Sun Java System Access Manager Policy Agent 2.2 Guide for Microsoft IIS 6.0 With Outlook Web Access 2007/SharePoint 2007

Part No: 820–4581–12
May 17, 2010

100518@24119

Contents

Preface

This Sun Java System Access Manager Policy Agent 2.2 Guide for Microsoft IIS 6.0 With Outlook Web Access 2007/SharePoint 2007 is a web agent guide. Therefore, it provides general information about web agents in the Sun Java System Access Manager Policy Agent 2.2 software set. This guide also provides specific information about Sun Java System Access Manager Policy Agent 2.2 for Microsoft Internet Information Services 6.0.

Furthermore, this guide provides specific information for installing this agent to protect Microsoft Office SharePoint Portal Server 2007 (referred to as Microsoft Office SharePoint throughout this guide) and Outlook Web Access for Microsoft Exchange Server 2007 (referred to as Outlook Web Access throughout this guide). Outlook Web Access is the web-based email service for Microsoft Exchange Server. For more information, see "Using Agent for Microsoft IIS 6.0 with Microsoft Office SharePoint or Outlook Web Access" on page 30.

Throughout this guide the Microsoft Internet Information Services 6.0 deployment container is referred to as Microsoft IIS 6.0. For support and compatibility information about Agent for Microsoft IIS 6.0, see "Supported Platforms for the Microsoft IIS 6.0 Agent" on page 27.

Included in this guide is information about installing, configuring, uninstalling, and troubleshooting web agents, with the focus being on Policy Agent for Microsoft IIS 6.0.

Who Should Use This Book

This *Sun Java System Access Manager Policy Agent 2.2 Guide for Microsoft IIS 6.0 With Outlook Web Access 2007/SharePoint 2007* is intended for use by IT professionals who manage access to their networks. Administrators should understand the following technologies:

- Directory technologies
- JavaServer Pages (JSP) technology
- HyperText Transfer Protocol (HTTP)
- HyperText Markup Language (HTML)
- eXtensible Markup Language (XML)
- Web Services
- Web Technologies

Before You Read This Book

Sun Java System Policy Agent software works with Sun Java System Access Manager. Both products work with Sun Java Enterprise System, a software infrastructure that supports enterprise applications distributed across a network or Internet environment. Furthermore, Sun Java System Directory Server is a necessary component in a new Access Manager deployment since it is used as the data store. To understand how these products interact and to understand this book, you should be familiar with the following documentation:

- Sun Java Enterprise System documentation set, which can be accessed online at `http://docs.sun.com`. All Sun technical documentation is available online through this web site, including the other documentation sets referred to in this list.

  You can browse the documentation archive or search for a specific book title, part number, or subject.
- Sun Java System Directory Server documentation set.
- Sun Java System Access Manager documentation set, which is explained in more detail subsequently in this chapter.
- Sun Java System Access Manager Policy Agent 2.2 documentation set, which is explained in more detail subsequently in this chapter.

How This Book Is Organized

This book is organized in the following manner:

Preface, this chapter, provides information about this book to help you use the book to your best advantage.

Chapter 1, “Introduction to Web Agents for Policy Agent 2.2,” introduces web agents in Policy Agent 2.2, focusing on what all web agents have in common in this release.

Chapter 2, “About Policy Agent 2.2 for Microsoft IIS 6.0,” provides information specific to Policy Agent 2.2 for Microsoft IIS 6.0, focusing on aspects of the agent that make it unique compared to other web agents.

Chapter 3, “Installing Policy Agent 2.2 for Microsoft IIS 6.0,” provides instructions for installing Policy Agent 2.2 for Microsoft IIS 6.0.

Chapter 4, “The Relationship Between the Agent Profile and Web Agents in Policy Agent 2.2,” provides information about the agent profile, which is an optional location for setting the credentials that the web agent must provide to authenticate with Access Manager.

Chapter 5, “Post-Installation Configuration: Policy Agent 2.2 for Microsoft IIS 6.0,” provides information about web agent configuration.

Chapter 6, “Managing Policy Agent 2.2 for Microsoft IIS 6.0,” provides information about the methods available for managing Policy Agent 2.2 for Microsoft IIS 6.0, with most of the information being applicable to all web agents in the Policy Agent 2.2 software set.

Chapter 7, “Uninstalling Policy Agent 2.2 for Microsoft IIS 6.0,” provides instructions for uninstalling Policy Agent 2.2 for Microsoft IIS 6.0.

Appendix A, “Microsoft Office SharePoint or Outlook Web Access: Deploying Agent for Microsoft IIS 6.0,” provides information and instructions for deploying Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint and Outlook Web Access.

Appendix B, “Troubleshooting a Web Agent Deployment,” provides troubleshooting instructions for problems that might occur in Policy Agent 2.2 for Microsoft IIS 6.0.

Appendix C, “Web Agent `AMAgent.properties` Configuration File,” provides a list of the properties in the web agent `AMAgent.properties` configuration file in Policy Agent 2.2 for Microsoft IIS 6.0, with most properties being applicable to all the web agents in the Policy Agent 2.2 software set.

Appendix D, “Error Codes,” provides a list of error codes that might be encountered during installation or configuration.

Related Books

Documentation sets, some of which are mentioned in this preface, are available at `http://docs.sun.com`.

Access Manager Documentation Set

Policy Agent 2.2 was first introduced with Access Manager 7, but now also supports Access Manager 7.1. The information in the table that follows specifies documents in the Access Manager 7 documentation set, which is available at the following location:

`http://docs.sun.com/coll/1292.1`

The Access Manager 7.1 documentation set is available at this location:

`http://docs.sun.com/coll/1292.2`

TABLE P–1 Access Manager 7 2005Q4 Documentation Set

Title	Description
Sun Java System Access Manager 7.1 Release Notes	Available after the product is released. Contains last-minute information, including a description of what is new in this current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.
Sun Java System Access Manager 7.1 Technical Overview	Provides an overview of how Access Manager components work together to consolidate identity management and to protect enterprise assets and web-based applications. Explains basic Access Manager concepts and terminology
Sun Java System Access Manager 7.1 Deployment Planning Guide	Provides information about planning a deployment within an existing information technology infrastructure
Sun Java System Access Manager 7.1 Performance Tuning and Troubleshooting Guide	Describes how to tune Access Manager and its related components.
Sun Java System Access Manager 7.1 Administration Guide	Describes how to use the Access Manager console as well as how to manage user and service data via the command line.
Sun Java System Access Manager 7.1 Federation and SAML Administration Guide	Provides information about the features in Access Manager that are based on the Liberty Alliance Project and SAML specifications. It includes information on the integrated services based on these specifications, instructions for enabling a Liberty-based environment, and summaries of the application programming interface (API) for extending the framework.
Sun Java System Access Manager 7.1 Developer's Guide	Offers information on how to customize Access Manager and integrate its functionality into an organization's current technical infrastructure. Contains details about the programmatic aspects of the product and its API.
Sun Java System Access Manager 7.1 C API Reference	Provides summaries of data types, structures, and functions that make up the Access Manager public C APIs.
Sun Java System Access Manager 7.1 Java API Reference	Are generated from Java code using the JavaDoc tool. The pages provide information on the implementation of the Java packages in Access Manager.
Sun Java System Access Manager Policy Agent 2.2 User's Guide	Provides an overview of Policy Agent software, introducing web agents and J2EE agents. Also provides a list of web agents and J2EE agents currently available.

Updates to the *Release Notes* and links to modifications of the core documentation can be found on the Access Manager page at the Sun Java System 2005Q4 documentation web site. Updated documents are marked with a revision date.

Policy Agent 2.2 Documentation Set

Other Policy Agent guides, besides this guide, are available as described in the following sections:

- "Sun Java System Access Manager Policy Agent 2.2 User's Guide" on page 11
- "Other Individual Agent Guides" on page 11
- "Release Notes" on page 12

Sun Java System Access Manager Policy Agent 2.2 User's Guide

The *Sun Java System Access Manager Policy Agent 2.2 User's Guide* is available in two documentation sets: the Access Manager documentation set as described in Table P–1 and in the Policy Agent 2.2 documentation set as described in this section.

Other Individual Agent Guides

The individual agents in the Policy Agent 2.2 software set, of which this book is an example, are available on a different schedule than Access Manager itself. Therefore, documentation for Access Manager and Policy Agent are available in separate sets, except for the *Sun Java System Access Manager Policy Agent 2.2 User's Guide*, which is available in both documentation sets.

The documentation for the individual agents is divided into two subsets: a web Policy Agent subset and a J2EE Policy Agent subset.

Each web Policy Agent 2.2 guide provides general information about web agents and installation, configuration, and uninstallation information for a specific web agent.

Each J2EE Policy Agent 2.2 guide provides general information about J2EE agents and installation, configuration, and uninstallation information for a specific J2EE agent.

The individual agent guides are listed along with supported server information in the following chapters of the *Sun Java System Access Manager Policy Agent 2.2 User's Guide*:

Web Agents Chapter 2, "Access Manager Policy Agent 2.2 Web Agents: Compatibility, Supported Servers, and Documentation," in *Sun Java System Access Manager Policy Agent 2.2 User's Guide*

J2EE Agents	Chapter 3, "Access Manager Policy Agent 2.2 J2EE Agents: Compatibility, Supported Servers, and Documentation," in *Sun Java System Access Manager Policy Agent 2.2 User's Guide*

Release Notes

The *Sun Java System Access Manager Policy Agent 2.2 Release Notes* are available online after an agent or set of agents is released. The release notes include a description of what is new in the current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.

Sun Java Enterprise System Product Documentation

For useful information for related products, see the following documentation collections on the Sun Java Enterprise System documentation web site (`http://docs.sun.com/prod/entsys.05q4`)

- Sun Java System Directory Server:

  `http://docs.sun.com/coll/1316.1`
- Sun Java System Web Server:

  `http://docs.sun.com/coll/1308.1`
- Sun Java System Application Server:

  `http://docs.sun.com/coll/1310.1`
- Sun Java System Message Queue:

  `http://docs.sun.com/coll/1307.1`
- Sun Java System Web Proxy Server:

  `http://docs.sun.com/coll/1311.1`

Accessing Resources Online

For product downloads, professional services, patches and support, and additional developer information, go to the following:

Software Downloads

`http://www.oracle.com/technology/software/index.html`

Oracle Advanced Customer Services for Systems
`http://www.oracle.com/us/support/systems/advanced-customer-services/index.html`

Patches and Support
`http://sunsolve.sun.com/`

Developer Information
`http://developers.sun.com/downloads/`

Contacting Technical Support

If you have technical questions about this product that are not answered in the product documentation, go to:

`http://www.sun.com/service/contacting/index.xml`

Related Third-Party Web Site References

Oracle is not responsible for the availability of third-party web sites mentioned in this document. Oracle does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Oracle will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Oracle Welcomes Your Comments

Oracle is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to `http://docs.sun.com` and click Feedback. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the guide or at the top of the document.

For example, the title of this guide is *Sun Java System Access Manager Policy Agent 2.2 Guide for Microsoft IIS 6.0 With Outlook Web Access 2007/SharePoint 2007*, and the part number is 820-4581-12.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://docs.sun.com`)
- Support (`http://www.sun.com/support/`)
- Training (`http://education.oracle.com`) – Click the Sun link in the left navigation bar.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to `http://docs.sun.com` and click Feedback.

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–2 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`

`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`

aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for shells that are included in the Solaris OS. Note that the default system prompt that is displayed in command examples varies, depending on the Solaris release.

TABLE P–3 Shell Prompts

Shell	Prompt
Bash shell, Korn shell, and Bourne shell	`$`
Bash shell, Korn shell, and Bourne shell for superuser	`#`
C shell	`machine_name%`
C shell for superuser	`machine_name#`

CHAPTER 1

Introduction to Web Agents for Policy Agent 2.2

The Sun Java System Access Manager Policy Agent 2.2 software set includes J2EE agents and web agents. This guide discusses web agents, the functionality of which has increased for this release. This chapter provides a brief overview of web agents in the 2.2 release as well as some concepts you need to understand before proceeding with a web agent deployment. For a general introduction of agents, both J2EE agents and web agents, see *Sun Java System Access Manager Policy Agent 2.2 User's Guide*.

Topics in this chapter include:

- "Uses of Web Agents" on page 17
- "How Web Agents Work" on page 18
- "What's New About Web Agents" on page 19

Uses of Web Agents

Web agents function with Sun Java System Access Manager to protect content on web servers and web proxy servers from unauthorized intrusions. They control access to services and web resources based on the policies configured by an administrator. Web agents perform these tasks while providing single sign-on (SSO) and cross domain single sign-on (CDSSO) capabilities as well as URL protection.

Web agents are installed on deployment containers for a variety of reasons. Here are three examples:

- A web agent on a human resources server prevents non-human resources personnel from viewing confidential salary information and other sensitive data.
- A web agent on an operations deployment container allows only network administrators to view network status reports or to modify network administration records.

- A web agent on an engineering deployment container allows authorized personnel from many internal segments of a company to publish and share research and development information. At the same time, the web agent restricts external partners from gaining access to the proprietary information.

In each of these situations, a system administrator must set up policies that allow or deny users access to content on a deployment container. For information on setting policies and for assigning roles and policies to users, see the *Sun Java System Access Manager 7.1 Administration Guide*.

How Web Agents Work

When a user points a browser to a particular URL on a protected deployment container, a variety of interactions take place as explained in the following numbered list. See the terminology list immediately following this numbered list for a description of terms.

1. The web agent intercepts the request and checks information from the request against not-enforced lists. If specific criteria are met, the authentication process is by passed and access is granted to the resource.
2. If authentication is required, the web agent validates the existing authentication credentials. If the existing authentication level is insufficient, the appropriate Access Manager Authentication Service will present a login page. The login page prompts the user for credentials such as username and password.
3. The authentication service verifies that the user credentials are valid. For example, the default LDAP authentication service verifies that the username and password are stored in Sun Java System Directory Server. You might use other authentication modules such as RADIUS and Certificate modules. In such cases, credentials are not verified by Directory Server but are verified by the appropriate authentication module.
4. If the user's credentials are properly authenticated, the web agent checks if the users is authorized to access the resource.
5. Based on the aggregate of all policies assigned to the user, the individual is either allowed or denied access to the URL.

Terminology: How Web Agents Work

Authentication Level The ability to access resources can be divided into levels. Therefore, different resources on a deployment container (such as a web server or proxy server) might require different levels of authentication

Service Access Manager is made of many components. A service is a certain type of component that performs specific tasks. Some of the Access Manager services available are Authentication Service, Naming Service, Session Service, Logging Service, and Policy Service.

Authentication Module An authentication interface, also referred to as an authentication module, is used to authenticate a user on Access Manager.

Roles Roles are a Directory Server entry mechanism. A role's members are LDAP entries that possess the role.

Policy A policy defines rules that specify access privileges to protected resources on a deployment container, such as a web server.

What's New About Web Agents

Several important features have been added to the web agents in the 2.2 release as follows:

- “Web Agents Developed in the OpenSSO Project” on page 19
- “Support for Fetching User Session Attributes” on page 20
- “Log Rotation” on page 21
- “Policy-Based Response Attributes” on page 22
- “Composite Advice” on page 23
- “Additional Method for Fetching the `REMOTE_USER` Server Variable” on page 23
- “Malicious Header Attributes Automatically Cleared by Agents” on page 24
- “Load Balancing Enablement” on page 24
- “Support for Heterogeneous Agent Types on the Same Machine” on page 25
- “Support for Turning Off FQDN Mapping” on page 25
- “Backward Compatibility With Access Manager 6.3” on page 26

Web Agents Developed in the OpenSSO Project

This version of Agent for Microsoft IIS 6.0 is a Policy Agent 2.2 web agent that was developed through the OpenSSO project.

However, the fact that this agent was developed through the OpenSSO project does not have a significant impact on how the agent functions.

Most web agents developed through the OpenSSO project are configured in a different manner than their non-OpenSSO project counterparts. This particular web agent (for Microsoft IIS 6.0 With Outlook Web Access 2007/SharePoint 2007) is unique in that it does not use the common installation and configuration commands used by other OpenSSO web agents. For more information about the OpenSSO Project, see `https://opensso.dev.java.net/`.

Support for Fetching User Session Attributes

Before this release of web agents, header and cookie information was retrieved, or *sourced*, solely from user profile properties. Now, header and cookie information can also be sourced from session properties.

Use the following property to choose how you want session attributes retrieved:

```
com.sun.am.policy.agents.config.session.attribute.fetch.mode
```

For the preceding property, the following modes are available as retrieval methods:

- NONE
- HTTP_HEADER
- HTTP_COOKIE

The following example illustrates this property with the retrieval method set to `HTTP_HEADER`:

```
com.sun.am.policy.agents.config.session.attribute.fetch.mode = HTTP_HEADER
```

The source of header and cookie information is controlled by the following configuration property from the web agent `AMAgent.properties` configuration file:

```
com.sun.am.policy.agents.config.session.attribute.map
```

This configuration property has the same format as an LDAP header property. The following is an example of how this configuration property can be set:

```
com.sun.am.policy.agents.config.session.attribute.map =
name-of-session-attribute1|name-of-header-attribute1,
name-of-session-attribute2|name-of-header-attribute2
```

Where *name-of-session-attribute1* and other similarly named properties, or *attributes*, in the preceding code represent actual property names.

Benefit - Support for Fetching User Session Attributes: The benefit of this feature is that session properties can be more effective for transferring information, especially dynamic information. Prior to this release, agents could only fetch users' profile attributes, which tend to be static attributes. However, session attributes allow applications to obtain dynamic user information when necessary. Since this feature allows you to fetch non-user profile attributes, you can fetch attributes such as SAML assertion.

Log Rotation

Starting with this release of web agents, when the current log file reaches a specific size, a new log file is created. Log information is then stored in the new log file until it reaches the size limit. This default behavior is configurable. Therefore, log rotation can be turned off and the size limit can be changed.

Note – The type of information stored in log files has not changed in Policy Agent 2.2. The following types of information are logged:

- Troubleshooting information
- Access denied information
- Access allowed information

The troubleshooting, or diagnostic, information is stored in log files, locally, with the web agent. The access denied and access allowed information, which is often referred to as audit-related information, can be stored both locally and with Access Manager.

Configuration that relates to the local log files is performed in the web agent `AMAgent.properties` configuration file. Configuration that relates to the audit related logs stored with Access Manager is performed in the Access Manager `AMConfig.properties` configuration file.

