：このプロパティは、エージェントが認証を適用しない URI のキャッシュと適用する URI のキャッシュに保持できるエントリの最大数を指定します。

有効な値：0 以外の符号なし整数値を指定します。実行時にキャッシュされる、認証が適用されない要求 URI と、認証が適用される要求 URI の数を表します。

注

- 有効ではあるが、配備環境に適さない値を設定すると、システムのパフォーマンスが低下することがあります。
- システムのパフォーマンスに最適な値は、配備するアプリケーションの種類、配備したアプリケーションで要求される URI の数、システムのユーザ負荷、キャッシュエントリに設定された有効期限、およびその配備に固有のその他の要因によって異なります。
- このプロパティの最適値を決定するには、アプリケーションを運用環境に配備する前に、制御されたテスト環境で運用テストする必要があります。

例：`com.sun.am.policy.config.filter.notEnforcedList.cacheSize=1000`

不適用リストキャッシュの有効期限

キー：`com.sun.am.policy.config.filter.notEnforcedList.cacheTime`

説明：このプロパティは、新しいキャッシュエントリ用にリソースを解放できるように、キャッシュされたエントリをキャッシュから削除するかどうかを評価する場合に使用される時間を秒単位で指定します。

有効な値：0 以外の符号なし整数値を指定します。クリーンアップの実行時に適用され、エントリがキャッシュ内に存在できる時間を秒単位で表します。

注

- 有効ではあるが、配備環境に適さない値を設定すると、システムのパフォーマンスが低下することがあります。
- システムのパフォーマンスに最適な値は、配備するアプリケーションの種類、配備したアプリケーションで要求される URI の数、システムのユーザ負荷、キャッシュエントリに設定された有効期限、およびその配備に固有のその他の要因によって異なります。
- このプロパティの最適値を決定するには、アプリケーションを運用環境に配備する前に、制御されたテスト環境で運用テストする必要があります。

例 : `com.sun.am.policy.config.filter.notEnforcedList.cacheTime=60`

LDAP 属性ヘッダーの有効化フラグ

キー : `com.sun.am.policy.config.filter.enableLDAPAttributeHeaders`

説明 : このプロパティは、現在の認証済みユーザに関連する LDAP 属性を使って、エージェントが `HttpServletRequest` を生成するかどうかを指定します。

有効な値 : `true` または `false`。このプロパティのデフォルト値は `false` です。必要に応じて変更してください。

例 : `com.sun.am.policy.config.filter.enableLDAPAttributeHeaders=true`

LDAP 属性のヘッダーマップ

キー : `com.sun.am.policy.config.filter.ldapAttribute[`*attr-name*`]`

説明 : このプロパティは、指定したヘッダー名で現在の認証済みユーザの LDAP 属性を生成することを指定します。

有効な値 : 有効な値は、このプロパティの構文に準拠している必要があります。指定する LDAP 属性は、有効な属性である必要があります。指定する HTTP ヘッダーの名前は、HTTP ヘッダーのネーミング規約に準拠している必要があります。このプロパティを指定する書式は次のとおりです。

`com.sun.am.policy.config.filter.ldapAttribute[`*attr-name*`]=`*header-name*

attr-name は、認証済みユーザを検索するための LDAP 属性の名前、*header-name* はこの値の格納に使われるヘッダーの名前です。

注

- 指定したヘッダー名が既存のヘッダー名と重複しないように注意してください。
- 上記の要件に適合している限り、設定できるプロパティの数に制限はありません。

例 :

```
com.sun.am.policy.config.filter.ldapAttribute[cn]=CUSTOM-Common-Name
com.sun.am.policy.config.filter.ldapAttribute[ou]=CUSTOM-Organizati
on-Unit
com.sun.am.policy.config.filter.ldapAttribute[o]=CUSTOM-Organization
com.sun.am.policy.config.filter.ldapAttribute[c]=CUSTOM-Country
com.sun.am.policy.config.filter.ldapAttribute[mail]=CUSTOM-Email

com.sun.am.policy.config.filter.ldapAttribute[employeenumber]=CUSTO
M-Employee-Number
```

LDAP 日時ヘッダー属性の書式文字列

キー : com.sun.am.policy.config.filter.ldapAttributeDateHeaderFormat

説明 : このプロパティは、属性ルックアップの結果として生成される日付と時刻の書式を指定します。ヘッダーの日時の値を返す javax.servlet.http.HttpServletReqeust インタフェースの特殊な get メソッドを使う場合は、これを指定する必要があります。

有効な値 : 有効な java.text.SimpleDateFormat Time Format Syntax 文字列です。詳細は、次の Web サイトを参照してください。
http://java.sun.com/j2se/1.3/docs/api/java/text/SimpleDateFormat.html

このプロパティのデフォルト値は、EEE, d MMM yyyy hh:mm:ss z です。必要に応じて変更してください。

注

このプロパティに無効な値を設定すると、アプリケーションが実行時例外をスローすることがあります。

例 :

```
com.sun.am.policy.config.filter.ldapAttributeDateHeaderFormat=
EEE, d MMM yyyy hh:mm:ss z
```

認証セッションのバインドフラグ

キー：com.sun.am.policy.config.filter.authSessionBinding

説明：このプロパティは、エージェントが認証にセッションバインドを適用するかどうかを指定します。

有効な値：true または false。このプロパティのデフォルト値は false です。

例：com.sun.am.policy.config.filter.authSessionBinding=false

SSO トークン URL のデコードフラグ

キー：com.sun.am.policy.config.filter.urlDecodeSSOToken

説明：このプロパティは、SSO トークンを使う前に、エージェントによる URL デコードが必要であるかどうかを指定します。

有効な値：true または false。このプロパティのデフォルト値は true です。

注	有効ではあるが、適切でない値をこのプロパティに設定すると、ユーザがアプリケーションにアクセスできなくなることがあります。

例：com.sun.am.policy.config.filter.urlDecodeSSOToken=true

デフォルト Web アプリケーション名

キー：com.sun.am.policy.config.filter.defaultWebAppName

説明：このプロパティは、アプリケーションサーバに配備されるデフォルトの Web アプリケーションの名前を指定します。

有効な値：デフォルトの Web アプリケーションの名前に使用できる、大文字と小文字のアルファベットで構成される文字列です。このプロパティのデフォルト値は DefaultWebApp です。

注	保護されたアプリケーションをデフォルトの Web アプリケーションとして配備するときは、このプロパティを設定する必要があります。

例：com.sun.am.policy.config.filter.defaultWebAppName=DefaultWebApp

グローバル不適用リスト

キー：com.sun.am.policy.config.filter.global.notEnforcedList[*index*]

