CA SSO

Agent for Oracle PeopleSoft Guide

r12.51

CA Technologies Product References

This document references the following CA Technologies products:

- CA SSO

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Overview and Architecture 7

Background 7
Increased Security with Tier 2 Integration 7
Architecture 8

Chapter 2: Single Sign-On Security Zones 11

SSO Security Zones and CA SSO Agent for Oracle PeopleSoft 11
Security Zones Benefits 12
Security Zone Basic Use Case 13

Chapter 3: Pre-Installation Steps 15

System Requirements 15
Determining User Attributes of PeopleSoft Username 15

Chapter 4: Installing and Configuring CA SSO Agent for Oracle PeopleSoft 17

Gather Information for the Installation Wizard 17
Run the Installation Wizard on Windows 18
Run the Installation Wizard on UNIX 18
Gather Information for the Configuration Wizard 19
    Gather Configuration Information for Host Registration 19
    Gather Configuration Information for Resource URI & Action 21
    Gather Configuration Information for Agent Name 21
    Gather Configuration Information for Log Configuration 21
Run the Configuration Wizard 22

Chapter 5: Post-Installation Configuration 23

Set Up a CA SSO Response 23
Create a DEFAULT_USER Account in PeopleSoft 24
Enable DEFAULT_USER on Web Server 24
How to Install the PeopleCode to the PeopleSoft Application Designer 24
    Install PeopleCode to PeopleSoft Application Designer 25
    Register PeopleCode for Authentication 25
Test the Installation 26

Disabling Existing Account Passwords 27
Replacing the signon.html File 27

Chapter 6: Upgrading CA SSO Agent for Oracle PeopleSoft 29

How to Prepare for the Upgrade 29
Upgrade CA SSO Agent for Oracle PeopleSoft on Windows 30
Upgrade CA SSO Agent for Oracle PeopleSoft on UNIX 30

Chapter 7: Troubleshooting and Messages 31

Users Challenged by PeopleSoft after Authenticating with CA SSO 31
Verify CA SSO Policies 32
Check the Web Agent Log 32
Examining PeopleCode Logs 33
Not Reaching Stage 1 (No log file) 34
Not Reaching Stage 2 34
Not Reaching Stage 3 34
Not Reaching Stage 4 34
Not Reaching Stage 5 35
Not Reaching Stage 6 35
Examining the Library Logs 35
Log Levels 35
Determine the Level to Set 36

Appendix A: NPSEncrypt and NPSVersion Tools 37

NPSEncrypt Tool 37
NPSVersion Tool 38

Chapter 1: Overview and Architecture

This section contains the following topics:

Background (see page 7)
Increased Security with Tier 2 Integration (see page 7)
Architecture (see page 8)

Background

In an effort to meet the requirements of customers and enable more widespread use of applications, many leading ERP vendors, including PeopleSoft, have developed web-based versions of their applications or web-based front ends for their applications. These web-based front ends provide:

- A standard look-and-feel for employees
- User authentication
- Basic security (such as login by username and password)
- Single Sign-On (SSO) capability for some of these front ends

CA SSO lets you create a centrally managed environment, providing a secure, personalized user experience across all web applications. Through published interfaces, CA SSO can authenticate users to PeopleSoft. This integration enables the PeopleSoft web-based infrastructure and applications to coexist with other portals and web applications, while offering the maximum user experience and benefit.