The log rotation described in this section refers to logs that store troubleshooting information locally.

Log rotation is controlled by the following configuration property in the web agent `AMAgent.properties` configuration file:

```
com.sun.am.policy.agents.config.local.log.rotate
```

Log rotation occurs automatically since the default value of this property is `true`. When this property is set to `false`, no rotation takes place for the local log file.

The following example shows this configuration property set to `true`:

```
com.sun.am.policy.agents.config.local.log.rotate = true
```

The following properties are also related to log rotation:

- The value for following configuration property indicates the location of the debug file:

  `com.sun.am.policy.agents.config.local.log.file`
- The value of following configuration property indicates the maximum number of bytes the debug file holds:

```
com.sun.am.policy.agents.config.local.log.size
```

The following code example demonstrates how to set the property that controls log file size so that a new log file is created when the current log file reaches a specific size.

```
com.sun.am.policy.agents.config.local.log.size: n
```

Where *n* represents the size of a file in bytes. The file size should be a minimum of 3000 bytes. The default size is 10 megabytes.

Note – By default, the log file size property is not exposed in the web agent `AMAgent.properties` configuration file. If you want to change the default size, add a line to the file setting this property to the file size desired.

When a new log file is created an index appends to the name of the log file as such:

amAgent-1
amAgent-2

Where *amAgent* represents the fully qualified path name to the log files excluding the appended number. The numbers *1* and *2* represent the appended number. The appended number indicates the chronological order in which information of a given size was filed away into its respective log file. There is no limit to the number of log files that can be rotated.

Benefit - Log Rotation: Prior to this release of web agents, all logging messages were written to the same log file. However, saving all log information to a single log file has the potential of exhausting disk space. The log rotation feature solves this problem.

Policy-Based Response Attributes

Starting with this release of web agents, a new method is available for retrieving LDAP user attributes based on Access Manager policy configurations.

Policy-based response attributes take advantage of functionality now available in Access Manager that involves querying policy decisions. In previous versions of Access Manager, header attributes could only be determined by the list of attribute-value pairs in the agent configuration. Now, header attributes can also be determined by Access Manager policy configurations. With policy-based response attributes you can define attribute-value pairs at each policy definition as opposed to the method used in prior versions of Access Manager, which only allowed pairs to be defined globally in the agent configuration. For more information on policy-based response attributes, see “Providing Personalization With Policy-Based Response Attributes” on page 60

Benefit - Policy-Based Response Attributes: The benefit of policy-based response attributes is that they allow for personalization, improve the deployment process, allow greater flexibility in terms of customization, and provide central and hierarchical control of attribute values.

Personalization is provided in that an application can retrieve specific user information, such as a name, from a cookie or HTTP header and present it to the user in the browser.

Defining attribute-value pairs at each policy definition instead of at the root level allows an attribute value to be distributed only to the applications that need it. Furthermore, you can customize attribute names allowing the same attribute name to have entirely different property values for two different applications.

Composite Advice

Starting with this release, web agents provide a composite advice feature. This feature allows the policy and authentication services of Access Manager to decouple the advice handling mechanism of the agents. This allows you to introduce and manage custom advices by solely writing Access Manager side plug-ins. Starting with this release, you are not required to make changes on the agent side. Such advices are honored automatically by the composite advice handling mechanism.

Benefit - Composite Advice: A benefit of composite advice is that you can incorporate a custom advice type without having to make changes to an agent deployment. Prior to the 2.2 release of web agents, no interface existed on the client side to write client-side plug-ins.

Additional Method for Fetching the `REMOTE_USER` Server Variable

Prior to this release of web agents, the only method for fetching the value of the `REMOTE_USER` variable set by an agent was from session properties. Starting with the 2.2 release, the value can also be fetched from user profiles. This fetching process uses LDAP.

By default the value for the `REMOTE_USER` is fetched from the session. If the value needs to be fetched from LDAP, the following property needs to be defined in the web agent `AMAgent.properties` configuration file:

```
com.sun.am.policy.am.userid.param.type = LDAP
```

The following property can still be used to configure the key (*key* refers to the value assigned to this property) that needs to be searched. In addition to setting the preceding property, you need to give the correct LDAP attribute name for the following property.

```
com.sun.am.policy.am.userid.param
```

For example the property will be set as follows:

```
com.sun.am.policy.am.userid.param = ldap-attribute-name
```

where *ldap-attribute-name* represents the name of an LDAP attribute.

To enable the `REMOTE_USER` setting for a globally not-enforced URL as specified in the web agent `AMAgent.properties` configuration file (this is a URL that can be accessed by unauthenticated users) you must set the following property in the web agent `AMAgent.properties` configuration file to `true`. While the following example, has the value is set to `true`, the default value is `false`:

```
com.sun.am.policy.agents.config.anonymous_user.enable = true
```

When you set this property value to `true`, the value of `REMOTE_USER` will be set to the value contained in the following property in the web agent `AMAgent.properties` configuration file. In the following example the value is set to `anonymous`, which is the default:

```
com.sun.am.policy.agents.config.anonymous_user = anonymous
```

Benefit - Additional Method for Fetching the REMOTE_USER Server Variable: The benefit of this feature is that it gives better customization for end users since the `REMOTE_USER` server variable can now be obtained from either session attributes or user profile attributes.

Also, you do not need to write server-side plug-in code in order to add session attributes after authentication, which is necessary when this value is fetched from session properties.

Malicious Header Attributes Automatically Cleared by Agents

Starting with this release of web agents, malicious header attributes are automatically cleared.

Benefit - Header Attributes Set by Agents Automatically Cleared: The benefit of this automatic clean up is that security is improved. Header information that is *not* automatically cleared has greater risk of being accessed.

Load Balancing Enablement

Starting with this release of web agents, the default agent host port and protocol settings can be overridden to enable load balancing. For more information, see “Enabling Load Balancing” on page 68.

Benefit - Load Balancing Enablement: The benefit of this override capability is that you do not need to manually change the hostname, port, and protocol settings to enable load balancing.

Support for Heterogeneous Agent Types on the Same Machine

Starting with this release of web agents, you can install different types of agents on the same machine. Prior to this release, you could not install web agents from different product groups on the same machine. For example, previously, an agent instance for Sun Java System Web Server 6.1 and an agent instance for Apache 2.0.52 could not be installed on the same machine. Now, they can.

Benefit - Support for Heterogeneous Agent Types on Same Machine: The benefit of this feature is that a deployment that has agents in a multi-server scenario requires fewer hardware sources.

Support for Turning Off FQDN Mapping

Starting with this release, fully qualified domain name (FQDN) mapping of HTTP requests can be disabled. In prior web agent releases, the methods employed for checking if a user is using a valid URL could not be turned off.

This checking capability is controlled by the FQDN default and the FQDN map properties in the web agent `AMAgent.properties` configuration file as follows:

- `com.sun.am.policy.agents.config.fqdn.default`
- `com.sun.am.policy.agents.config.fqdn.map`

A toggling capability has been introduced that allows FQDN checking to be turned off. The following property allows for this toggling:

```
com.sun.am.policy.agents.config.fqdn.check.enable
```

The following property specifies whether the request URLs that are present in user requests are checked against the FQDN default and the FQDN map properties by the web agent:

```
com.sun.am.policy.agents.config.fqdn.check.enable
```

The valid values are `true` and `false`.

`true` The request URLs that are present in user requests are checked against FQDN values.

`false` No checking occurs against FQDN values.

The default value is `true`. If no value is specified, then the default value, `true`, is used.

Benefit - Support for Turning Off FQDN Mapping: This feature allows you to turn off or on FQDN mapping comparison. This feature can be beneficial when a deployment includes a number of virtual servers for which the agent is configured using FQDN mapping.

Backward Compatibility With Access Manager 6.3

Most agents in the Policy Agent 2.2 software set are backward compatible with Access Manager 6.3 Patch 1 or greater. For compatibility information specific to Policy Agent 2.2 for Microsoft IIS 6.0, see "Compatibility of Agent for Microsoft IIS 6.0 With Access Manager" on page 28.

Note – Policy Agent 2.2 is only compatible with Access Manager 6.3 when the Access Manager patch has been applied.

Be aware that Policy Agent 2.2 takes advantage of certain features that exist in Access Manager 7 that do not exist in Access Manager 6.3, such as "composite advices," "policy-based response attributes," and others.

CHAPTER 2

About Policy Agent 2.2 for Microsoft IIS 6.0

This chapter provides information about Sun Java System Policy Agent 2.2 as it pertains specifically to Microsoft IIS 6.0.

While the individual web agents tend to be similar in terms of installation and configuration, they can have unique characteristics that allow them to interact with unique characteristics in the underlying deployment container, such as a web server or proxy server. Therefore, this chapter describes characteristics that are unique to this agent, Sun Java System Access Manager Policy Agent 2.2 for Microsoft IIS 6.0, and that are unique to just the deployment container, Microsoft IIS 6.0. This chapter also summarizes specific tasks you might need to perform because of the unique characteristics of the deployment container.

Supported Platforms and Compatibility of Agent for Microsoft IIS 6.0

- "Supported Platforms for the Microsoft IIS 6.0 Agent" on page 27
- "Compatibility of Agent for Microsoft IIS 6.0 With Access Manager" on page 28

Supported Platforms for the Microsoft IIS 6.0 Agent

The following table shows the supported platforms for the for the Microsoft Internet Information Services (IIS) 6.0 policy agent.

TABLE 2–1 Supported Platforms for the Microsoft IIS 6.0 Agent

Agent For	Supported Platforms
Microsoft IIS 6.0	Windows Server 2003, Enterprise Edition, 32-bit and 64-bit systems Windows Server2003, Standard Edition, 32-bit and 64-bit systems
Microsoft IIS 6.0 with Outlook Web Access 2003	Windows Server 2003, Enterprise Edition, 32-bit systems only Windows Server2003, Standard Edition, 32-bit systems only
Microsoft IIS 6.0 with Outlook Web Access 2007	Windows Server 2003, Enterprise Edition, 64-bit systems only Windows Server2003, Standard Edition, 64-bit systems only
Microsoft IIS 6.0 with SharePoint 2003 and Microsoft IIS 6.0 with SharePoint 2007	Windows Server 2003, Enterprise Edition, 32-bit systems only Windows Server2003, Standard Edition, 32-bit systems only
Microsoft IIS 6.0 64-bit agent on IIS 7.0 and IIS 7.5 with Office SharePoint Server 2007	Windows Server 2008, 64-bit systems only For more information see, Appendix E, "Configuring the IIS 6.0 64-bit Agent With IIS 7.x With Office SharePoint Server 2007 on Windows Server 2008."

Notes

- Support for Windows Server 2003, Enterprise Edition and Windows Server 2003, Standard Edition includes all service packs such as SP1, SP2, and so on.
- Support for Windows Server 2008 includes all service packs such as SP1, SP2, and so on.
- This agent does not apply to Microsoft Exchange 2003 or Microsoft Office SharePoint Portal Server 2003. For information about protecting those resources, see the *Sun Java System Access Manager Policy Agent 2.2 Guide for Microsoft Internet Information Services 6.0*.

Compatibility of Agent for Microsoft IIS 6.0 With Access Manager

The compatibility of Agent for Microsoft IIS 6.0 with Access Manager varies depending on the deployment the agent is protecting:

- Microsoft IIS 6.0
- Microsoft IIS 6.0 to protect Microsoft Office SharePoint 2007 or Outlook Web Access 2007

Compatibility of Policy Agent 2.2 With Specific Access Manager Versions

Most agents in the Policy Agent 2.2 release are compatible with Access Manager 6.3 Patch 1 forward.

Moreover, all the 2.2 agents are compatible to some degree with Access Manager 7.0 and Access Manager 7.1. This compatibility applies to both of the available modes of Access Manager: Realm Mode and Legacy Mode.

However, when Agent for Microsoft IIS 6.0 is deployed to protect Microsoft Office SharePoint 2007 or Outlook Web Access 2007, Access Manager 6.3 is not supported and not all the specific patch versions of Access Manager 7.0 and Access Manager 7.1 are supported. See the patch compatibility list that follows.

The best practice is to install the latest Access Manager patches to ensure that all enhancements and fixes are applied.

Microsoft IIS 6.0

- Access Manager 6.3 from Patch 1 forward

  However, certain limitations apply. For more information about the limitations, see "Backward Compatibility With Access Manager 6.3" on page 26
- Access Manager 7.0 all
- Access Manager 7.1 all

Microsoft IIS 6.0 to protect Microsoft Office SharePoint 2007 or Outlook Web Access 2007

- Access Manager 7.0 series from Patch 7 forward
- Access Manager 7.1 series from Patch 1 forward

Information Specific to Agent for Microsoft IIS 6.0

This section describes characteristics that are unique about this specific web agent.

Note – To work with this web agent, you should have a thorough understanding of Microsoft IIS 6.0. Besides an understanding of the overall architecture, you should have an understanding of various concepts and technologies as related to Microsoft IIS 6.0, including the following: application pools, web sites, and authentication methods.

Agent for Microsoft IIS 6.0 is an ISAPI (Internet Server API) extension application. It is deployed as a wildcard application mapping to a web site. Therefore, when deployed for a particular web site, this agent intercepts every request for accessing the resources on that web site. It does authentication and policy evaluation, thereby providing SSO.

However, for protecting Microsoft Office SharePoint and Outlook Web Access, the agent is deployed as an ISAPI filter. In this case, authentication is provided and SSO is enabled by the agent, but policy evaluation is managed by whichever application you have installed: Microsoft Office SharePoint or Outlook Web Access.

The following subsections describe unique characteristics of Agent for Microsoft IIS 6.0.

- "Using Agent for Microsoft IIS 6.0 with Microsoft Office SharePoint or Outlook Web Access" on page 30
- "Multiple Instances of Web Agent Not Supported on Same System" on page 31

Using Agent for Microsoft IIS 6.0 with Microsoft Office SharePoint or Outlook Web Access

Besides the option of having Agent for Microsoft IIS 6.0 protect Microsoft IIS 6.0 Server, you can also configure the agent to protect Microsoft Office SharePoint Portal Server 2007 (referred to as Microsoft Office SharePoint throughout this guide) or Outlook Web Access for Microsoft Exchange Server 2007 (referred to as Outlook Web Access throughout this guide). Outlook Web Access is the web-based email service for Microsoft Exchange Server.

This guide provides specific instructions for SharePoint and Outlook Web Access in Appendix A, "Microsoft Office SharePoint or Outlook Web Access: Deploying Agent for Microsoft IIS 6.0."

Microsoft Office SharePoint

When you install Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint, the agent enables single sign-on (SSO) for SharePoint with all the applications configured in Access Manager. When a user attempts to access SharePoint, Agent for Microsoft IIS 6.0 displays an Access Manager log-in screen. Once authenticated, the user can access SharePoint and all other applications that are secured by Access Manager.

Outlook Web Access

When you install Agent for Microsoft IIS 6.0 to protect Outlook Web Access, the agent enables single sign-on (SSO) for Outlook Web Access with all the applications configured in Access Manager. When a user attempts to access Outlook Web Access, Agent for Microsoft IIS 6.0 displays an Access Manager log-in screen. Once authenticated, the user can access the Outlook Web Access applications, such as email, and all the other applications that are secured by Access Manager.

Multiple Instances of Web Agent Not Supported on Same System

Policy Agent 2.2 for Microsoft IIS 6.0 is unique in that only one instance of Microsoft IIS 6.0 can be installed per computer system. Therefore, you cannot install multiple instances of Agent for Microsoft IIS 6.0 on the same computer system.

However, you can configure multiple web sites on one machine, allowing the agent to be configured for multiple web sites on multiple application pools. All the same, the agent cannot be configured for multiple web sites on the same application pool. Support is only provided for a single web site associated with a single application pool.

CHAPTER 3

Installing Policy Agent 2.2 for Microsoft IIS 6.0

Policy Agent 2.2 works in tandem with Access Manager to control user access to deployment containers (such as web servers) in an enterprise.

This chapter explains how to install Policy Agent 2.2 for Microsoft IIS 6.0. For information on supported platforms, see "Supported Platforms and Compatibility of Agent for Microsoft IIS 6.0" on page 27.

Note – Only one instance of Microsoft IIS 6.0 can be installed per computer system. You cannot install multiple instances of Agent for Microsoft IIS 6.0 on the same computer system. For more information, see "Information Specific to Agent for Microsoft IIS 6.0" on page 29.

This chapter leads you through the pre-installation, installation, and installation-related configuration steps. First, perform the pre-installation (preparation) steps. Then, perform the basic installation.

Next, perform the installation-related configuration, which is divided into two distinct phases. The first phase involves creating a configuration file while the second phase involves using this configuration file to configure Policy Agent 2.2 for Microsoft IIS 6.0.

If you want to protect Microsoft Office SharePoint or Outlook Web Access with the agent, additional information is required. See Appendix A, "Microsoft Office SharePoint or Outlook Web Access: Deploying Agent for Microsoft IIS 6.0."

After you complete the configuration, verify that the installation was successful.

Next, complete the required post-installation tasks described in Chapter 5, "Post-Installation Configuration: Policy Agent 2.2 for Microsoft IIS 6.0."

This chapter contains the following sections:

- "Preparing To Install Agent for Microsoft IIS 6.0" on page 34
- "Installing Agent for Microsoft IIS 6.0" on page 34

- “Configuring Agent for Microsoft IIS 6.0 for a Web Site” on page 40
- “Verifying a Successful Installation of Policy Agent 2.2” on page 42

Preparing To Install Agent for Microsoft IIS 6.0

Follow the specific steps outlined in this section before you install the web agent to reduce the chance of complications occurring during and after the installation.

▼ To Prepare To Install Agent for Microsoft IIS 6.0

Perform the following pre-installation tasks:

1. **Ensure that Policy Agent 2.2 for Microsoft IIS 6.0 is supported on the desired platform as listed in “Supported Platforms and Compatibility of Agent for Microsoft IIS 6.0” on page 27.**

2. **Install Microsoft IIS 6.0 if not already installed.**

   Refer to the Microsoft IIS 6.0 documentation for details on how best to install and configure this server for your platform.

3. **Ensure that Microsoft IIS 6.0 has the latest patches available.**

4. **Set your JAVA_HOME environment variable to a JDK version 1.4 or higher.**

Next Steps To install Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access you must perform additional pre-installation steps. See “Microsoft Office SharePoint and Outlook Web Access: Preparing to Install the Agent” on page 76.