説明：このプロパティは、要求された URI がエージェントによる認証を必要とするかどうかを評価するときに使われるパターンのリストを指定します。

有効な値：有効な値は、このプロパティの構文に準拠している必要があります。具体的な URI、またはワイルドカード文字「*」を使って、0 以上の文字と一致するパターンを指定できます。このプロパティの構文は次のとおりです。

```
com.sun.am.policy.config.filter.global.notEnforcedList[index]=pattern
```

index は、0 から始まり、このプロパティリストのエントリが増えるごとに 1 つずつ繰り上がる整数です。

pattern は、エージェントによる認証が適用されない要求 URI を表す文字列です。

pattern は、0 文字以上との一致を意味するワイルドカード文字「*」を含むことがあります。

index は、最初のエントリには 0、それ以降のエントリには順番に数字を割り当てる必要があります。index 値の割り当てに欠落があると、リストに含まれる一部のエントリ、またはすべてのエントリが認識されなくなります。

注　このプロパティにどのような値を設定しても、グローバル不適用リストが空として認識されることはありません。

例：

```
com.sun.am.policy.config.filter.global.notEnforcedList[0]=*.gif

com.sun.am.policy.config.filter.global.notEnforcedList[1]= public/*

com.sun.am.policy.config.filter.global.notEnforcedList[2]=
/images/*
```

アプリケーションフィルタの設定

このカテゴリの設定は、特定のアプリケーションのエージェントフィルタの設定に使われます。

アクセス拒否 URI

キー：com.sun.am.policy.config.filter.*AppName*.accessDeniedURI

説明：このプロパティは、保護されているアプリケーション用にアプリケーション固有アクセス拒否 URI を指定します。

有効な値：必要に応じて要求をブロックするアクセス拒否 URI として使われる、配備されたアプリケーション内の URI です。

注

- これは、保護されたアプリケーションに固有のプロパティです。このため、保護されたアプリケーションがシステムに複数配備されている場合は、アプリケーションごとに 1 つのプロパティが必要です。
- このプロパティには、アプリケーション内の URI を指定する必要があります。URI を指定しない場合は、実行時に内部サーバエラーとなることがあります。
- このプロパティを指定する書式は次のとおりです。

  com.sun.am.policy.config.filter.*AppName*.accessDeniedURI=*URI*

  AppName は配備されたアプリケーションのコンテキストパス名、*URI* は使用される URI です。*AppName* とアプリケーションのコンテキストパスの違いは、コンテキストパスは「/」から始まることです。
- 保護されたアプリケーションがデフォルトの Web アプリケーションである場合は、デフォルトの Web アプリケーション名プロパティの値として指定されているのと同じ文字列を *AppName* に設定する必要があります。
- 特定のアプリケーションにこのプロパティを指定しない場合は、エージェントは HTTP 状態コード 403 (アクセス禁止) を返すことで、アクセスがブロックされたことを示します。

例：

```
com.sun.am.policy.config.filter.Portal.accessDeniedURI=/Portal/AccessDenied.html

com.sun.am.policy.config.filter.BankApp.accessDeniedURI=/BankApp/Block.jsp
```

```
com.sun.am.policy.config.filter.DefaultWebApp.accessDeniedURI=/URLA
ccessDenied.htm
```

アプリケーションの不適用リスト

キー：com.sun.am.policy.config.filter.*AppName*.notEnforcedList[*index*]

説明：このプロパティは、特定のアプリケーションにおいて、要求された URI がエージェントによる認証を必要とするかどうかを評価するときに使われるパターンのリストを指定します。

有効な値：有効な値は、このプロパティの構文に準拠している必要があります。具体的な URI、またはワイルドカード文字「*」を使って、0 以上の文字と一致するパターンを指定できます。このプロパティの構文は次のとおりです。

com.sun.am.policy.config.filter.*AppName*.notEnforcedList[*index*]= *pattern*

AppName は、配備されたアプリケーションのコンテキストパス名から最初のスラッシュ (/) を除いた文字列です。

index は、0 から始まり、特定アプリケーションに指定されるプロパティが増えるごとに 1 つずつ繰り上がる整数です。

pattern は、エージェントによる認証が適用されない URI を表す文字列です。

注

- *pattern* は、0 文字以上との一致を意味するワイルドカード文字「*」を含むことがあります。
- *index* は、最初のエントリには 0、それ以降のエントリには順番に数字を割り当てる必要があります。index 値の割り当てに欠落があると、リストに含まれる一部のエントリ、またはすべてのエントリが認識されなくなります。index 値は、このプロパティリストに指定されるそれぞれの *AppName* で別々に割り当てられます。
- 保護されたアプリケーションがデフォルトの Web アプリケーションである場合は、デフォルトの Web アプリケーション名プロパティの値として指定されているのと同じ文字列を *AppName* に設定する必要があります。
- このプロパティにどのような値を設定しても、不適用リストが空として認識されることはありません。

例：

```
com.sun.am.policy.config.filter.Portal.notEnforcedList[0]=
/Portal/GuestPages/*

    com.sun.am.policy.config.filter.Portal.notEnforcedList[1]=
    /Portal/Registration/*
```

```
com.sun.am.policy.config.filter.Portal.notEnforcedList[2]=
/Portal/WebServices/PollServlet

com.sun.am.policy.config.filter.BankApp.notEnforcedList[0]=
/BankApp/ModuleGuestTour/*

com.sun.am.policy.config.filter.BankApp.notEnforcedList[1]=
/BankApp/index.html

com.sun.am.policy.config.filter.DefaultWebApp.notEnforcedList0]=
/index.html

com.sun.am.policy.config.filter.DefaultWebApp.notEnforcedList[1]
=/about.html
```

デバッグエンジンの設定

このカテゴリの設定は、診断情報を生成するデバッグエンジンの設定に使われます。

デバッグレベル

キー：com.sun.am.policy.config.debug.level

説明：このプロパティは、エージェントのデバッグエンジンが生成するデバッグメッセージの量を指定します。

有効な値：0、1、3、7、15、31 のいずれかです。これらの値の意味は次のとおりです。

0 = デバッグなし

1 = エラーメッセージのみ

3 = エラーメッセージと警告メッセージ

7 = エラーメッセージ、警告メッセージ、簡単な説明

15 = エラーメッセージ、警告メッセージ、詳細な説明

31 = エラーメッセージ、警告メッセージ、非常に詳細な説明

注

- システムのパフォーマンスを考慮する場合は、このプロパティには 0 を設定する必要があります。それ以外の値を設定した場合は、デバッグエンジンが生成する情報の量に応じてシステムのパフォーマンスに影響が出ます。
- 上記の有効な値リストに示した値以外の値を設定すると、デバッグエンジンの設定が無効となる場合があり、生成されるメッセージの量に影響します。