Increased Security with Tier 2 Integration

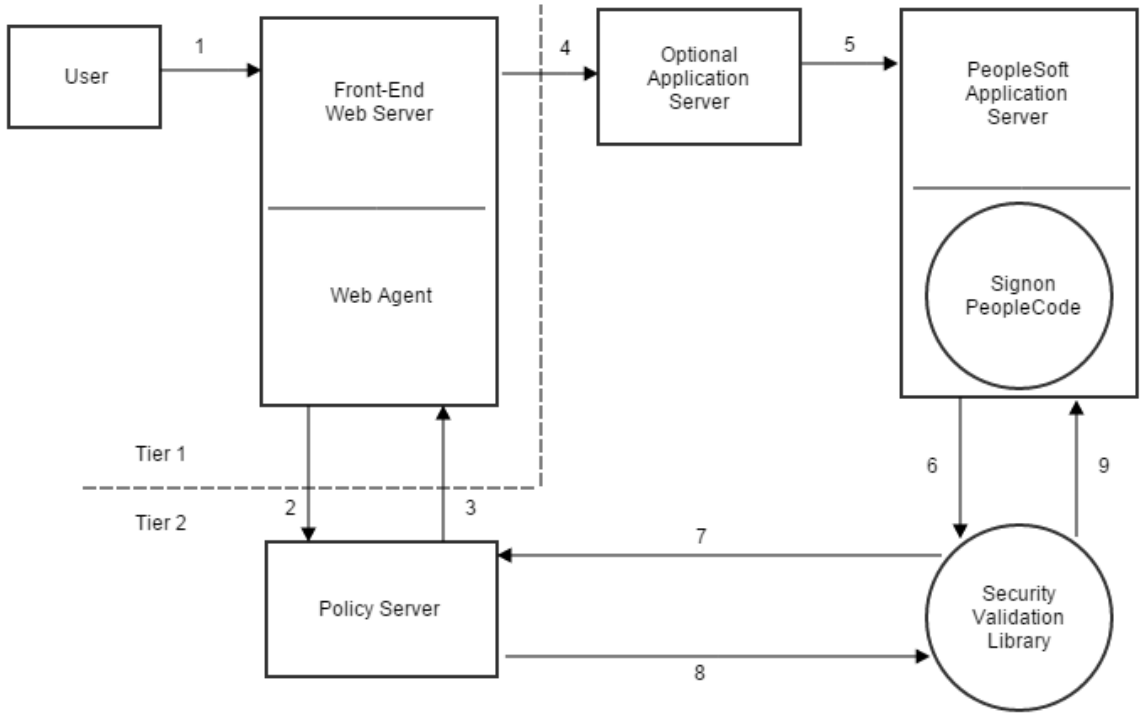
Successful approaches to centrally managed security have typically included the use of a standard Web Agent on a front-end web server and a small piece of PeopleCode that enables PeopleSoft to trust the contents of an HTTP header inserted by CA SSO. Such designs depend upon a level of trust of the front-end web server, a compromise of which could be used to gain elevated access to the system. CA SSO labels these designs *Tier 1* because the point of trust is entirely within the first tier—the front-end web server. In environments where the application server and front-end web server are located entirely on a trusted network, where security requirements are low to moderate, this design is adequate.

In *Tier 2* implementation, the point of trust moves away from the front-end web server and into a more trusted host—in this case the PeopleSoft application server. In Tier 2 integrations, the application that implements the application logic and security is given the ability to call CA SSO APIs to communicate with a Policy server, to validate the information that is presented from the Web Agent.

Many web-based applications use an independent session management scheme, frequently through the use of a cookie. Therefore, the replay prevention and session management logic of CA SSO may be bypassed. The possibility that the CA SSO and application sessions could lose synchronization with each other is one of the main security problems when integrating applications that maintain their own sessions. Tier 2 integration includes a component to prevent such session synchronization issues. The SessionLinker web server plug-in monitors the CA SSO Session ID header and PeopleSoft session cookies. When the two sessions diverge, action is taken to prevent the application from operating until a new session within PeopleSoft is established. The default action is to destroy the PeopleSoft session, causing PeopleSoft to create a new session with the correct user information.

Architecture

The following illustration shows a typical environment.



A description of the numbers in the preceding illustration follows:

1. The user accesses the PeopleSoft application using the front-end web server.
2. Web Agent hosted on the web server intercepts the request, and determines whether the Policy Server is protecting the requested application or resource. If the resource is protected, Web Agent attempts to authenticate and authorize the request for access to PeopleSoft web application resources.
3. The Policy Server verifies access permissions and returns PeopleSoft username back to the Web Agent for use in PeopleSoft.
4. The web server passes user security context to an Optional Application Server for transmittal to the PeopleSoft application server.
5. The application server transmits the request to the PeopleSoft application server. Credentials (user security context information) for the DEFAULT_USER variable are carried to PeopleSoft. The PeopleSoft application server begins a session by invoking Signon PeopleCode.

   Note: The DEFAULT_USER account has no access to the system with or without CA SSO's approval; if someone were to obtain and attempt to use that account without CA SSO, no access would be granted and no data would be available.
6. Signon PeopleCode retrieves the existing CA SSO session information and calls the Tier 2 Validation Library to validate the information.
7. The Validation Library (SMPSLoginLib) calls the Policy Server with existing session information.
8. The Policy Server returns authorization information to the Validation Library.
9. The Validation Library returns the result to PeopleCode, which sets the PeopleSoft security context to that of the user.