Installing Agent for Microsoft IIS 6.0

Two tasks follow required for basic installation of the agent. The installation program that installs Agent for Microsoft IIS 6.0 has one interface, a command-line interface.

▼ To Unzip Binaries of Agent for Microsoft IIS 6.0

This task applies to all deployments of Agent for Microsoft IIS 6.0, including deployments where the agent protects Microsoft Office SharePoint or Outlook Web Access.

You must have administrator privileges to run the installation program.

Before You Begin

Note – Prior to unzipping the product binaries, as shown in this task, you must copy the agent `.zip` file to a directory. For example, create a directory "`Agents`" in `C:\` and copy the `.zip` file to that `Agents` directory. The directory to which you install the web agent is referred to as the Policy Agent base directory, or *PolicyAgent-base*. When referred to in this guide, *PolicyAgent-base* represents the full path to the Policy Agent base directory.

For the directory scenario described in the preceding paragraph, the location of *PolicyAgent-base* varies as follows:

- The 64–bit Agent for IIS 6.0 version & the OWA 2007 version:

  These two versions of the agent use the following location for *PolicyAgent-base*:

  ```
  C:\Agents\iis_v6_x64_WINNT_agent\web_agents\iis6_agent
  ```

- The SharePoint 2007 version:

  This version of the agent uses the following location for *PolicyAgent-base*:

  ```
  C:\Agents\iis_v6_WINNT_agent\web_agents\iis6_agent
  ```

- **Unzip the product binaries.**

  `unzip` *binaryname*`.zip`

Note – On Microsoft Windows 2003, the zip file is not automatically unpacked. Therefore, after you download the agents zip file, be sure to extract the zip file to a directory. To extract the zip file, right click on the zip file in the FileManager and select Extract.

Next Steps After you have unzipped the agent binary file to install the agent,, you must create a configuration file for the web site (or web sites) that is to be protected by the agent and then you must configure the agent for that web site (or web sites).

Note – After you create the configuration files, the task that you implement next varies depending on your deployment. At that time, if you are installing Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access, you will skip to "Microsoft Office SharePoint and Outlook Web Access: Configuring the Agent" on page 81. For all other deployments, you will continue directly onto the next task ("Configuring Agent for Microsoft IIS 6.0 for a Web Site" on page 40).

▼ To Create Configuration Files: Agent for Microsoft IIS 6.0

This task applies to all deployments of Agent for Microsoft IIS 6.0, including deployments where the agent protects Microsoft Office SharePoint or Outlook Web Access.

The agent for Microsoft IIS 6.0 provides a Visual Basic (VB) script to help you create agent configuration files. When you run it, the VB script prompts for information related to the Web Site Identifier, the agent you are installing, and Access Manager. The script creates an agent configuration file based on the information you provide.

Note – When you are deploying the agent on multiple web sites, you must create a unique agent configuration file for each of the web sites. Use the following steps to create multiple agent configuration files. However, ensure that you give a unique file name to each of the configuration files.

---

1 **Change to the directory:**

PolicyAgent-base`\bin`

This directory stores the VB script required to create the agent configuration file.

2 **Open a command window as described in the substeps that follow.**

a. **Click Start.**

b. **Select Run.**

c. **Type cmd.**

At this point in the task, the following actions have been performed:

- The license agreement has been provided.
- The agent files have been distributed.

3 **Issue the following command (be aware that the command is case sensitive):**

```
cscript IIS6CreateConfig.vbs defaultConfig
```

`IIS6CreateConfig.vbs` A VB script that saves your responses to prompts about the Microsoft IIS 6.0 host and the Access Manager host in a file. For this example, the file is labeled *defaultConfig*.

defaultConfig The agent configuration file created by this command and for which you provide the actual name. This is a text file to which the output of the commands entered while running the script are written.

Note – Give a unique name for this agent configuration file since you will need the same file to unconfigure the agent.

The script prompts for information as it progresses with the creation of the agent configuration file. All the script prompts are displayed in this step. However, information about the responses are presented in the subsequent steps.

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Copyright c 2004 Sun Microsystems, Inc. All rights reserved
Use is subject to license terms
---------------------------------------------------------
    Microsoft (TM) Internet Information Server (6.0)
---------------------------------------------------------
Enter the Agent Resource File Name [IIS6Resource.en] :

Fully Qualified Host Name :
agentHost.example.com

Displaying the list of Web Sites and its corresponding Identifiers
Site Name (Site Id)
Default Web Site (1)
Site A (1701188044)

Web Site Identifier :
1

Agent Protocol [http] :

Agent Port Number [80] :

Agent Deployment URI [/amagent] :

------------------------------------------------
Sun Java (TM) Enterprise System Access Manager
------------------------------------------------
Primary Server Host :
amHost.example.com

Primary Server Protocol [http] :

Primary Server Port Number [58080] :

Primary Server Deployment URI [/amserver] :

Primary Server Console URI [/amconsole] :

Failover Server Host :
```

```
Agent-Access Manager Shared Secret :

Re-enter Shared Secret :

CDSSO Enabled [false] :

-----------------------------------------------------
Agent Configuration file created ==>  defaultConfig
-----------------------------------------------------
```

4 **When prompted, provide the following information about the Microsoft IIS 6.0 instance that this agent will protect:**

Agent Resource File Name: Accept the default for this prompt (`IIS6Resource.en`).

Host Name: Enter the fully qualified domain name (FQDN) of the system on which Microsoft IIS 6.0 is installed.

For example, if the host is `agentHost`and the domain is `example.com`, then the Host Name in this case is `agentHost.example.com`.

Web Site Identifier:

The Identifier column indicates the unique identifier associated with every web site. The information you enter varies depending on the version of this agent.

Enter the Web Site Identifier for the specific web site for which you are creating a configuration file. Microsoft IIS 6.0 has a unique identifier associated with every web site on the web server. The Web Site Identifier is displayed when you start Microsoft Internet Information Services Manager and click Web Sites. The Identifier column indicates the unique identifier associated with every web site.

- **For Outlook Web Access 2007:**

  Select the number pertaining to the Outlook Web Access application web site. The default is identifier 1.

- **For Microsoft Office SharePoint 2007:**

  Typically, for Microsoft Office SharePoint 2007, the identifier is not identifier 1, but the web site with default port 80.

  For example, if the SharePoint site with port 80 is listed with identifier 747147058: SharePoint - 80 (747147058), enter 747147058 as the Web Site Identifier.

Server Protocol: If this instance of Microsoft IIS 6.0 has been configured for SSL, then select HTTPS; otherwise select HTTP.

Server Port: Enter the port number of the Microsoft IIS 6.0 instance that will be protected by the agent.

Agent Deployment URI: Enter a Universal Resource Identifier (URI) that will be used to access Agent for Microsoft IIS 6.0. The default value is `/amagent`.

Note – The response you provide for this prompt is used by the agent to assign a value to the `com.sun.am.policy.agents.config.agenturi.prefix` property in the web agent `AMAgent.properties` configuration file. This property supports essential functions.

Providing a unique URL for this prompt is important. Therefore, the URL must be different from any other URL used for applications deployed on this Microsoft IIS 6.0 instance.

The URL value should be `http://`*host.domain:port/agent-deployment-uri* where *host*, *domain* and *port* are the FQDN and port number of the Microsoft IIS 6.0 instance where the agent is installed and *agent-deployment-uri* is the value mentioned at the beginning of this description (again, the default value is `amagent`).

The following is an example of an Agent Deployment URI:

```
http://agentHost.example.com:80/amagent
```

where the host name is `agentHost` and the domain name is `example.com`.

---

5 **When prompted, provide the following information about the Access Manager host:**

Primary Server Host: Enter the FQDN of the primary Access Manager host.

For example, if the host is `amHost` and the domain is `example.com`, then the Host Name in this case is `amHost.example.com`.

Primary Server Protocol: If the primary Access Manager host is SSL-enabled, select HTTPS. Otherwise, select HTTP.

Primary Server Port: Enter the port number for the primary Access Manager host.

Primary Server Deployment URI: Enter the location that was specified when Access Manager was installed. The default URI for Access Manager is `/amserver`.

Primary Console Deployment URI: Enter the location that was specified when Access Manager Console was installed. The default URI for Access Manager is `/amconsole`.

Failover Server Host: Enter the FQDN of the secondary Access Manager host if the primary Access Manager host becomes unavailable. If no failover server host exists, then leave this field blank.

Failover Server Port: Enter the port number of the secondary Access Manager host. If no failover server host exists, then leave this field blank.

Failover Server Protocol: If the failover Access Manager host is SSL-enabled, select HTTPS. Otherwise, select HTTP. If no failover server host exists, then leave this field blank.

Failover Server Deployment URI: Enter the location that was specified when Access Manager was installed. The default URI for Access Manager is `/amserver`. If no failover server host exists, then leave this field blank.

Failover Console Deployment URI: Enter the location that was specified when Access Manager Console was installed. The default URI for Access Manager is `/amconsole`. If no failover server host exists, then leave this field blank.

Agent Access Manager Shared Secret: Enter the password for the Access Manager internal LDAP authentication user. This user is also referred to as `amldapuser`.

For more information about the shared secret and its relationship with the Access Manager agent profile, see Chapter 4, "The Relationship Between the Agent Profile and Web Agents in Policy Agent 2.2."

Re-enter Shared Secret: Re-enter the password for the Access Manager internal LDAP authentication user (`amldapuser`).

CDSSO Enabled: Check this box if you want to enable CDSSO.

With the information you provide, the script creates the agent configuration file for you to use to configure this agent as described in the following section.

Next Steps At this point, the next task to be implemented varies depending on your deployment. If you are installing Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access, skip to "Microsoft Office SharePoint and Outlook Web Access: Configuring the Agent" on page 81. For all other deployments, continue with the task that follows (Configuring Agent for Microsoft IIS 6.0 for a Web Site).

Configuring Agent for Microsoft IIS 6.0 for a Web Site

Caution – Do not perform the task described in this section if you are installing Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access. Instead continue the configuration process by implementing the steps in "Microsoft Office SharePoint and Outlook Web Access: Configuring the Agent" on page 81. For all other deployments, continue with this task.

Configure Agent for Microsoft IIS 6.0 for a web site after you have created an agent configuration file. If you have not already installed the agent, install it as explained in "Installing Agent for Microsoft IIS 6.0" on page 34.

Note – If you want to configure the agent for multiple web sites, you must create a separate agent configuration file for each web site.

To configure the agent for a web site, follow these steps:

▼ To Configure Agent for Microsoft IIS 6.0 for a Web Site

1 **Change to the directory:**

PolicyAgent-base`\bin`

2 **Issue the following command (be aware that the command is case sensitive):**

`cscript IIS6admin.vbs -config` *defaultConfig*

`IIS6admin.vbs` A VB script that uses the output of the `IIS6CreateConfig.vbs` script. The output was saved to a configuration file, which for this example labeled *defaultConfig*.

`-config` The option that allows the output to be used to configure the web site to use Agent for Microsoft IIS 6.0.

defaultConfig A place holder for the name of the agent configuration file created previously as described in "To Create Configuration Files: Agent for Microsoft IIS 6.0" on page 35.

The script displays messages to indicate the progress of the configuration as shown in the following sample.

```
Microsoft (R) Windows Script Host Version 5.6
    Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

    Copyright c 2004 Sun Microsystems, Inc. All rights reserved
    Use is subject to license terms

    Enter the Agent Resource File Name [IIS6Resource.en] :

    Creating the Agent Config Directory
    Creating the AMAgent.properties File
    Updating the Windows Product Registry
    Loading the IIS 6.0 Agent
    Completed Configuring the IIS 6.0 Agent
```

3 **Ensure that the authentication method of Microsoft IIS 6.0 Server is set to Anonymous.**

4 **Restart the application pool to which the web site belongs.**

5 **Restart the web site.**

6 **Try accessing the web site (`http://`*fqdn:port*`/index.html`).**

This link should take you to the Access Manager login page. After a successful authentication, if the policy is properly defined, you should be able to view the resource.

If you want to view the agent log file `amAgent`, do so at the following location:

PolicyAgent-base`\debug\Identifier_`*site-identifier*

where *site-identifier* is a number, such as 1, that represents the identifier of the web site for which the agent is being configured.

Note – If you want to configure the agent for multiple web sites, you must follow the above steps for each of the web sites.

Next Steps The last step of this task addresses verification of the agent installation. See, the section that follows (Verifying a Successful Installation of Policy Agent 2.2) for an expanded explanation on verifying the agent installation.

Verifying a Successful Installation of Policy Agent 2.2

Note – The task presented in this section about verifying the agent installation does not apply to deployments where the agent protects Microsoft Office SharePoint or Outlook Web Access. To verify agent installation for those types of installations, see "Microsoft Office SharePoint and Outlook Web Access: Verifying a Successful Agent Installation" on page 91.

After installing a web agent, ensure that the agent is installed successfully. Two methods are available for verifying a successful web agent installation. Perform both for best results.

▼ To Verify a Successful Installation

1. **Attempt to access a resource on the deployment container where the agent is installed.**

   If the web agent is installed correctly, accessing any resource should take you to the Access Manager login page. After a successful authentication, if the policy is properly defined, you should be able to view the resource.

2. **Check the web agent `AMAgent.properties` configuration file.**

   Make sure that each property is set properly. For information on the properties in this file, see Appendix C, "Web Agent `AMAgent.properties` Configuration File."

◆ ◆ ◆ CHAPTER 4

The Relationship Between the Agent Profile and Web Agents in Policy Agent 2.2

This section describes how to create or update an agent profile in Access Manager Console and then how to make the corresponding changes in the web agent.

If you are only interested in resetting the shared secret in the web agent, not the agent profile name, see "Resetting the Shared Secret Password" on page 66. However, first read the introductory paragraphs that follow in this section to become acquainted with the process and terminology related to the credentials used by web agents to authenticate with Access Manager. A common reason to reset only the shared secret is that it was entered incorrectly when prompted for during the installation of the web agent.

A web agent uses a user name and password as credentials to authenticate with Access Manager. You can use the default values for these credentials or you can create an agent profile in Access Manager Console and use those credentials. In web agents, the term for the default user name is agent user name. The default value of the agent user name is `UrlAccessAgent`. The term for the default password is shared secret. The default value of the shared secret is the password of the Access Manager internal LDAP authentication user. This user is commonly referred to as `amldapuser`.

Creating an agent profile is not a requirement for web agents. You can use the default values and never change the agent user name or shared secret. However, in certain situations you might want to change these default values. Changing the default values of the agent user name and shared secret involves creating an agent profile using Access Manager Console.

The terms used for the credentials are different once you create them in the agent profile. Agent user name is then called agent profile name. Shared secret is then called agent profile password. After you create the agent profile, you must assign the values of the agent profile name and the agent profile password to the correct properties in the web agent `AMAgent.properties` configuration file.

Creating or Updating a Web Agent Profile

The instructions that follow in this section explain how to change both the agent profile name and the agent profile password on the Access Manager side.

Since the agent profile is created and updated in Access Manager Console, tasks related to the agent profile are discussed in Access Manager documentation. Nonetheless, tasks related to the agent profile are also described in this Policy Agent guide, specifically in this chapter. For related information about defining the Policy Agent profile in Access Manager Console, see the following section of the respective document: "Agents Profile" in *Sun Java System Access Manager 7.1 Administration Guide*.

▼ To Create or Update an Agent Profile in Access Manager

Perform the following tasks in Access Manager Console. The key steps of this task involve creating an agent ID (agent profile name) and an agent profile password.

1. **With the Access Control tab selected click the name of the realm for which you would like to create an agent profile.**

2. **Select the Subjects tab.**

3. **Select the Agent tab.**

4. **Click New.**

5. **Enter values for the following fields:**

   ID. Enter the agent profile name or identity of the agent.

   This is the agent profile name, which is the name the agent uses to log into Access Manager. Multi-byte names are not accepted. Do not use the web agent default value of `UrlAccessAgent`.

   Password. Enter the agent profile password.

   Do not use the web agent default value of this password. The web agent default value of this password is the password of the internal LDAP authentication user, commonly referred to as `amldapuser`.

   Password (confirm). Confirm the password.

   Device Status. Select the device status of the agent. The default status is Active. If set to Active, the agent will be able to authenticate to and communicate with Access Manager. If set to Inactive, the agent will not be able to authenticate to Access Manager.

6 **Click Create.**

The list of agents appears.

7 **(Optional) If you desire, add a description to your newly created agent profile:**

a. **Click the name of your newly created agent profile from the agent list.**

b. **In the Description field, enter a brief description of the agent.**

For example, you can enter the agent instance name or the name of the application it is protecting.

c. **Click Save.**

Updating the Agent Profile Name and the Agent Profile Password in Web Agents

After you have changed the agent profile in Access Manager Console, assign the values for the agent profile name and the agent profile password to the corresponding properties in the web agent `AMAgent.properties` configuration file. This process involves the following:

- Adding the agent profile name to the following property in the web agent `AMAgent.properties` configuration file: `com.sun.am.policy.am.username`
- Encrypting the agent profile password (shared secret) using the encryption utility
- Adding the encrypted agent profile password (shared secret) to the following property in the web agent `AMAgent.properties` configuration file: `com.sun.am.policy.am.password`

The procedures specified in the preceding list are detailed in the task description that follows.

▼ To Update the Agent Profile Name and Agent Profile Password

1 **Update the following property in the web agent `AMAgent.properties` configuration file:**