例 : com.sun.am.policy.config.debug.level=7

デバッグログファイル

キー : com.sun.am.policy.config.debug.logfile.name

説明 : このプロパティは、デバッグメッセージの記録に使われるデバッグログファイルを指定します。

有効な値 : エージェントがデバッグメッセージの記録に使うログファイルの完全パスです。

注

- このファイルに対する適切な書き込みアクセス権が WebLogic Server のプロセスに設定されていることを確認してください。
- このプロパティに誤ったパスを設定すると、デバッグメッセージがログファイルに記録されません。

例 :

```
com.sun.am.policy.config.debug.logfile.name=/debug/agent_debug.log
```

デバッグログファイルのローテーションフラグ

キー : com.sun.am.policy.config.debug.logfile.rotate

説明 : このプロパティは、エージェントがデバッグログファイルをローテーションするかどうかを指定します。

有効な値 : true または false。このプロパティのデフォルト値は false です。必要に応じて変更してください。

例 :

```
com.sun.am.policy.config.debug.logfile.rotate=false
```

デバッグログファイルのローテーションサイズ

キー : com.sun.am.policy.config.debug.logfile.rotate.size

説明 : このプロパティは、デバッグログファイルのおよそのサイズをバイト数で指定します。ログファイルのローテーションの必要性を評価するときに使われます。

有効な値 : ローテーションが必要なログファイルのサイズをバイト数で表す 0 以外の符号なし整数値です。このプロパティのデフォルト値は 52428800 バイト (50M バイト) です。必要に応じて変更してください。

注	デバッグログファイルのローテーションフラグが false に設定されている場合は、このプロパティは使われません。

例 :

```
com.sun.am.policy.config.debug.logfile.rotate.size=52428800
```

デバッグ日時の書式文字列

キー : com.sun.am.policy.config.debug.date.format

説明 : このプロパティは、デバッグメッセージが記録された日時の特定に使われるタイムスタンプの書式を指定します。

有効な値 : 有効な java.text.SimpleDateFormat Time Format Syntax 文字列です。詳細は、次の Web サイトを参照してください。

```
http://java.sun.com/j2se/1.3/docs/api/java/text/SimpleDateFormat.html
```

注	• このプロパティのデフォルト値は、<MMM d, yyyy h:mm:ss a z> '*Agent*' です。必要に応じて変更してください。 • このプロパティに無効な値を設定すると、デバッグメッセージにタイムスタンプデータが記録されません。

例 :

```
com.sun.am.policy.config.debug.date.format=[yyyy/MM/dd HH:mm:ss zzz]
```

デバッグ出力の STDOUT フラグ

キー : com.sun.am.policy.config.debug.print.stdout

説明 : このプロパティは、デバッグエンジンがデバッグメッセージを標準の出力ストリームに出力するかどうかを指定します。

有効な値 : true または false。このプロパティのデフォルト値は true です。必要に応じて変更してください。

注

- `true` に設定すると、デバッグエンジンはすべてのデバッグメッセージを標準の出力ストリームに出力します。この結果、WebLogic Server の起動スクリプトを実行したディレクトリのコンソールウィンドウにデバッグメッセージが表示されます。
- このプロパティは、デバッグエンジンがデバッグログファイルに情報を書き込む機能には影響しません。

例 : `com.sun.am.policy.config.debug.print.stdout=true`

エージェントと Sun ONE Identity Server SDK API の使用

Sun ONE Identity Server SDK API を使って、セキュリティと ID を認識するアプリケーションを作成できます。これらのアプリケーションは、Sun ONE Identity Server が提供する豊富なセキュリティおよびポリシーのインフラストラクチャを使って、アプリケーションレベルのポリシーの適用など、セキュリティと ID に関連するカスタムタスクを実行できます。WebLogic Server 向け Sun ONE Identity Server ポリシーエージェントをインストールすると、アプリケーションで Sun ONE Identity Server SDK を使用できるようになります。

SDK の存在だけでは、セキュリティの保護されたアプリケーションの開発には不十分ですが、WebLogic 向けのポリシーエージェントを併用することで、開発したシステムにログオンしているユーザは、いつでもシングルサインオン (SSO) トークンを利用できるようになります。

ログオンユーザが保護されたリソースにアクセスすると、そのユーザが適切に認証されていること、および対応する主体がシステムで利用可能であることをエージェントフィルタが確認します。主体のインスタンスには、`HttpServletRequest.getUserPrincipal()` や `EJBContext.getCallerPrincipal()` など、J2EE のプログラムによるセキュリティ呼び出しでアクセスすることができます。これらのメソッドが返す主体インスタンスは、Sun ONE Identity Server の認証サービスによって認証されたユーザを表します。この主体インスタンスを使うことで、アプリケーション内のどの場所からでも、次の簡単な手順でユーザの SSO トークンにアクセスできるようになります。

1. 次のクラスに主体をダウンキャストします。
   `com.iplanet.amagent.weblogic.realm.AgentUser`

   次に例を示します。

```
....
import java.security.Principal;
import com.iplanet.amagent.weblogic.realm.AgentUser;
....
Principal principal = getEJBContext().getCallerPrincipal();
AgentUser user = null;
if (principal instanceof AgentUser) {
    user = (AgentUser) principal;
}
...
```

2. `AgentUser` API を使って、この主体に関連づけられた SSO トークンを取得します。次に例を示します。

```
...
String ssoTokenId = null;
if (user != null) {
ssoTokenId = user.getSSOTokenID();
}
...
```

取得した SSO トークンは、Identity Server SDK API に対するそれ以後の呼び出しにも利用できます。

注 エージェントはこれらの設定情報を利用することで、エージェントレルムに含まれるユーザと関連した SSO トークンをいつでも利用できます。これらの値の設定が配備環境に適さない場合は、システム全体のパフォーマンスが低下することがあります。「エージェントの設定」を参照してください。

エージェントのアンインストール

WebLogic Server 用に Sun ONE Identity Server ポリシーエージェントをインストールすると、インストールディレクトリ内にアンインストールプログラムが作成されます。エージェントをシステムから完全に削除するときは、このアンインストールプログラムを使います。アンインストールプログラムは、システムにインストールされているすべてのファイルを削除しますが、監査ログメッセージなど、一部のファイルは削除されません。これらのファイルは、手動で削除できます。