Because PeopleSoft deployments vary, the preceding illustration may not reflect any particular customer’s actual environment. For example, the Optional Application Server might not be present. However, the diagram is representative of most typical deployments.

Chapter 2: Single Sign-On Security Zones

This section contains the following topics:

SSO Security Zones and CA SSO Agent for Oracle PeopleSoft (see page 11)
Security Zones Benefits (see page 12)
Security Zone Basic Use Case (see page 13)

SSO Security Zones and CA SSO Agent for Oracle PeopleSoft

This product supports SSO Security Zones. The SSO Security Zones are defined and managed by the Web Agent.

Note: For more information about configuring SSO Security Zones, see the CA SSO documentation.

Users have the ability to define single sign-on security zones within the same cookie domain, representing a single zone, or across multiple cookie domains, representing different zones. As a result, users have single sign-on within the same zone, but may be re-challenged when entering a different zone, depending on the trust relationship defined between the zones. Zones included in a trusted relationship will not rechallenge a user that has a valid session in any zone in the group.

Single sign-on security zones are implemented entirely by Web Agents. Each zone must reside on a separate Web Agent instance. Multiple zones cannot be created on the same Agent instance.

A security zone is identified by cookies generated by the Web Agent. By default, the Web Agent generates two cookies, a session cookie named SMSESSION and an identity cookie named SMIDENTITY. When you configure security zones, the Web Agent can generate session cookies and identity cookies with unique names so that the zone affiliation is reflected in the cookie names.

Security Zones Benefits

The SSO Security Zones feature is intended for use in situations where CA SSO administrators wish to segment their single sign-on environments within the same cookie domain. For example, consider the CA.COM domain. Under standard CA SSO functionality, all CA SSO protected applications in CA.COM would use the cookie SMSESSION to manage single sign-on. Consider the following scenario in which SSO Security Zones do not exist:

1. A user accesses an application (APP1). The user is challenged by CA SSO, logs in to CA SSO, and creates an SMSESSION cookie.
2. The user accesses a second application (APP2) and is once again challenged by CA SSO. (Rules prevent SSO from occurring because the user does not have access to APP2 using the APP1 user credentials.) The user logs in and creates a new SMSESSION cookie overwriting the old one with the new logged in session for APP2.
3. The user now returns to APP1 and is challenged yet again, since the user lost the original APP1 session and the APP2 session might not be accepted for APP1. Therefore, SSO does not occur between APP1 and APP2, causing a very frustrating situation.

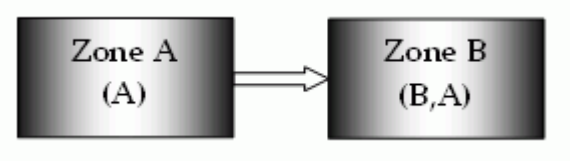
With SSO Security Zones, APP1 can be placed in zone Z1 and APP2 can be placed in zone Z2. Now logging into APP1 creates a Z1SESSION cookie and access to APP2 results in a Z2SESSION cookie. With different names, the cookies no longer overwrite each other so there is only one login per application now, not one for each time the user moves between the applications as in the example above.

Prior to the SSO Security Zones feature, the only way to perform the same grouping of SSO for applications was to create different network domains and therefore different cookie domains (CA1.COM, CA2.COM, and so on), and use various multi-cookie domain configurations with cookie providers. This is not desirable in most enterprises, since using multiple network domains has certain IT maintenance and support consequences.

Security Zone Basic Use Case

Single sign-on can, on a controlled basis, be broken into several security zones that have configurable trust relationships. For example, consider Zone A and Zone B:

- Zone A has only one trusted zone, its own Zone A.
- Zone B has two trusted zones, its own Zone B as well as Zone A.



The trust relationship in the above illustration is indicated by the arrow, meaning that the user sessions established in Zone A can be used for single sign-on in Zone B.

In this example, Zone A might be an administrator-only zone, while Zone B might be a common access zone. An administrator authenticated in Zone A gains access to Zone B without being rechallenged. However, a user authenticated in Zone B is re-challenged when trying to access Zone A.

User sessions in different zones are independent of each other. Suppose a user authenticates in Zone A first, and then authenticates again in Zone B. Two different sessions are created. In fact, the user may have different identities in both sessions. When the user returns to Zone A, the session established in that zone is used.