```
com.sun.am.policy.am.username
```

Replace the value of this property with the agent profile name you just updated in Access Manager Console.

2 **Go to the following directory:**

PolicyAgent-base`\bin`

3 **Execute the following script from the command line**

`cryptit` *agent-profile-password*

where *agent-profile-password* represents the agent profile password you just updated in Access Manager Console.

4 **Copy the output obtained after issuing the `cryptit` *agent-profile-password* command and paste it as the value for the following property:**

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

5 **Restart the deployment container and try accessing any resource protected by the agent.**

If the agent gets redirected to Access Manager, this indicates the above steps were executed properly.

◆ ◆ ◆ CHAPTER 5

Post-Installation Configuration: Policy Agent 2.2 for Microsoft IIS 6.0

The tasks described in this chapter are not required for this web agent to work, but might be desired.

After completing the tasks described in this chapter, perform the tasks to configure the web agent to your site's specific needs as explained in Chapter 6, "Managing Policy Agent 2.2 for Microsoft IIS 6.0."

Setting Up SSL With Agent for Microsoft IIS 6.0

Perform the tasks described in this chapter if you want to configure SSL with Agent for Microsoft IIS 6.0.

During installation, if you choose the HTTPS protocol, the agent for Microsoft IIS 6.0 is automatically configured and ready to communicate over SSL. Before proceeding with the tasks in this section, ensure that the Microsoft IIS 6.0 instance is configured for SSL.

⚠ **Caution –** You should have a solid understanding of SSL concepts and the security certificates required to enable communication over the HTTPS protocol. See the documentation that comes with Microsoft IIS 6.0.

▼ To Configure Notification on Agent for Microsoft IIS 6.0 for SSL

If Microsoft IIS 6.0 is running in SSL mode and is receiving notifications, first perform the following broadly defined steps:

1 **Add the Microsoft IIS 6.0 certificate's root CA certificate to the Access Manager's certificate database.**

2 **Mark the CA root certificate as trusted to enable Access Manager to successfully send notifications to the agent for Microsoft IIS 6.0.**

Default Trust Behavior of Agent for Microsoft IIS 6.0

This section only applies when Access Manager itself is running SSL. By default, Agent for Microsoft IIS 6.0 trusts any server certificate presented over SSL by the Access Manager host. The web agent does not check the root Certificate Authority (CA) certificate. If the Access Manager host is SSL-enabled and you want the agent to perform certificate checking, adhere to the guidelines as described in the following subsections:

- "Disabling the Default Trust Behavior of Agent for Microsoft IIS 6.0" on page 50
- "Installing the Access Manager Root CA Certificate on Microsoft IIS 6.0" on page 50

Disabling the Default Trust Behavior of Agent for Microsoft IIS 6.0

The following property exists in the web agent `AMAgent.properties` configuration file, and by default it is set to true:

```
com.sun.am.trust_server_certs
```

With this property set to true, the web agent does not perform certificate checking. Enabling the web agent to perform certificate checking is a one-step process that only involves setting this property to `false` as illustrated in the following task.

▼ To Disable the Default Trust Behavior of Agent for Microsoft IIS 6.0

● **Set the following property in the web agent** `AMAgent.properties` **configuration file to** `false` **as follows:**

```
com.sun.am.trust_server_certs = false
```

Installing the Access Manager Root CA Certificate on Microsoft IIS 6.0

The root CA certificate that you install on the Microsoft IIS 6.0 instance that the agent protects must be the same one that is installed on the Access Manager host.

▼ To Install the Access Manager Root CA Certificate on Microsoft IIS 6.0

1 **(Conditional) If the certificate database has not yet been created, create it at a unique location using a command such as the following:**

PolicyAgent-base`\bin\certutil -N -d .`

2 Install the root CA certificate.

Remember that the root CA certificate that you install on the Microsoft IIS 6.0 server must be the same certificate that is installed on the Access Manager host.

The following example demonstrates a command you can issue that uses the `certutil` utility to install the certificate:

```
PolicyAgent-base\bin\certutil -A -n cert-name -t
"C,C,C" -d cert-dir -i cert-file
```

cert-name represents the name of this root CA certificate

cert-dir represents the directory where the certificate and key stores are located.

cert-file represents the base-64 encoded root CA certificate file.

For more information on the `certutil` utility, see the online help by issuing the following command:

```
certutil -H
```

3 To verify that the certificate is properly installed, in the command line, issue the following command:

```
PolicyAgent-base\bin\certutil -L -d cert-dir
```

The root CA certificate is then listed in the output of the `certutil -L` command as illustrated in the following code example:

```
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
```

4 Restart the Microsoft IIS 6.0 server.

CHAPTER 6

Managing Policy Agent 2.2 for Microsoft IIS 6.0

After installation of a web agent, management of Policy Agent 2.2 for Microsoft IIS 6.0 is mostly performed by editing the web agent `AMAgent.properties` configuration file.

The following section provides details of how to perform various tasks by interacting with the web agent `AMAgent.properties` configuration file.

Key Features and Tasks Performed with the Web Agent `AMAgent.properties` Configuration File

The web agent `AMAgent.properties` configuration file is a text file of configuration properties that you can modify to change web agent behavior. However, the content of this file is very sensitive. Changes made can result in changes in how the agent works. Errors made can cause the agent to malfunction.

This section describes the most important details of the configuration file, such as how specific properties can be modified to produce specific results. The topics described are typically those of greatest interest in real-world deployment scenarios. For a list and description of every property in the configuration file, access the configuration file itself located as described in Table 6–1. Also a list of the properties is available in this guide, at Appendix C, "Web Agent `AMAgent.properties` Configuration File."

This section describes the following:

- "Locating the Web Agent `AMAgent.properties` Configuration File" on page 54
- "Using the Web Agent `AMAgent.properties` Configuration File" on page 54
- "Providing Failover Protection for a Web Agent" on page 55
- "Changing the Web Agent Caching Behavior" on page 56
- "Configuring the Not-Enforced URL List" on page 57
- "Configuring the Not-Enforced IP Address List" on page 58
- "Enforcing Authentication Only" on page 58

Locating the Web Agent `AMAgent.properties` Configuration File

The following table provides the default location for the web agent `AMAgent.properties` configuration file.

TABLE 6–1 Location of the Web Agent `AMAgent.properties` Configuration File

Server	Platform	Location
Microsoft Internet Information Services 6.0 (Microsoft IIS 6.0)	Windows	*PolicyAgent-base*\ `Identifier_`*site-identifier*`\config`

where *site-identifier* is a number, such as 1, that represents the identifier of the web site for which the agent is being configured.

Using the Web Agent `AMAgent.properties` Configuration File

Changing the web agent `AMAgent.properties` configuration file can have serious and far-reaching effects. When you make changes, keep the following in mind:

- Make a backup copy of this file before you make changes.
- Trailing spaces are significant; use them judiciously.
- Use a forward slash (`/`) to separate directories, not a backslash (`\`). Perhaps unexpected, but this applies to Windows systems.
- Spaces in the Windows file names are allowed.

Note – If you make changes to the web agent AMAgent.properties configuration file, restart the deployment container to make your changes take effect.

The web agent AMAgent.properties configuration file includes information for a variety of configurations, including the following:

- debugging
- fully qualified domain name (FQDN) map
- Access Manager services
- service and agent deployment descriptors
- session failover

The configuration file also contains configuration information on advanced features, such as forwarding LDAP user attributes through HTTP headers and POST data preservation.

Providing Failover Protection for a Web Agent

When you install a web agent, you can specify a *failover* or backup deployment container, such as a web server, for running Access Manager. This is essentially a high availability option. It ensures that if the deployment container that runs Access Manager service becomes unavailable, the web agent still processes access requests through a secondary, or failover, deployment container running Access Manager service.

Setting up failover protection for the web agent, requires modifying the web agent AMAgent.properties configuration file. However, you must first install two different instances of Access Manager on two separate deployment containers.

Then follow the instructions in this guide to about installing the web agent. The web agent installation program prompts you for the host name and port number of the failover deployment container that you have configured to work with Access Manager. The following property in the web agent AMAgent.properties configuration file, stores the failover deployment container name:

```
com.sun.am.policy.am.login.url
```

Set this property in order to store failover server information. Given the values in the following list, the property would be set as shown in Example 6–1.

amHost1 Name of the primary Access Manager host.

amHost2 Name of the first failoverAccess Manager host.

amHost3 Name of the second failoverAccess Manager host.

example Name of the domain.

58080	Default port number

EXAMPLE 6–1 Configuration Property Setting for Failover Protection of a Web Agent

```
com.sun.am.policy.am.login.url = http://amHost1.example.com:58080/
amserver/UI/Login http://amHost2.example.com:58080/amserver/UI/Login
http://amHost3.example.com:58080/amserver/UI/Login
```

A failover server name is configurable after it has been set during installation. When configuring this property, note that a space is required between each Access Manager login URL.

Changing the Web Agent Caching Behavior

Each web agent maintains a cache that stores the policies for every user's session. The cache can be updated by a cache polling mechanism and a cache notification mechanism.

Cache Updates

A web agent maintains a cache of all active sessions involving content that the agent protects. Once an entry is added to an agent's cache, it remains valid for a period of time after which the entry is considered expired and later purged.

The property `com.sun.am.policy.am.polling.interval` in the web agent `AMAgent.properties` configuration file determines the number of minutes an entry will remain in the web agent cache. Once the interval specified by this property has elapsed, the entry is dropped from the cache. By default, the expiration time is set to three minutes.

Hybrid Cache Updates

In this mode, cache entry expiration still applies. In addition, the web agent gets notified by the Access Manager service about session changes. Session changes include events such as session logout or a session timeout. When notified of a session or a policy change, the web agent updates the corresponding entry in the cache. Apart from session updates, web agents can also receive policy change updates. Policy changes include events such as updating, deleting, and creating policies.

Web agents have the hybrid cache update mode switched on by default. This is triggered by the property `com.sun.am.notification.enable` in the web agent `AMAgent.properties` configuration file, which is set to `true`. When the property is set to `false`, the web agent updates its cache through the cache polling mechanism only.

Restrictions due to firewalls, as well as the type of deployment container in use, might not allow notifications to work. In such cases, notification is turned off.

The web agent sets a timeout period on its cache entries. After its end of life, the cache entry is purged from the web agent's cache. The web agent does not refetch the cache data. The next attempt to access the same entry from cache fails and the web agent makes a round trip to the server and fetches it again to populate the cache. This lazy method of cache updating keeps the web agent cache performing optimally and reduces network traffic.

In a normal deployment situation, policy changes on the server are frequent, which requires sites to accept a certain amount of latency for web agents to reflect policy changes. Each site decides the amount of latency time that is acceptable for the site's specific needs. When setting the `com.sun.am.policy.am.polling.interval` property, set it to the lower of the two:

- The session idle timeout period
- Your site's accepted latency time for policy changes

Configuring the Not-Enforced URL List

The *not-enforced URL list* defines the resources that should not have any policies (neither allow nor deny) associated with them.

By default, the web agent denies access to all resources on the deployment container that it protects. However, various resources (such as a web site or an application) available through a deployment container might not need to have any policy enforced. Common examples of such resources include the HTML pages and `.gif` images found in the home pages of web sites and the cascading style sheets (CSS) that apply to these home pages. The user should be able to browse such pages without authenticating. For the home page example, all these resources need to be on the not-enforced URL list or the page will not be displayed properly. The property `com.sun.am.policy.agents.config.notenforced_list` is used for this purpose. Wild cards can be used to define a pattern of URLs. Space is the separator between the URLs mentioned in the list.

There can be a reverse, or "inverted", scenario when all the resources on the deployment container, except a list of URLs, are open to any user. In that case, the property `com.sun.am.policy.agents.config.notenforced_list.invert` would be used to reverse the meaning of `com.sun.am.policy.agents.config.notenforced_list`. If it is set to `true` (by default it is set to `false`), then the not-enforced URL list would become the enforced list.

EXAMPLE 6–2 Configuration Property Settings for Not-Enforced URL List

The following are examples:

Scenario 1: Not-Enforced URL List

EXAMPLE 6–2 Configuration Property Settings for Not-Enforced URL List *(Continued)*

```
com.sun.am.policy.agents.config.notenforced_list.invert = false
```

```
com.sun.am.policy.agents.config.notenforced_list =
http://agentHost.example.com:80/welcome.html
http://agentHost.example.com:80/banner.html
```

In this case, authentication and policies will not be enforced on the two URLs listed in the notenforcedList. All other resources will be protected by the web agent.

Scenario 2: Inverted Not-Enforced URL List

```
com.sun.am.policy.agents.config.notenforced_list.invert = true
```

```
com.sun.am.policy.agents.config.notenforced_list =
 http://agentHost.example.com:80/welcome.html
 http://agentHost.example.com:80/banner.html
```

In this case, authentication and policies will be enforced by the web agent on the two URLs mentioned in the notenforcedList. All other resources will be accessible to any user.

Caution – If feasible, keep this property set to false as such:

```
com.sun.am.policy.agents.config.notenforced_list.invert = false
```

A value of false reduces the chance of unintentionally allowing access to resources.

Configuring the Not-Enforced IP Address List

The com.sun.am.policy.agents.config.notenforced_client_ip_list property is used to specify a list of IP addresses. No authentication is required for the requests coming from these client IP addresses.

In other words, the web agent will not enforce policies for the requests originating from the IP addresses in the Not-Enforced IP Address list.

Enforcing Authentication Only

The property com.sun.am.policy.agents.config.do_sso_only is used to specify if only authentication is enforced for URLs protected by the web agent. If this property is set to true (by default it is set to false), it indicates that the web agent enforces authentication only, without enforcing policies. After a user logs onto Access Manager successfully, the web agent will not check for policies related to the user and the accessed URLs.

Providing Personalization Capabilities

Web agents in Policy Agent 2.2 can personalize page content for users in three distinct ways as described in the following subsections:

- "Providing Personalization With Session Attributes" on page 59
- "Providing Personalization With Policy-Based Response Attributes" on page 60
- "Providing Personalization With User Profile Attributes Globally" on page 61

Providing Personalization With Session Attributes

Web agents in Policy Agent 2.2 support a feature where a user's session attributes are fetched and set as headers or cookies. The following property responsible for this task:

```
com.sun.am.policy.agents.config.session.attribute.fetch.mode
```

This property can be set to one of the following values:

- NONE
- HTTP_HEADER
- HTTP_COOKIE

When set to NONE, no session attributes are fetched and the com.sun.am.policy.agents.config.session.attribute.map property is ignored. With this property set to either HTTP_HEADER or HTTP_COOKIE, the web agent fetches session attributes. Use the following property to configure attributes that are to be forwarded as HTTP headers or cookies: com.sun.am.policy.agents.config.session.attribute.map.

The following content is from the web agent AMAgent.properties configuration file. The text has been reformatted for this section. This section illustrates how the com.sun.am.policy.agents.config.session.attribute.map property maps session attributes to headers or cookies.

Session attributes are added to an HTTP header following this format:

```
session_attribute_name|http_header_name[,...]
```

The value of the attribute being fetched in session is session_attribute_name. This value gets mapped to a header value as follows: http_header_name.

Note – In most cases, in a destination application where http_header_name appears as a request header, it is prefixed with HTTP_ and the following type of conversion takes place:

Lower case letters	convert to upper case letters.
Hyphen "-"	converts to underscore "_"
"common-name"	as an example, converts to "HTTP_COMMON_NAME."

```
com.sun.am.policy.agents.config.session.attribute.map =
successURL | success-url, contextId | context-id
```

The session attribute is forwarded as a header or a cookie as determined by the end-user applications on the web container that the web agent is protecting. These applications can be considered the consumers of the forwarded header values. The forwarded information is used for the customization and personalization of web pages. You can also write server side plug-ins to put any user session attribute and define the corresponding attribute name and mapping in the preceding property to retrieve the value.

Providing Personalization With Policy-Based Response Attributes

Header attributes can also be determined by Access Manager policy configurations. With policy-based response attributes you can define attribute-value pairs at each policy.

Web agents in this release set policy-based response attributes as headers or cookies based on configuration. All subjects that match this attribute set obtain this attribute.

The following is a new property that has been added to the web agent AMAgent.properties configuration file to control this functionality:

```
com.sun.am.policy.agents.config.response.attribute.fetch.mode
```

This property can be set to one of the following values:

- NONE
- HTTP_HEADER
- HTTP_COOKIE

The following example shows this configuration property with the default setting, which is HTTP_HEADER:

```
com.sun.am.policy.agents.config.response.attribute.fetch.mode = HTTP_HEADER
```

Attribute mapping is available for response attributes. Therefore, the format of policy information can be mapped to the format of a header or a cookie. The below property is used for this type of mapping:

```
com.sun.am.policy.agents.config.response.attribute.map
```

Unlike profile attributes and session attributes, where only the mapped attributes are displayed as headers or cookies, by default, response attributes are set by the agent as headers or cookies based on the setting of this property:

```
com.sun.am.policy.agents.config.response.attribute.fetch.mode
```

If a response attribute map is specified, then the corresponding attribute mapped name is fetched from the map and its corresponding value is displayed as either a header or a cookie based on the setting of the above property.

Providing Personalization With User Profile Attributes Globally

Web agents in Policy Agent 2.2 have the ability to forward user profile attribute values via HTTP headers to end-web applications. The user profile attribute values come from the server side of Access Manager. The web agent behaves like a broker to obtain and relay user attribute values to the destination servlets, CGI scripts, or ASP pages. These applications can in turn use the attribute values to personalize page content.

This feature is configurable through two properties in the web agent AMAgent.properties configuration file. To turn this feature on and off, use the following property from the web agent AMAgent.properties configuration file:

```
com.sun.am.policy.agents.config.profile.attribute.fetch.mode
```

This property can be set to one of the following values:

- NONE
- HTTP_HEADER
- HTTP_COOKIE

When set to NONE, the web agent does not fetch LDAP attributes from the server and ignores the com.sun.am.policy.agents.config.profile.attribute.map property. In the other two cases, the web agent fetches the attribute.

To configure the attributes that are to be forwarded in the HTTP headers, use the following property:

```
com.sun.am.policy.agents.config.profile.attribute.map
```

Below is an example section from the web agent AMAgent.properties configuration file, which shows how this feature is used:

```
#
# 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.agents.config.profile.attribute.map = cn|common-name,ou|
organizational-unit,
o|organization,mail|email,employeenumber|employee-number,c|country
```

By default, some LDAP user attribute names and HTTP header names are set to sample values.

To find the appropriate LDAP user attribute names, check the following XML file on the machine where Access Manager is installed:

AccessManager-base/SUNWam/config/xml/amUser.xml

The attributes in this file could be either Access Manager user attributes or Access Manager dynamic attributes. For an explanation of these two types of user attributes, see *Sun Java System Access Manager 7.1 Administration Guide*.

The attribute and HTTP header names that need to be forwarded must be determined by the end-user applications on the deployment container that the web agent is protecting. Basically, these applications are the consumers of the forwarded header values (the forwarded information is used for the customization and personalization of web pages).

Setting the Fully Qualified Domain Name

To ensure appropriate user experience, it is necessary that the users access resources protected by the web agent using valid URLs. The configuration property com.sun.am.policy.agents.config.fqdn.default provides the necessary information needed by the web agent to identify if the user is using a valid URL to access the protected resource. If the web agent determines that the incoming request does not have a valid hostname in the URL, it redirects the user to the corresponding URL with a valid hostname. The difference between the redirect URL and the URL originally used by the user is only the hostname, which is changed by the web agent to a fully qualified domain name (FQDN) as per the value specified in this property.

This is a required configuration property without which the deployment container may not start up correctly. This property is set during the web agent installation and must not be modified unless absolutely necessary to accommodate deployment requirements. An invalid value for this property can result in the deployment container becoming unusable or the resources becoming inaccessible.

The property `com.sun.am.policy.agents.config.fqdn.map` provides another way by which the web agent can resolve partial or malformed access URLs and take corrective action. The web agent gives precedence to the entries defined in this property over the value defined in the `com.sun.am.policy.agents.config.fqdn.default` property. If none of the entries in this property matches the hostname specified in the user request, the agent uses the value specified for `com.sun.am.policy.agents.config.fqdn.default` property.

The `com.sun.am.policy.agents.config.fqdn.map` property can be used for creating a mapping for more than one hostname. This may be the case when the deployment container protected by this agent is accessible by more than one hostname. However, this feature must be used with caution as it can lead to the deployment container resources becoming inaccessible.

This property can also be used to override the behavior of the web agent in cases where necessary. The format for specifying the property `com.sun.am.policy.agents.config.fqdn.map` is:

```
com.sun.am.policy.agents.config.fqdn.map =
[invalid_hostname|valid_hostname][,...]
```

where:

`invalid_hostname` is a possible invalid hostname such as partial hostname or an IP address that the user may provide .

`valid_hostname` is the corresponding valid hostname that is fully qualified. For example, the following is a possible value specified for hostname `xyz.domain1.com`:

```
com.sun.am.policy.agents.config.fqdn.map = xyz|xyz.domain1.com,
xyz.domain1|xyz.domain1.com
```

This value maps `xyz` and `xyz.domain1` to the FQDN `xyz.domain1.com`.

This property can also be used in such a way that the web agent uses the name specified in this map instead of the deployment container's actual name.

If you want your server to be addressed as *xyz.hostname*.com whereas the actual name of the server is *abc.hostname*.com. The browser only knows *xyz.hostname*.com and you have specified policies using *xyz.hostname*.com in the Access Manager Console. In this file, set the mapping as com.sun.am.policy.agents.config.fqdn.map = valid|*xyz.hostname*.com.

Resetting Cookies

The cookie reset feature enables the web agent to reset some cookies in the browser session while redirecting to Access Manager for authentication.

This feature is configurable through two properties in the web agent AMAgent.properties configuration file.

- Enable Cookie Reset

  `com.sun.am.policy.agents.config.cookie.reset.enable = true`

  This property must be set to true if this web agent needs to reset cookies in the response while redirecting to Access Manager for authentication. By default, this is set to false.
- Cookie List

  This property gives the comma-separated list of cookies that need to be reset in the response while redirecting to Access Manager for authentication. This property is used only if the Cookie Reset feature is enabled.

  Cookie details must be specified in the following format:

  ```
  name[=value][;Domain=value]
  ```

  For example,

  ```
  com.sun.am.policy.agents.config.cookie.reset.list = LtpaToken, cookie1=value1,
  cookie2=value2;Domain=example.com
  ```

Configuring CDSSO

The cross domain single sign-on (CDSSO) feature is configurable through three properties in the web agent AMAgent.properties configuration file. To turn this feature on or off, use the following property:

```
com.sun.am.policy.agents.config.cdsso.enable = true
```

By default, this property is set to false, and the feature is turned off. To turn on CDSSO, set this property to true.

Set the URL where CDC controller is installed by specifying the URL in the following property:

```
com.sun.am.policy.agents.config.cdcservlet.url
```

The following is an example of how this property could be set:

```
com.sun.am.policy.agents.config.cdcservlet.url =
http://amHost.example.com:58080/amserver/cdcservlet
```

The third property, `com.sun.am.policy.agents.config.cookie.domain.list` allows you to specify a list of domains in which cookies have to be set in a CDSSO scenario. This property is used only if CDSSO is enabled. If you leave this property blank, then the fully qualified cookie domain for the web agent server will be used for setting the cookie domain. In such a case, it is a host cookie and not a domain cookie.

For more information on CDSSO, see *Sun Java System Access Manager 7.1 Technical Overview*

Setting the REMOTE_USER Server Variable

The property `com.sun.am.policy.am.userid.param` allows you to configure the user ID parameter passed by the session or user profile information from Access Manager. The user ID value is used by the agent to set the value of the `REMOTE_USER` server variable. By default, this parameter is set to `UserToken` and is fetched from session attributes.

It can be set to any other session attribute. Another property determines where to retrieve the value, from user profiles or from session properties.

Example 1: This example demonstrates how to set the user ID parameter with session attributes:

`com.sun.am.policy.am.userid.param.type=SESSION` (this is default)

`com.sun.am.policy.am.userid.param=UserToken` (UserId, Principal, or any other session attribute)

Example 2: This example demonstrates how to set the user ID parameter with LDAP user profile attributes:

`com.sun.am.policy.am.userid.param.type=LDAP`

`com.sun.am.policy.am.userid.param=cn` (any profile attribute)

Setting Anonymous User

For resources on the not-enforced list, the default configuration does not allow the `REMOTE_USER` variable to be set. To enable the `REMOTE_USER` variable to be set for not-enforced

URLs, you must set the following property in the web agent AMAgent.properties configuration file to TRUE (by default the value is FALSE):

```
com.sun.am.policy.agents.config.anonymous_user.enable = TRUE
```

When you set the value of this property to TRUE, the value of REMOTE_USER will be set to the value contained in the following property in the web agent AMAgent.properties configuration file:

```
com.sun.am.policy.agents.config.anonymous_user
```

By default, the value of this property is set to anonymous as follows:

```
com.sun.am.policy.agents.config.anonymous_user = anonymous
```

Validating Client IP Addresses

This feature can be used to enhance security by preventing the stealing or *hijacking* of SSO tokens.

The web agent AMAgent.properties configuration file contains a property titled com.sun.am.policy.agents.config.client_ip_validation.enable, which by default, is set to false.

If you set this property value to true, client IP address validation will be enabled for each incoming request that contains an SSO token. If the IP address from which the request was generated does not match the IP address issued for the SSO token, the request will be denied. This is essentially the same as enforcing a deny policy.

This feature should not be used, however, if the client browser uses a web proxy or if there is a load balancer somewhere between the client browser and the agent-protected deployment container. In such cases, the IP address appearing in the request will not reflect the real IP address on which the client browser runs.

Resetting the Shared Secret Password

This section describes how to reset the shared secret. The web agent stores the shared secret in the web agent AMAgent.properties configuration file.

If you are only interested in resetting the shared secret, not the agent profile name, continue reading this section. If you are interested in creating or updating the agent profile in Access Manager Console and then updating the same credential information in the web agent, see

Chapter 4, "The Relationship Between the Agent Profile and Web Agents in Policy Agent 2.2." The steps described in that chapter are comprehensive, integrating the simpler steps described in this section.

The chapter mentioned in the preceding paragraph also provides a useful explanation of the process and terminology related to the credentials used by web agents to authenticate with Access Manager. Refer to that chapter for more information.

This section specifically describes how to change the shared secret in web agents. The following situations might require you to reset the shared secret:

- You entered the shared secret incorrectly during web agent installation.
- You have been using the default shared secret, which is the `amldapuser` password, but this password has since been changed.

The value for the property `com.sun.am.policy.am.password` in the web agent `AMAgent.properties` configuration file is set with the encrypted shared secret during web agent installation. Therefore, if the shared secret is entered incorrectly during installation, the preceding property is assigned an incorrect value, preventing the web agent from authenticating with Access Manager.

To reset or change the shared secret, use the encryption utility to encrypt the shared secret and then set the value in the property as explained in the following task description.

▼ To Reset the Shared Secret

1 **Go to the following directory:**

PolicyAgent-base`\bin`

2 **Execute the following script from the command line**

`cryptit` *shared-secret*

where *shared-secret* represents the password, that along with the agent user name, allows the web agent to authenticate with Access Manager. The default value of the shared secret is the password of the Access Manager internal LDAP authentication user. This user is commonly referred to as `amldapuser`.

3 **Copy the output obtained after issuing the `cryptit` *shared-secret* command and paste it as the value for the following property:**

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

4 **Restart the deployment container and try accessing any resource protected by the agent.**

If the agent gets redirected to Access Manager, this indicates the above steps were executed properly.

Enabling Load Balancing

Various properties in the web agent AMAgent.properties configuration file can be used to enable load balancing. Edit the properties that apply, according to the location of the load balancer or load balancers in your deployment, as follows:

- "Load Balancer in Front of Access Manager" on page 68
- "Load Balancer in Front of Web Agent" on page 68
- "Load Balancers in Front of Both the Web Agent and Access Manager" on page 69

Load Balancer in Front of Access Manager

When a load balancer is deployed in front of Access Manager and a web agent interacts with the load balancer, the following properties must be edited:

```
com.sun.am.naming.url
com.sun.am.policy.am.login.url
com.sun.am.load_balancer.enable
```

EXAMPLE 6–3 Property Settings: Load Balancer in Front of Access Manager

This example illustrates property settings in the web agent AMAgent.properties configuration file that can be used to enable load balancing:

```
com.sun.am.naming.url = LB-url/amserver/namingservice
com.sun.am.policy.am.login.url = LB-url/amserver/UI/Login
com.sun.am.load_balancer.enable = true
```

where *LB-url* represents the load balancer URL. The following example is a conceivable load balancer URL:

```
http://hostname.example.com:8080
```

Load Balancer in Front of Web Agent

In many cases, when a load balancer is deployed in front of the web agent only the following property must be set:

```
com.sun.am.policy.agents.fqdnMap
```

EXAMPLE 6–4 Property Settings: Load Balancer in Front of Web Agent

```
com.sun.am.policy.agents.fqdnMap = valid|LB-hostname
```

where *LB-hostname* represents the name of the machine on which the load balancer is located.

However, if SSL-termination or a proxy server is used in the deployment, all the following properties in the web agent `AMAgent.properties` configuration file should be set in addition to the preceding property:

```
com.sun.am.policy.agents.config.override_protocol
com.sun.am.policy.agents.config.override_host
com.sun.am.policy.agents.config.override_port
com.sun.am.policy.agents.config.agenturi.prefix
```

This example illustrates how properties can be set to enable load balancing when the protocol, hostname, and port number of the load balancer differ from that of the web agent. However, if the load balancer and the web agent share one of these characteristics, such as the protocol or hostname, then the respective property would be left blank instead of being assigned a value of *true*.

```
com.sun.am.policy.agents.config.override_protocol = true
com.sun.am.policy.agents.config.override_host = true
com.sun.am.policy.agents.config.override_port = true
com.sun.am.policy.agents.config.agenturi.prefix = LB-url/amagent
```

where *LB-url* represents the load balancer URL. The following example is a conceivable load balancer URL:

```
http://hostname.example.com:8080
```

Load Balancers in Front of Both the Web Agent and Access Manager

This scenario is simply a combination of the scenarios described in the preceding sections. See "Load Balancer in Front of Access Manager" on page 68 and "Load Balancer in Front of Web Agent" on page 68.

Miscellaneous Property Configurations

The following subsections describe miscellaneous configurations you can implement by editing the web agent `AMAgent.properties` configuration file.

Preventing the Query Parameter `sunwMethod` in the Redirect URL

In CDSSO mode, a query parameter `sunwMethod` is used in the redirect URL from Access Manager to the application protected by the agent. This query parameter causes some applications to fail.

If you would like to prevent this query parameter from appearing in the redirect URL, set the value of the following property to `true` as indicated:

```
com.sun.am.remove_sunwmethod = true
```

Sending Composite Advice as a Query Parameter

For Policy Agent 2.2, composite advice is typically sent as POST data. However, if you would like to send composite advice as a query parameter instead, set the following property to true as indicated:

```
com.sun.am.use_redirect_for_advice = true
```

CHAPTER 7

Uninstalling Policy Agent 2.2 for Microsoft IIS 6.0

This chapter first presents you with methods for disabling a web agent. Then the chapter leads you through the uninstallation process, which first requires you to unconfigure the agent from each web site for which it is currently configured. This chapter is organized as follows:

- "Disabling a Web Agent in Policy Agent 2.2" on page 71
- "Agent Unconfiguration for Microsoft IIS 6.0" on page 72
- "Agent Uninstallation for Microsoft IIS 6.0" on page 73

Disabling a Web Agent in Policy Agent 2.2

Note – The task presented in this section does not apply to deployments where Agent for Microsoft IIS 6.0 protects Microsoft Office SharePoint or Outlook Web Access. To disable (or deactivate) the agent in such a scenario, see "Microsoft Office SharePoint: To Deactivate the Access Manager Policy Filter" on page 93.

In certain situations, you might want to disable a web agent temporarily. The task presented in this section requires you to remove the mapping. If you follow the instructions in this task, and you later want to enable the agent, you need to add the mapping again.

▼ To Disable Agent for Microsoft IIS 6.0

1 **From the Microsoft Windows Start menu, choose Programs > Administrative Tools > Internet Information Services Manager.**

2 **Right click the web site protected by the agent.**

3 **Open the Properties tab.**

4 **Click Home Directory.**

5 **Click Configuration.**

6 **Click** *PolicyAgent-base*`\bin\amiis6.dll`

7 **Click Remove.**

8 **Click Yes at the popup "Remove the selected Script Mapping(s)?".**

9 **Click OK.**

10 **Restart the application pool to which the web site belongs.**

11 **Restart the web site.**

Agent Unconfiguration for Microsoft IIS 6.0

If you no longer require Agent for Microsoft IIS 6.0 to protect a particular web site, you can unconfigure the agent from that web site. Furthermore, if you want to uninstall the agent, you must first unconfigure it from all the web sites for which it was configured.

Note – This task does not apply to deployments where Agent for Microsoft IIS 6.0 protects Microsoft Office SharePoint or Outlook Web Access. For such instructions, see "Microsoft Office SharePoint and Outlook Web Access: Unconfiguring and Uninstalling the Agent" on page 97.

Perform the following steps to unconfigure the agent for Microsoft IIS 6.0 from a web site. Make sure that you use the agent configuration file specific to the web site you want to unconfigure. If you need to unconfigure the agent from multiple web sites, you must repeat these steps for each of the web sites.

▼ To Unconfigure Agent for Microsoft IIS 6.0

1 **Stop the web site for which you have configured the agent.**

2 **Stop the application pool to which the web site belongs.**

3 **Change to the directory** *PolicyAgent-base*`\bin`

4 **Run the following VB script to unconfigure the agent (be aware that the command is case sensitive):**

`cscript IIS6admin.vbs -unconfig` *defaultConfig*

`IIS6admin.vbs` A VB script that uses the output of the `IIS6CreateConfig.vbs` script. The output was saved to a configuration file, which for this example labeled *defaultConfig*.

`-unconfig` The option that allows the output to be used to unconfigure the web site.

defaultConfig A place holder for the name of the agent configuration file created previously as described in "To Create Configuration Files: Agent for Microsoft IIS 6.0" on page 35.

The script unconfigures the agent and displays the following message:

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Copyright c 2004 Sun Microsystems, Inc. All rights reserved
Use is subject to license terms

Enter the Agent Resource File Name [IIS6Resource.en] :

Removing the Agent Config Directory
Removing the entries from Windows Product Registry
Unloading the IIS 6.0 Agent
Completed Unconfiguring the IIS 6.0 Agent
```

The unconfiguration does the following:

- Removes the agent configuration directory (specific to a web site)
- Removes the entries from Windows registry.
- Removes the wild card application mappings in Microsoft IIS 6.0.

5 **Accept the default when presented with the following prompt:**

```
Enter the Agent Resource File Name [IIS6Resource.en]:
```

Agent Uninstallation for Microsoft IIS 6.0

Before running the uninstallation program, ensure that you have already unconfigured the agent as follows:

If installed to protect web sites

unconfigure the agent from all the web sites for which it was configured as described in "Agent Unconfiguration for Microsoft IIS 6.0" on page 72.

If installed to protect Microsoft Office SharePoint and Outlook Web Access

unconfigure the agent from Microsoft Office SharePoint and Outlook Web Access as described in "Microsoft Office SharePoint and Outlook Web Access: Unconfiguring and Uninstalling the Agent" on page 97

Uninstallation of Agent for Microsoft IIS 6.0

This task applies to all deployments of Agent for Microsoft IIS 6.0, including deployments where the agent protects Microsoft Office SharePoint or Outlook Web Access.

Perform the following steps to uninstall the agent.

▼ To Uninstall Agent for Microsoft IIS 6.0

1. **Change to the following directory:**

   ```
   C:\Agents\
   ```

2. **Remove the following directory:**

   ```
   iis_v6_x64_WINNT_agent
   ```

3. **Restart the server.**

APPENDIX A

Microsoft Office SharePoint or Outlook Web Access: Deploying Agent for Microsoft IIS 6.0

This appendix provides information that enables you to deploy Agent for Microsoft IIS 6.0 in a manner that provides protection and single sign-on (SSO) to Microsoft Office SharePoint or Outlook Web Access.

Tasks that are specific to Microsoft Office SharePoint or Outlook Web Access are presented in this appendix. The aspects of the installation and configuration of this agent that do not vary when Microsoft Office SharePoint or Outlook Web Access are involved are covered in the appropriate sections, such as Chapter 3, "Installing Policy Agent 2.2 for Microsoft IIS 6.0." Therefore, use this appendix in conjunction with other sections of this guide. Cross references throughout this guide, direct you to and from this appendix as necessary.

This appendix contains the following sections:

Microsoft Office SharePoint and Outlook Web Access: Installing Agent for Microsoft IIS 6.0

You can use Agent for Microsoft IIS 6.0 to provide users with authenticated access to beyond that of web sites. Specifically, you can use this agent to protect Microsoft Office SharePoint or Outlook Web Access. However, to protect these particular resources additional configuration is required. That is to say, you must configure Access Manager as described in the instructions that follow.

Microsoft Office SharePoint and Outlook Web Access: Preparing to Install the Agent

This section focuses on pre-installation steps required for Microsoft Office SharePoint and Outlook Web Access. First, you need to perform the pre-installation steps that apply generally to Agent for Microsoft IIS 6.0, then you need to perform the pre-installation steps specific to Microsoft SharePoint and Outlook Web Access.