アンインストール前のタスク

アンインストールプログラムを起動する前に、次のタスクを次の順序で実行する必要があります。これらのタスクのすべて、または一部を実行しない場合、アプリケーションを利用できなくなったり、システム全体が不安定になるか、利用できなくなることがあります。

1. **エージェントフィルタの削除**：保護されているすべてのアプリケーションを WebLogic Server から削除し、それぞれの配備記述子を編集して、エージェントのインストール時に追加したエージェントフィルタへのすべての参照を削除します。また、これらのアプリケーション用にロールから主体へのマッピングを作成した場合は、これも削除します。配備記述子の編集が完了すると、アプリケーションを WebLogic Server に再配備できるようになります。すでにシステムに配備されているアプリケーションを削除し、再配備する方法については、WebLogic Server のマニュアルを参照してください。
2. **エージェントレルムの削除**：エージェントを正しくアンインストールするには、次にエージェントレルムの設定を削除します。これは、次の手順で行います。
   a. WebLogic Server の管理コンソールにログオンします。コンソールにログインするには、設定されているシステムユーザ名とパスワードを入力します。
   b. 左ペインの「Security」ノードをクリックします。これにより、コンソールの右ペインには WebLogic Server のセキュリティ設定が表示されます。
   c. 右ペインの「Filerealm」タブをクリックします。現在のファイルレルムの詳細がコンソールに表示されます。
   d. 右ペインに表示される書式の「Caching Realm」の下で、使用するエージェントキャッシングレルム以外の適切なキャッシングレルムを選択します。その他のキャッシングレルムが設定されていない場合は、「`defaultCachingRealm`」を選択します。
   e. 「Apply」ボタンをクリックします。

f. WebLogic Server を再起動します。

ファイルレルムの設定が完了し、WebLogic Server を再起動すると、WebLogic Server はそのエージェントレルムを使用しなくなります。この時点で、アンインストールプログラムを使ってエージェントをアンインストールできます。アンインストールプログラムの使用方法を説明している次の項に進んでください。ただし、この項の残りの手順を実行し、エージェントのインストール時に作成したカスタムレルムとキャッシングレルムを削除することをお勧めします。

3. **エージェントキャッシングレルムの削除**：エージェントキャッシングレルムを使わないようにファイルレルムを設定すると、システムからエージェントキャッシングレルムを削除できるようになります。これは、次の手順で行います。

   a. WebLogic Server の管理コンソールにログオンします。コンソールにログインするには、設定されているシステムユーザ名とパスワードを入力します。

   b. 管理コンソールの左ペインで、「+」記号をクリックして「Security」ノードを展開します。

   c. 左ペインの「Security」ノードの下にある「Caching Realms」をクリックします。これにより、システムで使用できるキャッシングレルムがコンソールの右ペインに表示されます。

   d. 右ペインで、エージェントキャッシングレルムに対応する行を探します。この行の最後の列にある削除ボタンをクリックします。エージェントキャッシングレルムの削除を確認するメッセージが表示されます。このメッセージを読み、削除するキャッシングレルムがエージェントキャッシングレルムであることを確認します。「Yes」をクリックして処理を確定します。コンソールに確認メッセージが表示されます。このメッセージの下に表示される「Continue」リンクをクリックします。

   e. キャッシングレルムのリストが更新され、右ペインに表示されます。このリストにエージェントキャッシングレルムが表示されていないことを確認します。

   f. WebLogic Server を再起動します。

   WebLogic Server を再起動すると、エージェントのインストール時に設定したエージェントレルムを削除できるようになります。

4. **エージェントレルムの削除**：エージェントキャッシングレルムを削除すると、これに関連するカスタムレルムであるエージェントレルムを削除できます。これは、次の手順で行います。

   a. WebLogic Server の管理コンソールにログオンします。コンソールにログインするには、設定されているシステムユーザ名とパスワードを入力します。

   b. 管理コンソールの左ペインで、「+」記号をクリックして「Security」ノードを展開します。

c. 左ペインの「Security」ノードの下にある「Realms」をクリックします。これにより、システムで使用できるレルムがコンソールの右ペインに表示されます。

d. 右ペインで、エージェントレルムに対応する行を探します。この行の最後の列にある削除ボタンをクリックします。エージェントレルムの削除を確認するメッセージが表示されます。このメッセージを読み、削除するレルムがエージェントレルムであることを確認します。「Yes」をクリックして処理を確定します。コンソールに確認メッセージが表示されます。このメッセージの下に表示される「Continue」リンクをクリックします。

e. レルムのリストが更新され、右ペインに表示されます。このリストにエージェントレルムが表示されていないことを確認します。

f. WebLogic Server を再起動します。

WebLogic Server を再起動すると、エージェントのインストール時にシステムに設定したすべての内容が削除されます。次の項に進み、エージェントのアンインストールプログラムの使用方法を参照して、システムからエージェントライブラリを削除してください。

アンインストールプログラムの起動

アンインストールプログラムを起動する前に、WebLogic Server が稼働していないことを確認します。WebLogic Server を終了していないと、WebLogic Server が起動しなくなり、システムを利用できなくなることがあります。WebLogic Server が稼働していないことを確認したら、アンインストールプログラムを起動して、システムからエージェントライブラリとその他のファイルを削除してください。

WebLogic Server 向け Sun ONE Identity Server ポリシーエージェントのアンインストールプログラムは、使っているプラットフォームの種類 (Solaris、Windows、HP-UX) により、それぞれ次の手順に従って起動する必要があります。

Solaris 8 でのアンインストールプログラムの起動

Solaris プラットフォームのアンインストールプログラムは、インストールディレクトリに生成されているアンインストールスクリプトを実行することで起動できます。エージェントをアンインストールする手順は次のとおりです。

1. root としてログインします。
2. エージェントがインストールされているディレクトリに移動します。

3. JAVA_HOME 環境変数をバージョン 1.3.1 以降の JDK に設定します。必要なバージョンの JDK がシステムに用意されていない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用します。この JDK は、次の場所に保存されています。

   WebLogic_Install_Dir/bea/jdk131

4. アンインストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。ほとんどの場合、エージェントのアンインストールには GUI 形式のアンインストールプログラムを使います。ただし、リモートサーバの telnet セッションでアンインストールを行う場合にウィンドウ機能を利用できない場合は、コマンド行形式のアンインストールプログラムを使ってエージェントをアンインストールします。アンインストールプログラムを起動するには、次のコマンドを実行します。