Consider what would happen if a user is validated using single sign-on in a zone where that user does not yet have a session. If the user authenticates in Zone A and then visits Zone B for the first time, then a user session is created in Zone B, based on the session information in Zone A, possibly updated by the Policy Server. Note that the user session in Zone A is not updated until the user returns to Zone A.

Chapter 3: Pre-Installation Steps

This section contains the following topics:

System Requirements (see page 15)
Determining User Attributes of PeopleSoft Username (see page 15)

System Requirements

The minimum requirements for using this product are as follows:

- Policy Server
- A web server with a Web Agent installed
- PeopleTools
- PeopleSoft components for web access—typically a WebLogic/WebSphere Application Server, the PeopleSoft servlets and support libraries
- Any necessary components to enable the Web Agent's web server to serve as a gateway to the PeopleSoft application suite

Determining User Attributes of PeopleSoft Username

As a pre-installation step, determine the user attributes of the PeopleSoft username.

A user attribute within the directory should contain the user's existing PeopleSoft username. PeopleSoft's existing concept of a user remains intact in this product. The existing roles, access permissions, and other user data remain untouched. CA SSO retrieves a user attribute from the user directory and inserts that value as an HTTP header. Signon PeopleCode retrieves this data, and after verification, provides it to PeopleSoft as the user's identity.

Because the Signon PeopleCode may be customized, the behavior can be significantly altered. For example, the mapping of a CA SSO username to a PeopleSoft username could be created in a database table within PeopleSoft. PeopleCode could use that table to retrieve the user's actual username at signon and alter the call to SetAuthenticationResult to use the new value.

Chapter 4: Installing and Configuring CA SSO Agent for Oracle PeopleSoft

This section contains the following topics:

Gather Information for the Installation Wizard (see page 17)
Run the Installation Wizard on Windows (see page 18)
Run the Installation Wizard on UNIX (see page 18)
Gather Information for the Configuration Wizard (see page 19)
Run the Configuration Wizard (see page 22)

Gather Information for the Installation Wizard

Important! Install this product on the computer that has the PeopleSoft Application Server installed.

The installation wizard requires the following information:

Install Folder

Specifies the directory where the product files are installed.

Default:

Windows:

```
C:\Program Files\CA\peoplesoft
```

UNIX:

```
<home-dir>/CA/peoplesoft
```

PeopleSoft Application Server Home Directory

Specifies the PeopleSoft Application Server home directory.

Default:

Windows:

```
drive:\peoplesoft_application_server_folder\
```

UNIX:

```
/peoplesoft_application_server_folder/
```

Run the Installation Wizard on Windows

The installation wizard installs the agent on your PeopleSoft server.

Follow these steps:

1. Double-click the following file:

   `ca-erp-peoplesoft-<version>-<operating_environmentprocessor_type>.exe`

   Note: To install using console, open a console window and then run the previous command with the -i console option.

   Note: To install unattended, open a console window and then run the previous command with the "-i silent –f <installer_properties_file>" options. The installer properties file (ca-peoplesoft-installer.properties) is located in the install_config_info directory of the product.

   Important! To install unattended, you must install the product using wizard or console once. The installer properties file is required for unattended installation. Before you perform the unattended installation, verify that the installer properties file is updated based on the host where you install the product.

2. Follow the prompts in the wizard.
3. (Optional) Run the configuration wizard when the installation wizard finishes.

More information:

Gather Information for the Configuration Wizard (see page 19)

Run the Installation Wizard on UNIX

The installation wizard installs the product on your PeopleSoft server.

Follow these steps:

1. Execute the following file:

   `ca-erp-peoplesoft-<version>-<operating_environmentprocessor_type>.bin`

   Note: To install using console, open a console window and then run the previous command with the -i console option.

   Note: To install unattended, open a console window and then run the previous command with the "-i silent –f <installer_properties_file>" options. The installer properties file (ca-peoplesoft-installer.properties) is located in the install_config_info directory of the product.

Important! To install unattended, you must install the product using wizard or console once. The installer properties file is required for unattended installation. Before you perform the unattended installation, verify that the installer properties file is updated based on the host where you install the product.

2. Follow the prompts to complete the installation.

Gather Information for the Configuration Wizard

Before you run the PeopleSoft agent configuration wizard, gather the following information:

- Gather Configuration Information for Host Registration (see page 19)
- Gather Configuration Information for Resource URI & Action (see page 21)
- Gather Configuration Information for Agent Name (see page 21)
- Gather Configuration Information for Log Configuration (see page 21)

Gather Configuration Information for Host Registration

To establish a connection between CA SSO Agent for Oracle PeopleSoft host and the Policy Server, you need to register the host with the Policy Server. After registration is complete, the registration tool creates the SmHost.conf file. After this file is created successfully, the client computer becomes a trusted host.