To Prepare to Install the Agent

Implement the general pre-installation steps regarding Agent for Microsoft IIS 6.0 as covered in “Preparing To Install Agent for Microsoft IIS 6.0” on page 34 before completing the task that follows.

▼ Microsoft SharePoint and Outlook Web Access: To Prepare for Installation

The steps described in this task are required after you perform the pre-installation steps for the basic installation on Microsoft IIS 6.0 as described in “Preparing To Install Agent for Microsoft IIS 6.0” on page 34.

These additional pre-installation steps are necessary to deploy a post-authentication module on Access Manager. In order to achieve SSO with Microsoft SharePoint or Outlook Web Access using Agent for Microsoft IIS 6.0, a post-authenitcation module is required to be deployed on Access Manager.

Perform the steps in this task on the Access Manager host.

Before You Begin

Caution – When installing Agent for Microsoft IIS 6.0 to protect Outlook Web Access, prior to installing the agent, ensure that the user repositories in Access Manager and Microsoft Exchange Server are synchronized.

For Outlook Web Access 2007, this synchronization can be avoided if the Active Directory instance used by Exchange Server is used as the Access Manager user repository, using the Access Manager LDAP v3 plug-in.

Note – This info serves as a reminder about the compatibility of this agent with versions of Access Manager when the agent is deployed to protect Microsoft SharePoint or Outlook Web Access. The following Access Manager versions are supported:

- Access Manager 7.0 series from Patch 7 forward
- Access Manager 7.1 series from Patch 1 forward

For more information about the compatibility of Agent for Microsoft IIS 6.0 with versions of Access Manager, see "Compatibility of Policy Agent 2.2 With Specific Access Manager Versions" on page 29.

The following information about Access Manager is helpful for this task:

AccessManager-base represents the Access Manager base installation directory. On Solaris systems, the default base installation directory is `/opt/SUNWam`.

The following is the default location of the `AMConfig.properties` file:

```
/etc/opt/SUNWam/config
```

1 **Set the `JAVA_HOME` variable to the location in which JDK binaries are installed.**

2 **Execute `DESgenKey.class` as follows:**

```
# java -classpath am_sdk.jarPath com.sun.identity.common.DESGenKey
```

where *am_sdk.jarPath* is a place holder for the path to the `am_sdk.jar` file.

For example:

```
java -classpath /opt/SUNWam/lib/am_sdk.jar com.sun.identity.common.DESGenKey

Key ==> cIlz47oZBJs=
```

Executing the `DESgenKey.class` returns a string output.

Note – The `am_sdk.jar` file, which is an Access Manager JAR file, is typically found in the `lib` folder of the Access Manager installation, such as `/opt/SUNWam/lib` in a package installation or `sun/webserver7/https-`*hostname*`/web-app/`*hostname*`/amserver/WEB-INF/lib` in single war file installation.

---

3 **Add the string produced in the previous step to a newly created text file as described in the substeps that follow.**

a. **Copy the string produced in the previous step.**

b. **Create a file, which for this example is named `des_key.txt`, in a directory of your choosing.**

The `des_key.txt` name is used in this guide as an example. Name the file differently if you wish.

c. **Save the copied string in the `des_key.txt` file.**

4 **Configure the `com.sun.am.replaypasswd.key` property in the `AMConfig.properties` configuration file as described in the substeps that follow.**

a. **Open the `AMConfig.properties` configuration file.**

b. **Add the following property to the file:**

```
com.sun.am.replaypasswd.key
```

c. **Copy the string from the `des_key.txt` file.**

d. **Add the copied string as the value of the `com.sun.am.replaypasswd.key` property.**

For example, if the string in the `des_key.txt` file is `wuqUJyr=5Gc=`, then the new property would be set as follows:

```
com.sun.am.replaypasswd.key = wuqUJyr=5Gc=
```

5 **Configure a property specific to Microsoft Office SharePoint or Outlook Web Access in the `AMConfig.properties` file as described in the substeps that follow.**

a. **Add the respective property and corresponding value to the file as indicated:**

- **Microsoft Office SharePoint:**

  For SharePoint, an optional property allows you to set an attribute in the Access Manager repository LDAP other than `uid` that allows users to log in to Access Manager to in turn log in to SharePoint:

  `com.sun.am.sharepoint_login_attr_name =` *SharePoint-login-value* where,*SharePoint-login-value* is a placeholder that represents an attribute in the user repository used by SharePoint to authenticate.

For example:

```
com.sun.am.sharepoint_login_attr_name = displayName
```

For example purposes, a user has a `uid` of `ak1234` and a `displayName` of `andy`. In this example, the user logs in to Access Manager using the `uid` (`ak1234`). However, the SharePoint repository has a record for `andy`, not `ak1234`, and the user uses `andy` to log in to the SharePoint application.

Therefore, this property maps `ak1234` to `andy` as the user accesses the SharePoint application after authenticating with Access Manager.

In other words, this property provides a method for mapping any user attribute used by SharePoint to authenticate to the attribute used by Access Manager to authenticate.

- **Outlook Web Access**

  Add the following property and value if you are installing the agent for Outlook Web Access.

  ```
  com.sun.am.iis_owa_enabled = true
  ```

b. **Save and close the `AMConfig.properties` file.**

6 **Restart Access Manager.**

7 **Deploy the post-authentication plug-in, ReplayPasswd, as described in the substeps that follow.**

This step requires the use of Access Manager Console.

a. **Log in to Access Manager as `amadmin`.**

b. **With the Access Control tab selected, click the name of the realm you wish to configure.**

c. **Click the Authentication tab.**

d. **Click Advanced Properties.**

The Advanced Properties button is in the General section.

e. **Scroll down to the Authentication Post Processing Classes field.**

f. **Add the text related to Authentication Post Processing Classes in the manner appropriate for the Access Manager version you are using:**

- **Access Manager 7.0 series from Patch 7 forward**

  For these patches of the Access Manager 7.0 series, execute the following substeps:

  i. **In the Authentication Post Processing Classes field, enter the required text:**

  ```
  com.sun.identity.authentication.spi.ReplayPasswd
  ```

- **Access Manager 7.1 series from Patch 1 forward**

  For these patches of the Access Manager 7.1 series, execute the following substeps:

  i. **In the Authentication Post Processing Classes section, enter the required text:**

  ```
  com.sun.identity.authentication.spi.ReplayPasswd
  ```

  ii. **Click Add.**

g. **Click Save.**

h. **Click Log Out to log out of the Access Manager Console.**

8 **Verify the deployment of the post-authentication plug-in, ReplayPasswd, as described in the substeps that follow.**

a. **Stop Access Manager.**

b. **Access the** `AMConfig.properties` **configuration file.**

c. **Note the value of the following property before changing it to** `message`**, as indicated:**

```
com.iplanet.services.debug.level = message
```

You must change this value back to its original value at the completion of this step.

d. **Save and close the file.**

e. **Start Access Manager.**

f. **Log in to Access Manager Console.**

Again use `amadmin`.

g. **Click Log Out to immediately log out of the Access Manager Console.**

h. **Change directories to the Access Manager debug log files.**

   The default location of the debug log files is `/var/opt/SUNWam/debug`.

i. **Verify the existence of a file named `ReplayPasswd`.**

   The existence of this file indicates the successful deployment of the post-authentication plug-in.

j. **Reset the debug value to its original value.**

k. **Restart Access Manager.**

Microsoft Office SharePoint and Outlook Web Access: Installing the Agent

Once you have completed the preceding pre-installation steps, perform the actual installation as described in “Installing Agent for Microsoft IIS 6.0” on page 34.

Microsoft Office SharePoint and Outlook Web Access: Configuring the Agent

You should come to this section after you have installed the agent as described in “Installing Agent for Microsoft IIS 6.0” on page 34.

Perform the tasks that follow if you are installing Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access.

▼ Microsoft Office SharePoint and Outlook Web Access: To Configure the Agent

1. **Verify that settings are correct in the *defaultConfig* file.**

   If settings are incorrect, edit as required.

   For this task, the *defaultConfig* file is a place holder that you must replace with the name of the agent configuration file created in “To Create Configuration Files: Agent for Microsoft IIS 6.0” on page 35.

2. **Change to the following directory:**

   PolicyAgent-base`\bin`

3 Issue the appropriate command (be aware that the command is case sensitive):

- **Outlook Web Access**

  `cscript OwaAdmin.vbs -config` *defaultConfig*

- **Microsoft Office SharePoint**

  `cscript SPAdmin.vbs -config` *defaultConfig*

  `OwaAdmin.vbs` & `SPAdmin.vbs` — VB scripts that can be used to install the required ISAPI filter. The `OwaAdmin.vbs` script installs the ISAPI filter `amowafilter64.dll` while the `SPAdmin.vbs` script installs the ISAPI filter `amsharepointfilter32.dll`.

  `-config` — The option that allows the output to be used to configure Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access.

4 Accept the default when presented with the following prompt:

```
Enter the Agent Resource File Name [IIS6Resource.en]:
```

The preceding prompt appears in the following context:

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Copyright c 2004 Sun Microsystems, Inc. All rights reserved
Use is subject to license terms

Enter the Agent Resource File Name [IIS6Resource.en]:
```

After you accept the default, a message such as the following should appear:

```
Creating the Agent Config Directory
Creating the AMAgent.properties File
Updating the Windows Product Registry
Completed Configuring the IIS 6.0 Agent
```

Troubleshooting If you experience difficulty after issuing the `OwaAdmin.vbs` script or the `SPAdmin.vbs` script, see the related troubleshooting symptom, “Troubleshooting Symptom 3” on page 103.

▼ Microsoft SharePoint and Outlook Web Access: To Edit the Agent Properties File

This section applies to both Microsoft Office SharePoint and Outlook Web Access. For the task presented in this section, you must edit the web agent `AMAgent.properties` configuration file.

If you are installing this agent to protect Outlook Web Access, more configuration is required, some of which also involves editing the web agent `AMAgent.properties` configuration file. Those instructions are presented in “Outlook Web Access Only: Configuring Agent for Microsoft IIS 6.0” on page 86.

The instructions provided in this section are similar to the instructions for adding the property, `com.sun.am.replaypasswd.key` to the `AMConfig.properties` configuration file as described in “Microsoft SharePoint and Outlook Web Access: To Prepare for Installation” on page 76.

The same property and respective value added to the `AMConfig.properties` configuration file must now be added to the web agent `AMAgent.properties` configuration file.

For information about the location of the web agent `AMAgent.properties` configuration file, see “Locating the Web Agent `AMAgent.properties` Configuration File” on page 54.

1. **Open the web agent `AMAgent.properties` configuration file.**

2. **Add the following property to the file:**

   ```
   com.sun.am.replaypasswd.key
   ```

3. **Copy the string from the `des_key.txt` file.**

   For more information on the `des_key.txt` file, see “Microsoft SharePoint and Outlook Web Access: To Prepare for Installation” on page 76.

4. **Add the copied string as the value of the `com.sun.am.replaypasswd.key` property.**

   For example, if the string in the `des_key.txt` file is `wuqUJyr=5Gc=`, then the new property would be set as follows:

   ```
   com.sun.am.replaypasswd.key = wuqUJyr=5Gc=
   ```

5. **(Conditional) If you are configuring the agent for Microsoft SharePoint, save and close the web agent `AMAgent.properties` configuration file.**

Next Steps At this point, the next task to be implemented varies depending on if you are deploying this agent to protect Microsoft Office SharePoint or Outlook Web Access.

If you are installing this agent to protect Microsoft Office SharePoint, continue to the next section, "Microsoft Office SharePoint Only: Configuring Agent for Microsoft IIS 6.0" on page 84, to complete an additional configuration task specific to Microsoft Office SharePoint.

If you are installing this agent to protect Outlook Web Access, skip to "Outlook Web Access Only: Configuring Agent for Microsoft IIS 6.0" on page 86, to complete additional configuration tasks specific to Outlook Web Access.

▼ Microsoft Office SharePoint and Outlook Web Access: To Add a New System Environment Variable for NSPR Threads

This is a required task. The system environment variable added in this task is used internally by the NSPR libraries.

1. **Add a new System environment variable for NSPR threads as follows:**

   ```
   NSPR_NATIVE_THREADS_ONLY
   ```

   The value is as follows:

   ```
   1
   ```

2. **Restart the Microsoft IIS 6.0 server.**

Microsoft Office SharePoint Only: Configuring Agent for Microsoft IIS 6.0

If you are installing Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint, tasks specific to Microsoft Office SharePoint are required. This section provides those configuration instructions in a series of tasks.

▼ Microsoft Office SharePoint: To Enable the Authentication Method to Basic

To protect Microsoft Office SharePoint with this agent you must ensure that the authentication method for the Microsoft IIS 6.0 Server is set to Basic authentication as described in this task.

1. **As an administrator, log in to Windows 2003 Server where Microsoft Office SharePoint is running.**

2 **In the Microsoft Windows Start menu, choose run.**

3 **Type the following: inetmgr**

4 **Click OK.**

5 **Expand the local computer.**

6 **Expand the Web Sites folder.**

7 **Right click the SharePoint site that you are protecting with the agent.**

The agent-protected SharePoint site is typically the site using port 80 (SharePoint — 80).

8 **In the options list, click Properties.**

The Default Web Site Properties dialog box appears.

9 **Select the Directory Security tab.**

10 **Click Edit in the Authentication and access control section.**

11 **Select Basic authentication in the Authenticated access section.**

Ensure that no other authentication option is checked.

12 **Click OK.**

13 **Click OK again to close the Web site properties.**

▼ Microsoft Office SharePoint: To Modify the `signout.aspx` File to Properly Handle the Logout Process

1 **Back up the `signout.aspx` file.**

This file is typically available in the following directory:

```
C:\Program Files\Common Files\Microsoft Shared\web server extensions\
12\TEMPLATE\LAYOUTS
```

2 **Open the `signout.aspx` file.**

3 **Replace the lines of code indicated within this step.**

Original Code Snippet (replace this code snippet):

```
function _spBodyOnLoad()
[
   try
   [
      document.execCommand("ClearAuthenticationCache");
   ]
   catch (e) []
   window.close();
]
```

Replacement Code Snippet (Use this code snippet to replace the original code snippet):

```
function _spBodyOnLoad()
[
   window.location="https://amHost:amPort/amserver/UI/Logout";
]
```

Where *amHost* and *amPort* are place holders that you must replace with the fully qualified domain name (FQDN) host name and port number, respectively, of the Access Manager server.

4 **Save and close the `signout.aspx` file.**

5 **Restart the Microsoft IIS 6.0 server using the `iisreset` command.**

Next Steps Now you can verify the installation of the agent as described in "Microsoft Office SharePoint and Outlook Web Access: Verifying a Successful Agent Installation" on page 91.

Outlook Web Access Only: Configuring Agent for Microsoft IIS 6.0

If you are installing Agent for Microsoft IIS 6.0 to provide SSO to Outlook Web Access, tasks specific to Outlook Web Access are required. This section provides those configuration instructions in a series of tasks.

▼ Outlook Web Access: To Edit the Agent Properties File

Before You Begin Open the web agent `AMAgent.properties` configuration file if it is not already open.

1 **In the web agent `AMAgent.properties` configuration file, set the following property to false as shown:**

```
com.sun.am.notification.enable = false
```

2 **Add the following property with its value set to `true` as indicated:**

```
com.sun.am.policy.agents.config.iis.owa_enabled = true
```

3 **Add the property illustrated in this step with its value set to the URL of a local idle session timeout page.**

The value for the property in the example that follows represents the location of a local idle session timeout page (`timeout.aspx`). However, the instructions for creating the local idle session timeout page are presented in the task that follows: "Outlook Web Access: To Create a Local Idle Session Timeout Page" on page 87. You can either complete that task first or set this property now by choosing a name at this time for the local idle session timeout page and its full path.

Example Property Setting:

```
com.sun.am.policy.agents.config.iis.owa_enabled_session_timeout_url =
https://agentHost.domain-name:444/timeout.aspx
```

The `timeout.aspx` page is an example timeout page name, which is used in this guide in reference to the timeout page used with Agent for Microsoft IIS 6.0 when protecting Outlook Web Access. However, `timeout.aspx` is only an example. You might chose to use a different page name.

4 **Save and close the web agent `AMAgent.properties` configuration file.**

▼ Outlook Web Access: To Create a Local Idle Session Timeout Page

This task consists of steps that vary in specificity. These steps are to be performed on the Microsoft IIS 6.0 Server. The purpose of this task is to create a local web site to redirect timeout requests to the Access Manager timeout page.

1 **Create a new virtual server (a different web site) in the Microsoft IIS 6.0 Server administration console.**

2 **For the new virtual server, create a corresponding application pool with a new document folder.**

An example name for this folder is `C:\Inetpub\test`.

While the preceding example folder name is used throughout this task, it is only an example. You might chose to use a different name.

3 **Install SSL on the newly created web site.**

Tip –

- Ensure that this web site is accessible from a browser.
- Configure the port number.

  An example port number for this port is 444. However, 444 is only an example. You might chose to use a different port number.
- Ensure that the Outlook Web Access server runs on a different port (therefore, for the example used in this task, *not* port 444).

4. **Ensure that the web site is enabled to run scripts and executable files as described in the substeps that follow:**

   a. **As an administrator, log in to Windows 2003 Server where Outlook Web Access Server is running.**

   b. **In the Microsoft Windows Start menu, choose run.**

   c. **Type the following: inetmgr.**

   d. **Click OK.**

   e. **Expand the local computer.**

   f. **Expand the Web Sites folder.**

   g. **Right-click Default Web Site.**

      An options list appears.

   h. **In the options list, click Properties.**

      The Default Web Site Properties dialog box appears.

   i. **Select the Home Directory tab.**

   j. **Under the Application settings section, in the Execute permissions drop down list, select Scripts and Executables.**

5. **Create a** `.aspx` **page, such as** `timeout.aspx`**, in the folder** `C:\Inetpub\test`**.**

   As explained previously, `timeout.aspx` is only an example. However, ensure that you use the same name for this page as used in "Outlook Web Access: To Edit the Agent Properties File" on page 86.

6 Add the markup information provided in this step to the `timeout.aspx` file, editing the place holders as appropriate.

In the markup information provided in this step, the following place holders apply:

amHost A place holder that you must replace with the name of the host machine on which Access Manager is running.

amHost.domain-name The fully qualified domain name of the Access Manager host machine.

agentHost A place holder that you must replace with the name of the host machine (or the alias name, if an alias name is used instead of the actual host name) on which the agent is running.

agentHost.domain-name A place holder that you must replace with the fully qualified domain name of the agent host machine.