   ```
   #./uninstall_wlagent -nodisplay
   ```

   ただし、GUI 形式のアンインストールプログラムを利用する場合は、GUI アンインストールプログラムウィンドウが正しいコンソールに表示されるように DISPLAY 環境変数を設定する必要があります。

注 -nodisplay オプションを指定してコマンド行形式のアンインストールプログラムを使用する場合は、次の手順を省略し、アンインストール手順を詳しく説明している次の項に進んでください。

5. 次のアンインストールスクリプトを呼び出して GUI 形式のアンインストールプログラムを起動します。

   ```
   # ./uninstall_wlagent
   ```

   アンインストールプログラムを利用するには、手順 3 で説明した方法で JAVA_HOME 変数が正しく設定されている必要があります。JAVA_HOME 変数が正しく設定されていない状態でアンインストールスクリプトを実行すると、JAVA_HOME の正しい値を入力するように求められます。

   ```
   Enter JAVA_HOME location (Enter "." to abort):
   ```

   アンインストールプログラムの起動に使う JDK インストールディレクトリの完全パスを入力します。アンインストールを中止するときは、ピリオド (.) を入力します。

   GUI 形式のアンインストールプログラムをコンソールに表示するには、シェルの DISPLAY 環境変数を正しく設定する必要があります。DISPLAY 環境変数が正しく設定されていない状態でアンインストールスクリプトを実行すると、DISPLAY の正しい値を入力するように求められます。

   ```
   Please enter the value of DISPLAY variable (Enter "." to abort):
   ```

上記プロンプトに DISPLAY 変数の正しい値を入力します。インストールを中止するときは、ピリオド (.) を入力します。

注 uninstall_Sun_ONE_Identity_Server_Policy_Agent.class ファイルを使ってエージェントをアンインストールすることもできます。このファイルは、エージェントのインストールディレクトリにあります。

Windows 2000 Server でのアンインストールプログラムの起動

Windows プラットフォームのアンインストールプログラムは、インストールディレクトリに生成されているアンインストールスクリプトを実行することで起動できます。

1. アンインストールプログラムを実行するには、管理者特権が必要です。管理者特権がない場合は、管理者ユーザとしてログインするか、マシンまたはドメインのシステム管理者に依頼して、使用アカウントに適切な権限を付与してもらいます。
2. エージェントがインストールされているディレクトリに移動します。
3. アンインストールスクリプト uninstall_wlagent.bat はこのディレクトリにあります。uninstall_wlagent.bat スクリプトを使ってアンインストールプログラムを起動するには、バージョン 1.3.1 以降の JDK が必要です。これは、コマンドプロンプトウィンドウで次のコマンドを実行することで確認できます。

   ```
   C:¥> java -version

   java version "1.3.1_02"

   Java(TM) 2 Runtime Environment, Standard Edition (build
   1.3.1_02-b02)

   Java HotSpot(TM) Client VM (build 1.3.1_02-b02, mixed mode)
   ```

   必要なバージョンの JDK をシステムパスから利用できない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用します。これは次の場所にあります。

   WebLogic_Install_Dir¥bea¥jdk131

   uninstall_wlagent.bat を実行するには、コマンドプロンプトウィンドウでファイルが保存されている場所まで移動してからファイル名を入力するか、Windows Explorer 上でファイルをダブルクリックします。次に例を示します。

   ```
   C:¥Sun>uninstall_wlagent.bat
   ```

   アンインストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。前述の方法でコマンドプロンプトウィンドウから uninstall_wlagent.bat ファイルを呼び出すか、Windows Explorer 上でこのファイルをダブルクリックすると、GUI 形式のアンインストー

ルプログラムが起動します。ただし、コマンド行形式のアンインストールプログラムを使ってエージェントをアンインストールする必要があるときは、次のように、引数 -nodisplay を指定して uninstall_wlagent.bat ファイルを実行します。

```
C:¥Sun>uninstall_wlagent.bat -nodisplay
```

このコマンドは、コマンドプロンプトウィンドウ上で、このファイルがあるディレクトリで実行します。

注

- システムパスに必要な JDK が設定されている場合は、コントロールパネルの「アプリケーションの追加と削除」コントロールを使ってプログラムをアンインストールできます。システムにインストールされているプログラムのリストから「Sun ONE Identity Server Policy Agent for WebLogic」を選択し、「変更 / 削除」ボタンをクリックします。適切なバージョンの JDK がシステムパスに設定されていない場合は、アンインストールは失敗します。
- コントロールパネルの「アプリケーションの追加と削除」コントロールを使ってエージェントのプログラムをアンインストールする場合は、GUI 形式のアンインストールプログラムだけが起動されます。コマンド行形式のアンインストールプログラムを起動するには、前述のアンインストールスクリプトを使ってください。

HP-UX 11 でのアンインストールプログラムの起動

HP-UX プラットフォームのアンインストールプログラムは、インストールディレクトリに生成されているアンインストールスクリプトを実行することで起動できます。実行手順は次のとおりです。

1. root としてログインします。
2. エージェントがインストールされているディレクトリに移動します。
3. JAVA_HOME 環境変数をバージョン 1.3.1 以降の JDK に設定します。必要なバージョンの JDK がシステムに用意されていない場合は、WebLogic Server 6.1 SP2 サーバに付属する JDK を使用します。この JDK は、次の場所に保存されています。

   WebLogic_Install_Dir/bea/jdk131
4. アンインストールプログラムには、グラフィカルユーザインタフェース (GUI) とコマンド行インタフェースが用意されています。ほとんどの場合、エージェントのアンインストールには GUI 形式のアンインストールプログラムを使います。ただし、リモートサーバの telnet セッションでアンインストールを行う場合にウィ

ンドウ機能を利用できないときは、コマンド行形式のアンインストールプログラムを使ってエージェントをアンインストールします。この場合は次のように、コマンド行に引数 -nodisplay を指定してアンインストールスクリプトを実行します。

```
#./uninstall_wlagent -nodisplay
```

ただし、GUI 形式のアンインストールプログラムを利用する場合は、GUI プログラムウィンドウが正しいコンソールに表示されるように DISPLAY 環境変数を設定する必要があります。

注 -nodisplay オプションを指定してコマンド行形式のアンインストールプログラムを使用する場合は、次の手順を省略し、アンインストール手順を詳しく説明している次の項に進んでください。