Admin User Name

Specifies the name of a Policy Server administrator allowed to register the host with the Policy Server. This administrator should already be defined at the Policy Server and have the permission to register trusted hosts.

Admin Password

Specifies the Policy Server administrator account password.

Enable Shared Secret Rollover

Specifies that the Policy Server generates a new shared secret periodically, which is used to encrypt the communication to this product.

Trusted Host Name

Specifies a unique name that represents the trusted host to the Policy Server. This name does not have to be the same as the physical client system that you are registering; it can be any unique name, for example, mytrustedhost.

Note: This name must be unique among trusted hosts and not match the trusted host name of any other Agent.

Host Configuration Object

The name of the Host Configuration Object in the Policy Server that defines the connection between the trusted host and the Policy Server. For example, to use the default, enter DefaultHostSettings. In most cases, you will have created your own Host Configuration Object.

Note: This value must match the Host Configuration Object entry preconfigured on the Policy Server.

Policy Server IP Address

The IP address, or host name, and authentication port of the Policy Server where you are registering the host. The default port is 44442. If you do not provide a port, the default is used.

You can specify a non-default port number, but if your Policy Server is configured to use a non-default port and you omit it when you register a trusted host, the following error is displayed:

Registration Failed (bad ipAddress[:port] or unable to connect to Authentication server (-1)

Note also that if you specify a non-default port, that port is used for the Policy Server's authentication, authorization, and accounting ports; however, the unified server responds to any Agent request on any port. The entry in the SmHost.conf file will look like:

policyserver="ip_address,5555,5555,5555"

FIPS Encryption Mode

Determines whether the Agent communicates with the Policy Server using certified Federal Information Processing Standard (FIPS) 140-2 compliant cryptographic libraries.

FIPS Compatibility Mode (Default)

Specifies non-FIPS mode, which lets the Policy Server and the Agents read and write information using the existing CA SSO encryption algorithms. If your organization does not require the use of FIPS-compliant algorithms, the Policy Server and the Agents can operate in non-FIPS mode without further configuration.

FIPS Only Mode

Specifies full-FIPS mode, which requires that the Policy Server and Web Agents read and write information using only FIPS 140-2 algorithms.

Important! A CA SSO installation that is running in Full FIPS mode cannot interoperate with, or be backward compatible to, earlier versions of CA SSO, including all agents, custom software using older versions of the Agent API, and custom software using PM APIs or any other API that the Policy Server exposes. You must re-link all such software with the corresponding versions of the respective SDKs to achieve the required support for Full FIPS mode.

Gather Configuration Information for Resource URI & Action

Specify the resource & action used in CA SSO policies.

Resource

Specifies a resource protected with an authentication scheme.

Example:

```
/psp/ps/?cmd=start
```

Action

Specify the action configured for the above resource in CA SSO policies.

Values: GET, POST

Gather Configuration Information for Agent Name

Specify the agent object name used for protecting the above resource in CA SSO policies.

The agent object need not have 4.x capability because this product uses trusted host communication instead of 4.x communication.

Gather Configuration Information for Log Configuration

Specify the Agent log file path and log level for configuring the agent logging.

0

Disables logging. Log files are not kept.

1

Records errors in initialization and communication only. These messages appear with ERR in the log file.

Important! The highest logging level we recommend for production environments is 1. Higher levels quickly produce large log files.

2

Records informational errors, possibly suggesting the cause of the problem. The specific cause is not always shown. These messages appear with INF in the log file.

3

Records debugging information. These messages appear with DBG in the log file.

4

Records extra information designed to locate problems in the library code. Use this setting only if directed to by CA Technical Support. These messages appear with XXX in the log file.

The configured settings are saved in the peoplecode.txt file.

Run the Configuration Wizard

Use the configuration wizard to configure the product.

Follow these steps:

1. Run the appropriate file for your operating environment:

   In Windows:

   `<agent_install_dir>\install_config_info\ca-peoplesoft-config.exe`

   In UNIX:

   `<agent_install_dir>/install_config_info/ca-peoplesoft-config.bin`

   Note: To configure using console, open a console window and then run the previous command with the -i console option.