```
 <%@ Page language="c#" AutoEventWireup="false"  %>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<%
string cookieValue="";
if (Request.Cookies["UserContext"] != null)
{
   cookieValue=Request.Cookies["UserContext"].Value;
   HttpCookie myCookie = new HttpCookie("UserContext",cookieValue);
   myCookie.Expires = DateTime.Now.AddDays(-1d);
   myCookie.Path = "/";
   Response.Cookies.Add(myCookie);
}

%>

<script language="javascript">
   function RefreshParent()
   {
      gotoUrl="https://amHost.domain-name:443/amserver/UI/Logout?goto=
https://agentHost.domain-name:443/owa";
                window.location.href = gotoUrl;
                window.parent.location.href = gotoUrl;
                window.parent.parent.location.href = gotoUrl;
           window.opener.parent.location.href = gotoUrl;
   }
   function CallRefresh()
   {
      RefreshParent();
      if(!window.close())
      {
         window.close();
      }
   }
</script>


</head>
<body onload="javascript:CallRefresh()">
</body>
</html>
```

7 **Save and close the `timeout.aspx` file.**

▼ Outlook Web Access: To Modify the `logoff.aspx` File to Properly Handle the Logout Process

1 **Back up the file `C:\Program Files\Microsoft\Exchange Server\ClientAccess\Owa\auth\logoff.aspx`.**

2 **Retrieve the cookie domain name as described in the substeps that follow.**

The cookie domain name you are retrieving in this step is required in the next step for the `logoff.aspx` file.

a. **Log in to Access Manager as `amadmin`.**

b. **Select the Configuration tab.**

c. **Scroll as necessary to click Platform under the System Properties section.**

d. **In the Current Values list, take note of name of the appropriate cookie domain.**

The Current Values list is in the Cookie Domains section. The domain name you need to record for later use is the domain where Microsoft IIS 6.0 Server is installed and running.

3 **Replace the contents of the `logoff.aspx` file with the markup information provided in this step.**

In the markup information that follows, *amHost.domain-name* and *agentHost.domain-name* are place holders described in the task "Outlook Web Access: To Create a Local Idle Session

Timeout Page" on page 87. In this case, *cookie-domain* is a place holder that you must replace with the cookie domain name retrieved in the previous step.

```
<%@ Page language="c#" AutoEventWireup="false"  %>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<%
string str="owa";
if (Request.Cookies["owaAuthCookie"] != null)
{
    HttpCookie myCookie = new HttpCookie("owaAuthCookie","amOwaValue");
    myCookie.Expires = DateTime.Now.AddDays(-1d);
    myCookie.Domain = ".cookie-domain";
    myCookie.Path = "/";
    Response.Cookies.Add(myCookie);
}
%>
<meta http-equiv="Refresh" content="0;url=
https://amHost.domain-name:443/amserver/UI/Logout?goto=
https%3A%2F%2FagentHost.domain-name%3A443%2F<%=str%>%2F">
</head>
</html>
```

4 **Save and close the** `logoff.aspx` **file.**

Next Steps Now you can verify the installation of the agent as described in "Microsoft Office SharePoint and Outlook Web Access: Verifying a Successful Agent Installation" on page 91.

Microsoft Office SharePoint and Outlook Web Access: Verifying a Successful Agent Installation

This section describes the methods for verifying the installation of Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint and Outlook Web Access. Refer to the section that applies to your deployment.

Microsoft Office SharePoint If the agent is installed correctly, an attempt to access a protected resource results in the presentation of the Access Manager login page. Entering proper credentials at this point, successfully authenticates users, and if they have the appropriate SharePoint access rights to the resource, they are granted access. Then when users attempt to access any other application secured by the same Access Manager server, they are not prompted for authentication. They are granted or denied access to the resource depending on defined policies.

Outlook Web Access

If the agent is installed correctly, an attempt to access the Outlook Web Access URL, which is `https://`*agentHost.domain-name*`/owa`, results in the presentation of the Access Manager login page. Entering proper credentials at this point, successfully authenticates users and provides access to the Outlook Web Access inbox. Then when users attempt to access any other application secured by the same Access Manager server, they are not prompted for authentication. They are granted or denied access to the resource depending on defined policies.

Microsoft Office SharePoint and Outlook Web Access: Deactivating and Reactivating the Access Manager Policy Filter

If you decide to temporarily stop using Access Manager for SSO from Microsoft Office SharePoint or Outlook Web Access to other applications, you can accomplish this by deactivating the policy filter. Therefore, uninstalling the agent would not be necessary. If you are interested in uninstalling the agent instead, see Chapter 7, “Uninstalling Policy Agent 2.2 for Microsoft IIS 6.0.” You can deactivate the policy filter from the Microsoft IIS 6.0 Server. The assumption is that you would reactivate the filter later.

In this guide, the tasks for deactivating and reactivating the policy filter are covered for Microsoft Office SharePoint and Outlook Web Access separately as follows:

- “Microsoft Office SharePoint: Deactivating and Reactivating the Access Manager Policy Filter” on page 92
- “Outlook Web Access: Deactivating and Reactivating the Access Manager Policy Filter” on page 94

Be certain to use the appropriate instructions depending on your specific deployment.

Microsoft Office SharePoint: Deactivating and Reactivating the Access Manager Policy Filter

Two tasks follow: one for deactivating the policy filter and one for reactivating it.

▼ Microsoft Office SharePoint: To Deactivate the Access Manager Policy Filter

1. **As an administrator, log in to Windows 2003 Server where Microsoft Office SharePoint is running.**
2. **In the Microsoft Windows Start menu, choose run.**
3. **Type the following: inetmgr.**
4. **Click OK.**
5. **Expand the local computer.**
6. **Expand the Web Sites folder.**
7. **Right-click Default Web Site.**

   An options list appears.
8. **In the options list, click Properties.**

   The Default Web Site Properties dialog box appears.
9. **Click the ISAPI Filters tab.**
10. **Click the amspfilter filter.**
11. **Click Remove.**

Caution – After removing the filter manually, as described in this task, if you want to reactivate it or if you want to unconfigure the agent by issuing the `SPAdmin.vbs -unconfig` command, you must first add the filter back manually (see "Microsoft Office SharePoint: To Reactivate the Access Manager Policy Filter" on page 94) with the same filter name, for example amspfilter. Otherwise, an error will be issued.

12. **Click OK.**
13. **Restart the Microsoft IIS 6.0 Server.**

    A method for restarting this server is to enter `iisreset` in a command window.

▼ Microsoft Office SharePoint: To Reactivate the Access Manager Policy Filter

1. **As an administrator, log in to Windows 2003 Server where Microsoft Office SharePoint is running.**
2. **In the Microsoft Windows Start menu, choose run.**
3. **Type the following: inetmgr.**
4. **Click OK.**
5. **Expand the local computer.**
6. **Expand the Web Sites folder.**
7. **Right click Default Web Site.**

   An options list appears.
8. **In the options list, Click Properties.**

   The Default Web Site Properties dialog box appears.
9. **Click the ISAPI Filters tab.**
10. **Click Add.**
11. **In the Filter Name field, enter the following: amspfilter**
12. **In the Executable field, enter** *PolicyAgent-base*`\bin\amsharepointfilter32.dll`.
13. **Click OK.**

Outlook Web Access: Deactivating and Reactivating the Access Manager Policy Filter

Two tasks follow: one for deactivating the policy filter and one for reactivating it.

▼ Outlook Web Access: To Deactivate the Access Manager Policy Filter

1. **As an administrator, log in to Windows 2003 Server where Outlook Web Access Server is running.**

2. **In the Microsoft Windows Start menu, choose run.**

3. **Type the following: inetmgr.**

4. **Click OK.**

5. **Expand the local computer**

6. **Expand the Web Sites folder.**

7. **Right click Default Web Site.**

   An options list appears.

8. **In the options list, click Properties.**

   The Default Web Site Properties dialog box appears.

9. **Click the ISAPI Filters tab.**

10. **Click the amowafilter filter.**

11. **Click Remove.**

Caution – After removing the filter manually, as described in this task, if you want to reactivate it or if you want to unconfigure the agent by issuing the `OwaAdmin -unconfig` command, you must first add the filter back manually (see "Outlook Web Access: To Reactivate the Access Manager Policy Filter" on page 95) with the same filter name, for example amowafilter. Otherwise, an error will be issued.

---

12. **Click OK.**

13. **Restart the Microsoft IIS 6.0 Server.**

▼ Outlook Web Access: To Reactivate the Access Manager Policy Filter

1. **As an administrator, log in to Windows 2003 Server where Outlook Web Access Server is running.**

2. **In the Microsoft Windows Start menu, choose run.**
3. **Type the following: inetmgr.**
4. **Click OK.**
5. **Expand the local computer.**
6. **Expand the Web Sites folder.**
7. **Right click Default Web Site.**

   An options list appears.
8. **In the options list, click Properties.**

   The Default Web Site Properties dialog box appears.
9. **Click the ISAPI Filters tab.**
10. **Click Add.**
11. **Enter `amowafilter` in the Filter Name field**

    ⚠ **Caution –** Putting a different name here will cause `OwaAdmin -unconfig` to fail during uninstallation.
12. **Enter** *PolicyAgent-base*`\bin\amowafilter64.dll` **in the Executable field.**
13. **Click OK.**
14. **Click Apply.**
15. **Click OK again**
16. **Restart Microsoft IIS 6.0 Server using the `iisreset` command.**

Microsoft Office SharePoint and Outlook Web Access: Unconfiguring and Uninstalling the Agent

If you no longer require Agent for Microsoft IIS 6.0 to protect Microsoft Office SharePoint or Outlook Web Access, you can unconfigure the agent. Be aware that to uninstall the agent, you must first unconfigure it.

The task that follows in this section is similar to the task in “Agent Unconfiguration for Microsoft IIS 6.0” on page 72. However, this task is specific to deployments where Agent for Microsoft IIS 6.0 protects Microsoft Office SharePoint or Outlook Web Access. Though the unconfiguration task varies, the uninstallation task does not. The uninstallation task that applies to all deployments of this agent is as follows “Agent Uninstallation for Microsoft IIS 6.0” on page 73.

Microsoft Office SharePoint and Outlook Web Access: To Unconfigure Agent for Microsoft IIS 6.0

1. **Change to the directory** *PolicyAgent-base*`\bin`

2. **Run the appropriate VB script to unconfigure the agent (be aware that the command is case sensitive):**

   - **Outlook Web Access**

     `cscript OwaAdmin.vbs -unconfig` *defaultConfig*

   - **Microsoft Office SharePoint**

     `cscript SPAdmin.vbs -unconfig` *defaultConfig*

   `OwaAdmin.vbs` & `SPAdmin.vbs`
   VB scripts that can be used to uninstall the ISAPI filters. The `OwaAdmin.vbs` script uninstalls `amowafilter64.dll` while the`SPAdmin.vbs` script uninstalls `amsharepointfilter32.dll`.

   `-unconfig`
   The option that allows the output to be used to unconfigure the agent from protecting Microsoft Office SharePoint or Outlook Web Access.

   defaultConfig
   A place holder for the name of the agent configuration file created previously as described in “To Create Configuration Files: Agent for Microsoft IIS 6.0” on page 35.

Each of the preceding scripts unconfigures the agent and displays the following message:

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Copyright c 2004 Sun Microsystems, Inc. All rights reserved
Use is subject to license terms

Enter the Agent Resource File Name [IIS6Resource.en] :

Removing the Agent Config Directory
Removing the entries from Windows Product Registry
Unloading the IIS 6.0 Agent
Completed Unconfiguring the IIS 6.0 Agent
```

The unconfiguration does the following:

- Removes the agent configuration directory
- Removes the entries from Windows registry.
- Removes the wild card application mappings in Microsoft IIS 6.0.

3. **Accept the default when presented with the following prompt:**

   ```
   Enter the Agent Resource File Name [IIS6Resource.en]:
   ```

4. **Change to the following directory:** `C:\Agents\`**.**

5. **Remove the appropriate directory depending on the version of this agent.**

   - **The 64–bit Agent for IIS 6.0 version & the OWA 2007 version**

     For these two versions of the agent, remove the following directory:

     `iis_v6_x64_WINNT_agent`

   - **The SharePoint 2007 version**

     For this version of the agent, remove the following directory:

     `iis_v6_WINNT_agent`

6. **Restart the Microsoft IIS 6.0 Server.**

Next Steps Once you have completed the unconfiguration process, see "Agent Uninstallation for Microsoft IIS 6.0" on page 73 to uninstall the agent.

Microsoft Office SharePoint and Outlook Web Access: Tasks Not Specified

Refer to the respective sections of this guide to perform tasks that are not specifically described or referenced in this appendix since such sections apply to all deployments of Agent for Microsoft IIS 6.0, including Microsoft Office SharePoint and Outlook Web Access deployments. For example, see "Uninstallation of Agent for Microsoft IIS 6.0" on page 74 to uninstall Agent for Microsoft IIS 6.0 whether Microsoft Office SharePoint or Outlook Web Access are involved in the deployment or not.

APPENDIX B

Troubleshooting a Web Agent Deployment

This appendix applies to Agent for Microsoft IIS 6.0. If a problem is discussed in this appendix, it either applies only to this agent or it applies to two or more agents with one of them being this agent. This appendix explains how you can resolve problems that you might encounter while deploying or using this web agent. Be sure to also check the *Sun Java System Access Manager Policy Agent 2.2 Release Notes*, to see if the problem that you encounter is a known limitation of this web agent. If workarounds are available for such problems, they will be provided in the release notes.

Troubleshooting Symptoms in Agent for Microsoft IIS 6.0

This section includes problems you might encounter. The explanation of the symptom is followed by possible causes and solutions.

Troubleshooting Symptom 1

Symptom: Cannot install the web agent after a previous installation has been removed.

Possible Causes:

- You might have an existing installation of the web agent.
- You might have a previously-installed web agent and did not use the web agent's uninstallation program to uninstall the agent.
- The installation program's `productregistry` file might be corrupted.

Possible Solution: To resolve the issue, manually remove the web agent as explained in the following task description.

▼ To Manually Remove Agent for Microsoft IIS 6.0

1. **Stop all of the web sites.**

2. **Stop all of the application pools.**

3. **Remove entries from product registry**

   a. **Issue the following command from the command line:**

   ```
   regedit
   ```

   b. **Traverse to the following:**

   ```
   HKEY_LOCAL_MACHINE
   ```

   c. **Click Software**

   d. **Click Sun Microsystems**

   e. **Remove the following entry:**

   Access Manager IIS6 Agent

4. **Remove the** *PolicyAgent-base* **directory from the server.**

   where *PolicyAgent-base* represents the directory in which the web agent was originally installed.

5. **Remove the appropriate directory depending on the version of this agent.**

   - **The 64–bit Agent for IIS 6.0 version & the OWA 2007 version**

     For these two versions of the agent, remove the following directory:

     ```
     iis_v6_x64_WINNT_agent
     ```

   - **The SharePoint 2007 version**

     For this version of the agent, remove the following directory:

     ```
     iis_v6_WINNT_agent
     ```

6. **Restart the server.**

Troubleshooting Symptom 2

Symptom: When a user attempts to access a resource using Internet Explorer as the browser, access is denied.

Possible Cause: Internet Explorer overrides the port number of the web agent with the Access Manager port number. In such cases, the agent log file lists the URL that is being evaluated. The port number for that URL is incorrect.

Possible Solution: You can ensure this problem does not occur by setting the following property in the web agent `AMAgent.properties` configuration file to `true` as shown:

```
com.sun.am.policy.agents.config.override_port = true
```

Troubleshooting Symptom 3

Symptom: This troubleshooting symptom applies when Agent for Microsoft IIS 6.0 protects Microsoft Office SharePoint or Outlook Web Access. After issuing the appropriate configuration command (`OwaAdmin.vbs -config` *defaultConfig* for the Outlook Web Access 2007 version of this agent or `SPAdmin.vbs -config` *defaultConfig* for the Microsoft Office SharePoint 2007 version of this agent), one of the following messages appear, depending on the version of this agent:

For Outlook Web Access 2007:

```
"C:\Agents\iis_v6_WINNT_agent\web_agents\iis6_agent\bin\OwaAdmin.vbs
(null):
Cannot create a file when that file already exists. "
```

For Microsoft Office SharePoint 2007”

```
"C:\Agents\iis_v6_WINNT_agent\web_agents\iis6_agent\bin\SPAdmin.vbs(307, 4)
(null):
Cannot create a file when that file already exists."
```

Possible Cause: As explained in the preceding messages, a `.dll` file was manually removed and not reloaded with the same name.

For Outlook Web Access 2007: The `amowafilter64.dll` file was manually removed and not loaded back with the same name "amowafilter."

For Microsoft Office SharePoint 2007: The `amsharepointfilter32.dll` was manually removed and not loaded back with the same name "amspfilter".

Possible Solution: As explained in the preceding messages, issue the appropriate scripts to clear the first issuance and to then reissue the original command.

For Outlook Web Access 2007: Run the command `cscript OwaAdmin.vbs -unconfig c:\amconfig.txt` followed by `cscript OwaAdmin.vbs -config c:\amconfig.txt`.

For Microsoft Office SharePoint 2007: Run the command `cscript SPAdmin.vbs -unconfig c:\amconfig.txt` followed by`cscript SPAdmin.vbs -config c:\amconfig.txt`.

APPENDIX C

Web Agent AMAgent.properties Configuration File

The web agent AMAgent.properties configuration file contains the necessary configuration properties needed for the web agent to function properly. It also contains the necessary information needed for the Sun Java System Access Manager SDK to function properly in a client installation mode as used by the web agent.

Properties in the Web Agent AMAgent.properties Configuration File

The web agent AMAgent.properties configuration file is located as described in Table 6–1. For a more detailed discussion of the key tasks you can perform using this configuration file, see "Key Features and Tasks Performed with the Web Agent AMAgent.properties Configuration File" on page 53.

For detailed information about every property, see the actual web agent AMAgent.properties configuration file in the product itself for a description of each property.

Most property names in the web agent AMAgent.properties configuration file have changed for Policy Agent 2.2. The following list highlights the change in property names by presenting the current property name in the release paired with the former property name from the 2.1 release. You can use this information to map the former property name to the current property name. Most properties apply to all web agents in the 2.2 release. A few properties are specific to one or a few web agents.