5. 次のアンインストールスクリプトを呼び出して GUI 形式のアンインストールプログラムを起動します。

   ```
   # ./uninstall_wlagent
   ```

   アンインストールプログラムを利用するには、手順 3 で説明した方法で JAVA_HOME 変数が正しく設定されている必要があります。JAVA_HOME 変数が正しく設定されていない状態でアンインストールスクリプトを実行すると、JAVA_HOME の正しい値を入力するように求められます。

   ```
   Enter JAVA_HOME location (Enter "." to abort):
   ```

   アンインストールプログラムの起動に使う JDK インストールディレクトリの完全パスを入力します。アンインストールを中止するときは、ピリオド (.) を入力します。

   GUI 形式のアンインストールプログラムをコンソールに表示するには、シェルの DISPLAY 環境変数を正しく設定する必要があります。DISPLAY 環境変数が正しく設定されていない状態でアンインストールスクリプトを実行すると、DISPLAY の正しい値を入力するように求められます。

   ```
   Please enter the value of DISPLAY variable (Enter "." to abort):
   ```

   上記プロンプトに DISPLAY 変数の正しい値を入力します。アンインストールを中止するときは、ピリオド (.) を入力します。

GUI によるエージェントのアンインストール

アンインストールプログラムを起動すると、初期画面が表示されます。質問に答えながら、「Next」をクリックして画面を切り替えます。

1. 「Uninstall Type Selection」画面で「Full」を選択し、「Next」をクリックします。
2. 「Ready to Uninstall」画面で、アンインストール情報を確認します。変更が必要な場合は、「Back」をクリックします。それ以外の場合は「Uninstall Now」をクリックします。
3. アンインストールの進捗状況が「Uninstall Progress」画面に表示されます。
4. 「Uninstallation Summary」画面で、「Details」をクリックすると、アンインストール中に処理された設定情報の詳細を参照できます。「Exit」をクリックしてプログラムを終了します。

注 アンインストールの状態が「Failed」と表示されるときは、「Details」ボタンをクリックしてアンインストールのログファイルの内容を確認し、問題のあったタスクを特定する必要があります。状況によっては、これらの失敗を復旧し、システムを元の状態に戻すことができます。システムを元の状態に戻す方法については、次の項を参照してください。

アンインストールのトラブルシューティング

インストール時に、エージェントのインストールプログラムはシステム上のいくつかの既存ファイルに変更を加えます。これらの変更を完全に元の状態に戻すため、変更時に対象ファイルのバックアップが作成されます。アンインストール時に問題が生じた場合は、これらのバックアップファイルを使ってシステムを手動で元の状態に戻すことができます。

エージェントのインストール時にバックアップされ、アンインストール時に復元されるファイルは、次のとおりです。

- WebLogic Server の起動スクリプト
- `java.security` ファイル

WebLogic Server の起動スクリプト

WebLogic Server の起動スクリプトは、インストール時に変更され、同じディレクトリにバックアップが作成されます。バックアップファイルの名前は *WebLogic_Startup_Script_Name*`-preAgent` です。たとえば、WebLogic Server のインストール時に起動スクリプトの名前を次のように指定したとします。

```
/bea/wlserver6.1/config/examples/startExamplesServer.sh
```

この場合、バックアップファイルの名前は次のようになります。

```
/bea/wlserver6.1/config/examples/startExamplesServer.sh-preAgent
```

java.security ファイル

エージェントのインストールプログラムは、JDK のインストールに WebLogic Server が使用する JCE および JSSE 拡張機能をインストールします。これらの拡張機能を実行時に利用するには、JDK インストールのセキュリティファイルに具体的なプロバイダが指定されている必要があります。エージェントのアンインストール時に元の状態に復元できるように、プロバイダがセキュリティファイルに記録される前にセキュリティファイルのバックアップが作成されます。バックアップファイルの名前は `java.security-preAgent` で、JDK のインストールディレクトリ内の `jre/lib/security` ディレクトリに保存されます。たとえば、インストール時に WebLogic `JAVA_HOME` の値を `/bea/jdk131` に設定した場合は、`java.security` ファイルの名前は次のようになります。

```
/bea/jdk131/jre/lib/security/java.security
```

この場合、バックアップファイルの名前は次のようになります。

```
/bea/jdk131/jre/lib/security/java.security-preAgent
```

インストーラが実行する設定タスク

この付録では、インストールに失敗した場合にアプリケーションサーバを復旧する方法について説明します。

WebLogic 6.1 SP2

WebLogic 6.1 SP2 サーバ向け Sun ONE Identity Server ポリシーエージェントのインストーラは、特定の設定タスクを実行します。システム設定やその他の要因によって、これらのタスクが失敗し、利用不可能なインストールが生成されることがあります。幸い、ほとんどの場合はこれらのタスクを手動で実行することで復旧することができます。次に、WebLogic 6.1 SP2 サーバ向け Sun ONE Identity Server ポリシーエージェントのインストーラを設定する方法について説明します。

WebLogic Server の起動スクリプトの変更

インストーラは、WebLogic Server の起動スクリプトを変更し、新たにインストールされたライブラリ、および Java 仮想マシンの特定の起動プロパティを CLASSPATH に追加します。

Solaris と HP-UX の CLASSPATH の変更

WebLogic Server の起動スクリプトに新たに追加されるのは、次の行です。これは、CLASSPATH 変数が定義される行の上の部分です。

```
AM_INSTALL_DIR=/opt/SUNWam/wlAgent

AM_SDK_DIR=$AM_INSTALL_DIR/amSDK
AM_SDK_LIB_DIR=$AM_SDK_DIR/lib
AM_SDK_LOCALE_DIR=$AM_SDK_DIR/locale
AM_SDK_JAR1=$AM_SDK_LIB_DIR/am_sdk.jar
AM_SDK_JAR2=$AM_SDK_LIB_DIR/am_services.jar
```

```
AM_SDK_JAR3=$AM_SDK_LIB_DIR/am_sso_provider.jar
AM_SDK_JARS=$AM_SDK_JAR1:$AM_SDK_JAR2:$AM_SDK_JAR3
AM_SDK_PATH1=$AM_SDK_DIR:$AM_SDK_LIB_DIR
AM_SDK_PATH2=$AM_SDK_LOCALE_DIR:$AM_SDK_JARS
AM_SDK_CLASSPATH=$AM_SDK_PATH1:$AM_SDK_PATH2

AM_AGT_DIR=$AM_INSTALL_DIR/amAgent
AM_AGT_CONFIG_DIR=$AM_AGT_DIR/config
AM_AGT_LOCALE_DIR=$AM_AGT_DIR/locale
AM_AGT_LIB_DIR=$AM_AGT_DIR/lib
AM_AGT_JAR1=$AM_AGT_LIB_DIR/amagent_core.jar
AM_AGT_JAR2=$AM_AGT_LIB_DIR/amagent_weblogic.jar
AM_AGT_JAR3=$AM_AGT_LIB_DIR/amagent_filter.jar
AM_AGT_JAR4=$AM_AGT_LIB_DIR/amagent_tools.jar
AM_AGT_JARS12=$AM_AGT_JAR1:$AM_AGT_JAR2
AM_AGT_JARS34=$AM_AGT_JAR3:$AM_AGT_JAR4
AM_AGT_JARS=$AM_AGT_JARS12:$AM_AGT_JARS34
AM_AGT_PATH1=$AM_AGT_DIR:$AM_AGT_CONFIG_DIR
AM_AGT_PATH2=$AM_AGT_LOCALE_DIR:$AM_AGT_LIB_DIR:$AM_AGT_JARS
AM_AGT_CLASSPATH=$AM_AGT_PATH1:$AM_AGT_PATH2

AM_CLASSPATH=$AM_SDK_CLASSPATH:$AM_AGT_CLASSPATH
```