   Note: To configure unattended, open a console window and then run the previous command with the “-i silent –f <configuration_properties_file>” options. The configuration properties file (ca-peoplesoft-configuration.properties) is located in the install_config_info directory of the product.

   Important! To configure unattended, you must configure the product using wizard or console once. The configuration properties file is required for unattended configuration. Before you perform the unattended configuration, verify that the configuration properties file is updated based on the host where you configure the product.

2. Provide the required information that was gathered before to complete the wizard.

Chapter 5: Post-Installation Configuration

This section contains the following topics:

Set Up a CA SSO Response (see page 23)
Create a DEFAULT_USER Account in PeopleSoft (see page 24)
Enable DEFAULT_USER on Web Server (see page 24)
How to Install the PeopleCode to the PeopleSoft Application Designer (see page 24)
Test the Installation (see page 26)
Disabling Existing Account Passwords (see page 27)
Replacing the signon.html File (see page 27)

Set Up a CA SSO Response

Within CA SSO, create a set of relevant objects to protect the PeopleSoft application.

- The realm to be protected can be /psp/
- A rule for *
- An HTTP header response named PSUSERNAME containing the user's PeopleSoft Username.

  Note: The name of the HTTP header cannot be changed. Changing the name of this header requires changes to both the PeopleCode and the SmPSLoginLib library.
- Configure the SessionLinker response mentioning the PeopleSoft cookie, PS_TOKEN and the WebLogic/WebSphere session cookie. For more information, see *CA SSO SessionLinker Guide*.

  Note: The exact value of the WebLogic/WebSphere session cookie name should be the value for the portalServletSessionCookieName parameter in the configuration.properties file of the PIA installation.
- A policy that grants access to an appropriate set of users and that links the rule described in this section to the correct CA SSO response.

For example, if the variable name is PSUSERNAME and the attribute name is PeoplesoftUsername, CA SSO protects the entire PeopleSoft environment and returns the user's PeopleSoft username as an HTTP header, named HTTP_PSUSERNAME.

Create a DEFAULT_USER Account in PeopleSoft

The sole purpose of the default user account is for initial communication between PeopleSoft web server and PeopleSoft application server. The default user account enables execution of the Sign-On PeopleCode, which in turn enables the CA SSO integration and retrieves the CA SSO headers, verifies their content, and starts a session. For security reasons, configure this account so that it has no access to the system.

Follow these steps:

1. Invoke PeopleSoft, and navigate to User Profiles.
2. Select Add a New Value, and create a user.
3. In the User ID field, enter DEFAULT_USER.
4. In the Password field, enter the password.
5. Click the Add button.
6. On the ID and Roles tabs, make sure the user does not have access to any data in the system (no privileges and no roles).
7. Save your changes, and exit the PeopleSoft application.

Enable DEFAULT_USER on Web Server

To enable the DEFAULT_USER on PeopleTools version 8.5x onwards, modify the following properties on the security page of the web profile configuration:

- Enable the Public Users—Allow public Access property by selecting the check box.
- Set Public Users—User ID property to DEFAULT_USER
- Set Public Users—Password property to a long string that will be difficult to enter again

Restart the WebLogic/WebSphere server to activate these changes.

How to Install the PeopleCode to the PeopleSoft Application Designer

To install PeopleCode to the PeopleSoft application designer, use the following process:

1. Install the PeopleCode to the PeopleSoft application designer.
2. Register the PeopleCode for authentication.

Install PeopleCode to PeopleSoft Application Designer

A file named peoplecode.txt is installed in the *PeopleSoft Agent Installation Folder*\peoplesoft\PeopleCode folder. The file includes the following:

- Debugging entries, which you can remove before the code is used in a production system
- Declarations for functions included in the Login Library, and a function called SITEMINDER_SSO that makes calls to the library

Follow these steps:

1. Start the PeopleSoft Application Designer program.
2. Log in as a privileged user with write permissions to the FUNCLIB_LDAP record.
3. Open the Open Object window and specify the following:
   - Object Type: Record
   - Name: FUNCLIB_LDAP
4. Click Open.
5. Select the LDAP Auth row, click the right mouse button, and select View PeopleCode.
6. Append the contents of the peoplecode.txt file to the end of the existing PeopleCode source code.
7. Click Save and exit from the Application Designer.

Register PeopleCode for Authentication

Registering PeopleCode enables the SITEMINDER_SSO function.