TABLE C–1 Changes in the Web Agent AMAgent.properties Configuration File for Policy Agent 2.2

2.2 Property Name	Former Property Name: 2.1 and Prior
com.sun.am.cookie.name	com.sun.am.cookieName
com.sun.am.cookie.encode	com.sun.am.cookieEncoded

TABLE C–1 Changes in the Web Agent AMAgent.properties Configuration File for Policy Agent 2.2 *(Continued)*

2.2 Property Name	Former Property Name: 2.1 and Prior
com.sun.am.log.level	com.sun.am.logLevels
com.sun.am.naming.url	com.sun.am.namingURL
com.sun.am.sslcert.dir	com.sun.am.sslCertDir
com.sun.am.certdb.prefix	com.sun.am.certDbPrefix
com.sun.am.trust_server_certs	com.sun.am.trustServerCerts
com.sun.am.notification.enable	com.sun.am.notificationEnabled
com.sun.am.notification.url	com.sun.am.notificationURL
com.sun.am.load_balancer.enable	com.sun.am.loadBalancer_enable
com.sun.am.policy.am.login.url	com.sun.am.policy.am.loginURL
com.sun.am.policy.am.username (unchanged)	com.sun.am.policy.am.username
com.sun.am.policy.am.password (unchanged)	com.sun.am.policy.am.password
com.sun.am.policy.am.url_comparison.case_ignore	com.sun.am.policy.am.urlComparison.caseIgnore
com.sun.am.policy.am.polling.interval	com.sun.am.policy.am.cacheEntryLifeTime
com.sun.am.policy.am.userid.param	com.sun.am.policy.am.userIdParam
com.sun.am.policy.am.lb.cookie.name	com.sun.am.policy.am.ias_SLB_cookie_name
com.sun.am.policy.am.fetch_from_root_resource	com.sun.am.policy.am.fetchFromRootResource
com.sun.am.policy.agents.config.local.log.file	com.sun.am.logFile
com.sun.am.policy.agents.config.local.log.rotate	NEW PROPERTY
com.sun.am.policy.agents.config.local.log.size	NEW PROPERTY
com.sun.am.policy.agents.config.remote.log	com.sun.am.serverLogFile
com.sun.am.policy.agents.config.profile.attribute.fetch.mode	com.sun.am.policy.am.ldapattribute.mode
com.sun.am.policy.agents.config.profile.attribute.map	com.sun.am.policy.am.headerAttributes

TABLE C–1 Changes in the Web Agent AMAgent.properties Configuration File for Policy Agent 2.2 *(Continued)*

2.2 Property Name	Former Property Name: 2.1 and Prior
`com.sun.am.policy.agents.config.profile.attribute.cookie.prefix`	`com.sun.am.policy.am.ldapattribute.cookiePrefix`
`com.sun.am.policy.agents.config.profile.attribute.cookie.maxage`	`com.sun.am.policy.am.ldapattribute.cookieMaxAge`
`com.sun.am.policy.agents.config.session.attribute.fetch.mode`	NEW PROPERTY
`com.sun.am.policy.agents.config.session.attribute.map`	NEW PROPERTY
`com.sun.am.policy.agents.config.response.attribute.fetch.mode`	NEW PROPERTY
`com.sun.am.policy.agents.config.add_response_attrs`	NEW PROPERTY
`com.sun.am.policy.agents.config.version`	`com.sun.am.policy.agents.version`
`com.sun.am.policy.agents.config.audit.accesstype`	`com.sun.am.policy.agents.logAccessType`
`com.sun.am.policy.agents.config.agenturi.prefix`	`com.sun.am.policy.agents.agenturiprefix`
`com.sun.am.policy.agents.config.locale`	`com.sun.am.policy.agents.locale`
`com.sun.am.policy.agents.config.instance.name`	`com.sun.am.policy.agents.instanceName`
`com.sun.am.policy.agents.config.do_sso_only`	`com.sun.am.policy.agents.do_sso_only`
`com.sun.am.policy.agents.config.accessdenied.url`	`com.sun.am.policy.agents.accessDeniedURL`
`com.sun.am.policy.agents.config.url.redirect.param`	`com.sun.am.policy.agents.urlRedirectParam`
`com.sun.am.policy.agents.config.fqdn.default`	`com.sun.am.policy.agents.fqdnDefault`
`com.sun.am.policy.agents.config.fqdn.map`	`com.sun.am.policy.agents.fqdnMap`
`com.sun.am.policy.agents.config.cookie.reset.enable`	`com.sun.am.policy.agents.cookie_reset_enabled`
`com.sun.am.policy.agents.config.cookie.reset.list`	`com.sun.am.policy.agents.cookie_reset_list`

TABLE C–1 Changes in the Web Agent AMAgent.properties Configuration File for Policy Agent 2.2 *(Continued)*

2.2 Property Name	Former Property Name: 2.1 and Prior
com.sun.am.policy.agents.config.cookie.domain.list	com.sun.am.policy.agents.cookieDomainList
com.sun.am.policy.agents.config.anonymous_user	com.sun.am.policy.agents.unauthenticatedUser
com.sun.am.policy.agents.config.anonymous_user.enable	com.sun.am.policy.agents.anonRemoteUserEnabled
com.sun.am.policy.agents.config.notenforced_list	com.sun.am.policy.agents.notenforcedList
com.sun.am.policy.agents.config.notenforced_list.invert	com.sun.am.policy.agents.reverse_the_meaning_of_notenforcedList
com.sun.am.policy.agents.config.notenforced_client_ip_list	com.sun.am.policy.agents.notenforced_client_IP_address_list
com.sun.am.policy.agents.config.postdata.preserve.enable	com.sun.am.policy.agents.is_postdatapreserve_enabled
com.sun.am.policy.agents.config.postcache.entry.lifetime	com.sun.am.policy.agents.postcacheentrylifetime
com.sun.am.policy.agents.config.cdsso.enable	com.sun.am.policy.agents.cdsso-enabled
com.sun.am.policy.agents.config.cdcservlet.url	com.sun.am.policy.agents.cdcservletURL
com.sun.am.policy.agents.config.client_ip_validation.enable	com.sun.am.policy.agents.client_ip_validation_enable
com.sun.am.policy.agents.config.logout.url	com.sun.am.policy.agents.logout.url
com.sun.am.policy.agents.config.logout.cookie.reset.list	com.sun.am.policy.agents.logout.cookie_reset_list
com.sun.am.policy.agents.config.get_client_host_name	com.sun.am.policy.agents.getClientHostname
com.sun.am.policy.agents.config.convert_mbyte.enable	com.sun.am.policy.agents.convertMbyteEnabled
com.sun.am.policy.agents.config.ignore_path_info	com.sun.am.ignore_path_info
com.sun.am.policy.agents.config.override_protocol	com.sun.am.policy.agents.overrideProtocol

TABLE C–1 Changes in the Web Agent AMAgent.properties Configuration File for Policy Agent 2.2 *(Continued)*

2.2 Property Name	Former Property Name: 2.1 and Prior
`com.sun.am.policy.agents.config.override_host`	`com.sun.am.policy.agents.overrideHost`
`com.sun.am.policy.agents.config.override_port`	`com.sun.am.policy.agents.overridePort`
`com.sun.am.policy.agents.config.override_notification.url`	`com.sun.policy.agents.overrideNotificationUrl`
`com.sun.am.policy.agents.config.connection_timeout`	NEW PROPERTY
`com.sun.am.remove_sunwmethod`	NEW PROPERTY
`com.sun.am.use_redirect_for_advice`	NEW PROPERTY

APPENDIX D

Error Codes

This appendix lists the error codes you might encounter while installing and configuring a web agent. It also provides explanations for the each code item.

Error Code List

This list of error codes includes locations that are reserved for error codes that do not currently exist.

0. AM_SUCCESS	The operation completed successfully.
1. AM_FAILURE	The operation did not complete successfully. Please refer to the log file for more details.
2. AM_INIT_FAILURE	The C SDK initialization routine did not complete successfully. All the other APIs may be used only if the initialization went through successfully.
3. AM_AUTH_FAILURE	The authentication did not go through successfully. This error is returned either by the Authentication API or the Policy Initialization API, which tries to authenticate itself as a client to Access Manager.
4. AM_NAMING_FAILURE	The naming query failed. Please look at the log file for further information.
5. AM_SESSION_FAILURE	The session operation did not succeed. The operation may be any of the operations provided by the session API.
6. AM_POLICY_FAILURE	The policy operation failed. Details of policy failure may be found in the log file.

7. This is a reserved error code.	Currently, no error code exists at this location.
8. AM_INVALID_ARGUMENT	The API was invoked with one or more invalid parameters. Check the input provided to the function.
9. This is a reserved error code.	Currently, no error code exists at this location.
10. This is a reserved error code.	Currently, no error code exists at this location.
11. AM_NO_MEMORY	The operation failed because of a memory allocation problem.
12. AM_NSPR_ERROR	The underlying NSPR layer failed. Please check log for further details.
13. This is a reserved error code.	Currently, no error code exists at this location.
14. AM_BUFFER_TOO_SMALL	The web agent does not have memory allocated to receive data from Access Manager.
15. AM_NO_SUCH_SERVICE_TYPE	The service type input by the user does not exist. This is a more specific version of `AM_INVALID_ARGUMENT` error code. The error can occur in any of the API that take `am_policy_t` as a parameter.
16. AM_SERVICE_NOT_AVAILABLE	Currently, no error code exists at this location.
17. AM_ERROR_PARSING_XML	During communication with Access Manager, there was an error while parsing the incoming XML data.
18. AM_INVALID_SESSION	The session token provided to the API was invalid. The session may have timed out or the token is corrupted.
19. AM_INVALID_ACTION_TYPE	This exception occurs during policy evaluation, if such an action type does not exist for a given policy decision appropriately found for the resource.
20. AM_ACCESS_DENIED	The user is denied access to the resource for the kind of action requested.
21. AM_HTTP_ERROR	There was an HTTP protocol error while contacting Access Manager.
22. AM_INVALID_FQDN_ACCESS	The resource provided by the user is not a fully qualified domain name. This is a web container

specific error and may be returned by the `am_web_is_access_allowed` function only.

23. AM_FEATURE_UNSUPPORTED — The feature being invoked is not implemented as of now. Only the interfaces have been defined.

24. AM_AUTH_CTX_INIT_FAILURE — The Auth context creation failed. This error is thrown by `am_auth_create_auth_context`.

25. AM_SERVICE_NOT_INITIALIZED — The service is not initialized. This error is thrown by `am_policy` functions if the provided service was not initialized previously using `am_policy_service_init`.

26. AM_INVALID_RESOURCE_FORMAT — This is a plug-in interface error. Implementors of the new resource format may throw this error if the input string does not meet their specified format. This error is thrown by the `am_web` layer, if the resource passed as parameter does not follow the standard URL format.

27. AM_NOTIF_NOT_ENABLED — This error is thrown if the notification registration API is invoked when the notification feature is disabled in the configuration file.

28. AM_ERROR_DISPATCH_LISTENER — Error during notification registration.

29. AM_REMOTE_LOG_FAILURE — This error code indicates that the service that logs messages to Access Manager has failed. The details of this error can be found in the web agent's log file.

APPENDIX E

Configuring the IIS 6.0 64-bit Agent With IIS 7.x With Office SharePoint Server 2007 on Windows Server 2008

The IIS 6.0 64-bit agent is supported on Microsoft IIS 7.0 and IIS 7.5 with Office SharePoint Server 2007 on Windows Server 2008, 64-bit systems. To protect Office With SharePoint Server 2007, the IIS 6.0 64-bit agent is deployed as an ISAPI filter.

You can deploy the IIS 6.0 64-bit agent with the following releases:

- OpenSSO Enterprise 8.0 and later
- Access Manager 7.1 patch 1 and later
- Access Manager 7 2005Q4 patch 7 and later

Contents

Downloading the IIS 6.0 64-bit Agent

▼ To Download the IIS 6.0 64-bit Agent

1 **Login to the server where you want to install the IIS 6.0 64-bit agent.**

2 **Download the agent distribution file from the following site:**

- Download site: `http://www.oracle.com/technology/software/sun_az_index.html`
- Agent name: Access Manager Policy Agent 2.2-04 for Microsoft IIS 6.0/7.0
- Agent distribution file: `iis_v6_x64_WINNT_agent.zip`

3 **Unzip the agent distribution file. The unzipped files are in the following directory:**

AgentHome`/web_agents/iis6_agent`, where *AgentHome* is where you unzipped the file.

For example: `/opt/web_agents/iis6_agent`

Configuring the IIS 6.0 64-bit Agent with IIS 7.x With Office SharePoint Server 2007

To configure the agent, you run the new `CreateConfig.vbs` script (rather than the `IIS6CreateConfig.vbs` script). The following configuration procedure enables the IIS 6.0 Authentication Filter to work with IIS 7.x on Windows 2008 64-bit systems.

▼ To Configure the IIS 6.0 Agent with Office SharePoint Server 2007

1 Enable the IIS 6.0 compatibility features in IIS 7.x.

If IIS 7.x is already installed and running, execute Server Manager > Select IIS Web Server Role > Add Role Services, and Enable the following options. Or, if IIS 7.x is not installed, execute Server Manager > Add Role > Select Web Server (IIS), and then in the Web Server window, select these options:

- IIS Web server role > Application Development > ISAPI Extensions
- IIS Web server role > Application Development > ISAPI Filters
- IIS Web server role > IIS 6 Management Compatibility > IIS 6 Metabase Compatibility

2 Run the `CreateConfig.vbs` script to generate the agent configuration file.

The `CreateConfig.vbs` script is in the *PolicyAgent-base*`\bin` directory. This script works similar to the `IIS6CreateConfig.vbs` script, except that the new configuration file has an empty value for `@AGENT_ENCRYPTED_PASSWORD@`. You must generate the password as described in the next step.

For example, to run the `IIS6CreateConfig.vbs` script:

`cscript CreateConfig.vbs` *configuration-file-name*

When the script prompts you, provide the following values (or accept the default values):

Policy Agent Prompts

- Agent Resource File Name. Default: `IIS6Resource.en`
- Fully Qualified Host Name. For example: `agent-host.example.com`
- Site Name (Site Id). Default Web Site (1)
- Agent Protocol. Default: `http`
- Agent Port Number. Default: `80`
- Agent Deployment URI. Default: `/amagent`

Access Manager or OpenSSO Enterprise Prompts

- Primary Server Host. For example: `amhost.example.com`
- Primary Server Protocol. Default: `http`
- Primary Server Port Number. Default: `58080`
- Primary Server Deployment URI. Default: `/amserver`
- Primary Server Console URI. Default: `/amconsole`

3 **Generate the encrypted agent profile password:**

a. **Encrypt the plain text password using the `cryptit.exe` utility.**

The `cryptit.exe` utility is in the *PolicyAgent-base*`\bin` directory. For example:

```
cryptit.exe agent-profile-password-in-plain-text
```

b. **Copy the encrypted password generated in Step 2 into the agent configuration file:**

```
@AGENT_ENCRYPTED_PASSWORD@ = encrypted-agent-profile-password
```

4 **Also, in the agent configuration file, set the agent profile name by replacing `UrlAccessAgent` with the actual agent profile name. For example:**

```
@AGENT_PROFILE_NAME@ = agent-profile-name
```

5 **Run the `SPAdmin.vbs` script using the configuration file to install the SharePoint Server 2007 filter.**

The `SPAdmin.vbs` script is in the *PolicyAgent-base*`\bin` directory. For example:

```
cscript SPAdmin.vbs -config configuration-file-name
```

6 **Generate the replay password key using `DESgenKey.class`.**

- For Access Manager 7.x deployments:

  ```
  # java -classpath am_sdk.jarPath/am_sdk.jar com.sun.identity.common.DESGenKey
  ```

  where *am_sdk.jarPath* is the complete path to the `am_sdk.jar` file.

- For OpenSSO Enterprise deployments:

  ```
  # java -classpath amserver.jarPath/amserver.jar com.sun.identity.common.DESGenKey
  ```

  where *amserver.jarPath* is the complete path to the `amserver.jar` file.

Executing the `DESgenKey.class` returns a string as output. For example: `c1QBAWv7vHk=`

7 **If you are using the agent with Access Manager, add the replay password key to the `AMConfig.properties` file. For example:**

```
com.sun.am.replaypasswd.key = c1QBAWv7vHk=
```

If you are using the agent with OpenSSO Enterprise, add the key in the OpenSSO Administration console:

a. **Log in to the OpenSSO Administration console.**

b. **Click Configuration, Servers and Sites, and then the** *server-name***.**

c. **Click Advanced and add the following properties:**

- `com.sun.am.replaypasswd.key` with the replay password key value. For example:

  ```
  com.sun.am.replaypasswd.key = c1QBAWv7vHk=
  ```

- `com.sun.am.sharepoint_login_attr_name` with an attribute name in the user repository used by SharePoint Server 2007 to authenticate. For example:

  ```
  com.sun.am.sharepoint_login_attr_name = displayName
  ```

Note. Ignore any warnings after you add these keys.

d. **Click Save.**

8 **Configure the post-authentication plug-in as follows:**

a. **Log in to Access Manager or OpenSSO Administration Console as** `amadmin`**.**

b. **Access the Authentication Post Processing Classes field, depending on the version you are using:**

- Access Manager 7.x: Click Access Control, *realm-name*, Authentication, Advanced Properties, and then scroll down to Authentication Post Processing Classes.
- OpenSSO Enterprise 8.0: Click Access Control, Top Level Realm, Authentication, Advance Properties, and then scroll down to Authentication Post Processing Classes.

c. **Add** `com.sun.identity.authentication.spi.ReplayPasswd` **to the Authentication Post Processing Classes.**

d. **Click Save and log out of the Console.**

9 **Restart the Access Manager or OpenSSO Enterprise server.**

10 **In the agent's** `AMAgent.properties` **file, add the** `com.sun.am.replaypasswd.key` **property. For example:**

```
com.sun.am.replaypasswd.key = c1QBAWv7vHk=
```

11 **For IIS 7.x web sites where the filter is configured, set the authentication method as Basic Authentication.**

12 **If required for you deployment, set the** `NSPR_NATIVE_THREADS_ONLY` **system environment variable to 1 for NSPR threads.**

13 **Restart the IIS 7.x server.**

Index