これらのエントリを追加すると、CLASSPATH 変数に AM_CLASSPATH が追加されます。

```
CLASSPATH=$AM_CLASSPATH:$WL_HOME:$WL_HOME/lib/weblogic_sp.jar:$W
L_HOME/lib/weblogic.jar:$WL_HOME/samples/eval/cloudscape/lib/clo
udscape.jar:./config/examples/serverclasses
```

Windows の CLASSPATH の変更

WebLogic Server の起動スクリプトに次の行が追加され、CLASSPATH が変更されます。これは、CLASSPATH 変数の定義の直後にあたります。

```
set AM_INSTALL_DIR=C:¥Sun¥SUNWam¥wlAgent

set AM_SDK_DIR=%AM_INSTALL_DIR%¥amSDK
set AM_SDK_LIB_DIR=%AM_SDK_DIR%¥lib
set AM_SDK_LOCALE_DIR=%AM_SDK_DIR%¥locale
set AM_SDK_JAR1=%AM_SDK_LIB_DIR%¥am_sdk.jar
set AM_SDK_JAR2=%AM_SDK_LIB_DIR%¥am_services.jar
set AM_SDK_JAR3=%AM_SDK_LIB_DIR%¥am_sso_provider.jar
set AM_SDK_JARS=%AM_SDK_JAR1%;%AM_SDK_JAR2%;%AM_SDK_JAR3%
set AM_SDK_PATH1=%AM_SDK_DIR%;%AM_SDK_LIB_DIR%
set AM_SDK_PATH2=%AM_SDK_LOCALE_DIR%;%AM_SDK_JARS%
set AM_SDK_CLASSPATH=%AM_SDK_PATH1%;%AM_SDK_PATH2%

set AM_AGT_DIR=%AM_INSTALL_DIR%¥amAgent
```

```
set AM_INSTALL_DIR=C:¥Sun¥SUNWam¥wlAgent
set AM_AGT_CONFIG_DIR=%AM_AGT_DIR%¥config
set AM_AGT_LOCALE_DIR=%AM_AGT_DIR%¥locale
set AM_AGT_LIB_DIR=%AM_AGT_DIR%¥lib
set AM_AGT_JAR1=%AM_AGT_LIB_DIR%¥amagent_core.jar
set AM_AGT_JAR2=%AM_AGT_LIB_DIR%¥amagent_weblogic.jar
set AM_AGT_JAR3=%AM_AGT_LIB_DIR%¥amagent_filter.jar
set AM_AGT_JAR4=%AM_AGT_LIB_DIR%¥amagent_tools.jar
set AM_AGT_JARS12=%AM_AGT_JAR1%;%AM_AGT_JAR2%
set AM_AGT_JARS34=%AM_AGT_JAR3%;%AM_AGT_JAR4%
set AM_AGT_JARS=%AM_AGT_JARS12%;%AM_AGT_JARS34%
set AM_AGT_PATH1=%AM_AGT_DIR%;%AM_AGT_CONFIG_DIR%
set
AM_AGT_PATH2=%AM_AGT_LOCALE_DIR%;%AM_AGT_LIB_DIR%;%AM_AGT_JARS%
set AM_AGT_CLASSPATH=%AM_AGT_PATH1%;%AM_AGT_PATH2%

set AM_CLASSPATH=%AM_SDK_CLASSPATH%;%AM_AGT_CLASSPATH%

set CLASSPATH=%AM_CLASSPATH%;%CLASSPATH%
```

追加される行のうち最後の行は、エージェントが提供するライブラリを CLASSPATH に追加しています。

Java 仮想マシンへのパラメータの追加

Solaris 8、Windows 2000、および HP-UX 11 へのインストールでは、WebLogic Server を呼び出す Java 仮想マシン起動コマンドに次のパラメータが追加されます。

```
-D"com.iplanet.coreservices.configpath=/opt/SUNWam/wlAgent/amSDK
/config/ums"
-D"max_conn_pool=10"
-D"min_conn_pool=1"
```

Solaris および HP-UX でコマンドを実行すると、次のように出力されます。

```
java $JAVA_OPTIONS -classpath $CLASSPATH
-Dweblogic.Domain=examples -Dweblogic.Name=examplesServer
-Dweblogic.management.password=$WLS_PW -Dbea.home=/bea
-Dcloudscape.system.home=./samples/eval/cloudscape/data
-Djava.security.policy==$WL_HOME/lib/weblogic.policy
-D"com.iplanet.coreservices.configpath=/opt/SUNWam/wlAgent/amSDK
/config/ums" -D"max_conn_pool=10" -D"min_conn_pool=1"
weblogic.Server
```

Windows では、次のように出力されます。

```
"%JAVA_HOME%¥bin¥java" -hotspot -ms64m -mx64m -classpath
"%CLASSPATH%" -Dweblogic.Domain=examples
-Dweblogic.Name=examplesServer
-Dweblogic.management.password=%WLS_PW% -Dbea.home="C:¥bea"
-Dcloudscape.system.home=./samples/eval/cloudscape/data
-Djava.security.policy=="C:¥bea¥wlserver6.1/lib/weblogic.policy"
-D"com.iplanet.coreservices.configpath=C:/Sun/SUNWam/wlAgent/amS
DK/config/ums" -D"max_conn_pool=10" -D"min_conn_pool=1"
weblogic.Server
```

拡張機能 JCE 1.2.1 および JSSE 1.0.2 のインストール

インストーラは、拡張機能 JCE 1.2.1 および JSSE 1.0.2 のインストールも行います。これにより、JAVA_HOME/jre/lib/secrurity/java.security ファイルの内容が変更され、各種 jar ファイルが JAVA_HOME/jre/lib/ext ディレクトリにコピーされます。これらの拡張機能のインストールに失敗した場合は、それぞれを手動でインストールできます。拡張機能とインストール方法を説明したマニュアルを入手するときは、次の Web サイトにアクセスしてください。

http://java.sun.com/products/jce および

http://java.sun.com/products/jsse

WebLogic ポリシーエージェントのデバッグエンジンの使用

WebLogic 6.1 SP2 アプリケーションサーバ向け Sun ONE Identity Server ポリシーエージェントには、配備に関する統計情報の収集、各種アプリケーションを保護するエージェントの実行の監視、およびインストールに関する特定の難しい問題の解決に役立つデバッグエンジンが用意されています。

このエンジンを使用するには、デバッグエンジンの設定に合わせてプロパティを正しく設定する必要があります。

特に、次の点に注意してください。

- デバッグレベルに 0 以外の値が設定されたデバッグエンジンが実行可能な場合、エージェントはさまざまな情報をデバッグエンジンに提供します。この情報を適切に保存または表示するには、デバッグエンジンはこの情報を処理する必要があります。この処理によりオーバヘッドが生じるため、システムのパフォーマンスが最優先される場合は、この処理を回避する必要があります。つまり、運用中のシステムでデバッグエンジンを利用することはお勧めできません。配備したアプリケーションを制御された環境でテストする場合に使用してください。
- デバッグレベルに 0 以外の値が設定されたデバッグエンジンを有効にした場合、`Print STDOUT` が無効になり、デバッグファイルが正しく指定されなくなることがあります。このような場合、パフォーマンスが低下し、さらにすべてのデバッグメッセージが失われます。デバッグレベルを 0 に設定して、デバッグエンジンを無効にしてください。
- デバッグエンジンの日時の書式は変更可能なので、特定の文字列を生成してアプリケーションサーバのコンソールで生成されるその他のメッセージとエージェントメッセージを区別することができます。

ロールと主体のマッピングに関するサンプルシナリオ

この付録では、ロールと主体のマッピングの作成に関するサンプルシナリオを提供します。

宣言によるセキュリティ

メソッドの 1 つに対してアクセスが保護された Enterprise JavaBean コンポーネントについて考えてみます。この保護は、`ejb-jar.xml` 配備記述子の assembly-descriptor 要素にセキュリティロール (security-role) とメソッド許可 (method-permission) の要素を追加することで適用できます。

```
<?xml version="1.0"?>

<!DOCTYPE ejb-jar PUBLIC
'-//Sun Microsystems, Inc.//DTD Enterprise JavaBeans 2.0//EN'
'http://java.sun.com/dtd/ejb-jar_2_0.dtd'>
<ejb-jar>
   ...
   <assembly-descriptor>
      <security-role>
         <role-name>FOO</role-name>
      </security-role>
      <method-permission>
         <role-name>FOO</role-name>
      <method>
            <ejb-name>WebProxy</ejb-name>
            <method-intf>Remote</method-intf>
            <method-name>getWebPage</method-name>
            <method-params>
               <method-param>java.lang.String</method-param>
               <method-param>java.lang.String</method-param>
            </method-params>
       </method>
       </method-permission>
```

```
    </assembly-descriptor>
</ejb-jar>
```

「FOO」というセキュリティロールは、weblogic-ejb-jar.xml 配備記述子を使って実際の主体にマッピングできます。

```
<?xml version="1.0"?>
<!DOCTYPE weblogic-ejb-jar PUBLIC '-//BEA Systems, Inc.//DTD
WebLogic 6.0.0 EJB//EN'
    'http://www.bea
.com/servers/wls6000/dtd/weblogic-ejb-jar.dtd'>
<weblogic-ejb-jar>
   <weblogic-enterprise-bean>
      <ejb-name>WebProxy</ejb-name>
      <jndi-name>ejb.WebProxy</jndi-name>
   </weblogic-enterprise-bean>
   <security-role-assignment>
    <role-name>FOO</role-name>
    <principal-name>amAdmin</principal-name>
   </security-role-assignment>
</weblogic-ejb-jar>
```

プログラムによるセキュリティ

サーブレットの 1 つが HttpServletRequest.isUserInRole(String) などのプログラムによるセキュリティ API を使うサンプルアプリケーションについて考えてみます。サーブレットのコードで使うロール名は、配備記述子名と同じ「SAMPLE-ROLE」であると仮定します。

```
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web
Application 2.3//EN" "http://java.sun.c
om/dtd/web-app_2_3.dtd">
<web-app>
  <display-name>Sample Security Aware application</display-name>
    <filter>
      <filter-name>Agent</filter-name>
      <display-name>Agent</display-name>
      <description>Identity Server Policy Agent for WebLogic
Server 6.1</description>
      <filter-class>
         com.iplanet.amagent.weblogic.filter.AgentFilter
      </filter-class>
    </filter>
    <filter-mapping>
      <filter-name>Agent</filter-name>
      <url-pattern>/*</url-pattern>
```

```
    </filter-mapping>
  <servlet>
    <servlet-name>SampleServlet</servlet-name>
    <servlet-class>
       com.iplanet.sample.SampleServlet
    </servlet-class>
    <security-role-ref>
       <description>
          This role is a sample test role for the
          security aware servlet
       </description>
       <role-name>SAMPLE-ROLE</role-name>
       <role-link>SAMPLE-ROLE-LINK</role-link>
    </security-role-ref>
  </servlet>
  <servlet-mapping>
    <servlet-name>SampleServlet</servlet-name>
    <url-pattern>/Test</url-pattern>
  </servlet-mapping>
  <security-role>
    <description>Some description</description>
    <role-name>SAMPLE-ROLE-LINK</role-name>
  </security-role>
</web-app>
```

SAMPLE-ROLE-LINK は、weblogic.xml 配備記述子を使って実際の主体にマッピングされます。

```
<!DOCTYPE weblogic-web-app PUBLIC "-//BEA Systems, Inc.//DTD Web
Application 6.1//EN" "http://www.be
a.com/servers/wls610/dtd/weblogic-web-jar.dtd">
<weblogic-web-app>
   <security-role-assignment>
     <role-name>SAMPLE-ROLE-LINK</role-name>
     <principal-name>Employee</principal-name>
   </security-role-assignment>
</weblogic-web-app>
```

索引

ま

り