Follow these steps:

1. In the PeopleSoft program, navigate to SignOn PeopleCode.
2. Select the Invoke as user signing in radio button.
3. Add a new row and enter the specified information in the following fields:
   - Enabled: checkmark
   - Record: FUNCLIB_LDAP
   - Field Name: LDAPAUTH

- Event Name: FieldDefault
- Function Name: SITEMINDER_SSO
- Exec Auth Fail: Leave blank (no checkmark)

4. Click Save, and log out of PeopleSoft.

More information:

Install PeopleCode to PeopleSoft Application Designer (see page 25)

Test the Installation

Access to PeopleSoft through the existing signon.html page should be blocked. Before replacing this page, test the integration (installation).

Follow these steps:

1. In UNIX, use the ca_peoplesoft_env.sh script located in the following directory to export the environment variables:

   `/<agent_install_dir>/`

2. Restart the PeopleSoft application and web servers.
3. Access the PeopleSoft application through the startup page by using the command start.

   Example:

   For PeopleSoft 8.5x onwards:

   `http://peoplesoft.acme.com/psp/ps/?cmd=start`

4. At the CA SSO login screen, enter the CA SSO user credentials.

   After you login to CA SSO, access to PeopleSoft should immediately be granted.

If, at any time, PeopleSoft prompts for credentials, some portion of the integration is not operating correctly.

More information:

Troubleshooting and Messages (see page 31)

Disabling Existing Account Passwords

Because existing accounts are not blocked from logging into PeopleSoft in Tier 2 mode or through some other mechanism, you should lock out passwords for accounts accessing PeopleSoft through the web. See PeopleSoft documentation for assistance in this process.

Replacing the signon.html File

To prevent users from attempting to sign-on to PeopleSoft through existing bookmarks, replace the existing signon.html. Installer installs sample signon.html files in the following location:

<PeopleSoft_Agent_Installation_folder>\peoplesoft\Documentation

These files provide the existing warnings and error messages, while removing the username and password prompts.

Replace the existing file with the relevant sample sign-on file after renaming it signon.html or configure the system to use the new file.

Chapter 6: Upgrading CA SSO Agent for Oracle PeopleSoft

This section contains the following topics:

How to Prepare for the Upgrade (see page 29)

How to Prepare for the Upgrade

This section describes the steps you need to take before you start the upgrade process.

Follow these steps:

1. Verify that the PeopleSoft application server and web servers are stopped before the upgrade.

   Important! On AIX, after stopping the PeopleSoft application server and web servers, run the slibclean utility as the root user to remove unused shared libraries from memory in AIX.

2. Upgrade is supported only if the operating environments for both old version and new version of the product are the same.

   Note: For more information, see the Platform Support Matrix on the CA Support website.

3. Upgrade the other CA SSO components (such as the Policy Server and Web Agent) in your environment first.

   Note: For more information about upgrading the CA SSO components, see the CA SSO documentation.

4. Gather the information for the upgrade.
5. To upgrade the product, run the installation wizard.
6. Configure your upgraded product.

Upgrade CA SSO Agent for Oracle PeopleSoft on Windows

The installation wizard detects the older version and upgrades the software.

Follow these steps:

1. Double-click the following file:

   `ca-erp-peoplesoft-<version>-<operating_environmentprocessor_type>.exe`

   Note: To upgrade using console, open a console window and then run the previous command with the -i console option.

   Note: To upgrade unattended, open a console windows and then run the previous command with "-i silent –f <installer_properties_file>" options. The installer properties file (ca-peoplesoft-installer.properties) is located in the install_config_info directory of the product.

2. Use the gathered information to complete the wizard.
3. (Optional) Run the configuration wizard after completing the installation wizard.

More information:

Gather Information for the Configuration Wizard (see page 19)

Upgrade CA SSO Agent for Oracle PeopleSoft on UNIX

The installation wizard detects the older version and upgrades the software.

Follow these steps:

1. Execute the following file:

   `ca-erp-peoplesoft-<version>-<operating_environmentprocessor_type>.bin`

   Note: To upgrade using console, open a console window and then run the previous command with the -i console option.

   Note: To upgrade unattended, open a console windows and then run the previous command with "-i silent –f <installer_properties_file>" options. The installer properties file (ca-peoplesoft-installer.properties) is located in the install_config_info directory of the product.

2. Use the gathered information to complete the wizard.

More information:

Gather Information for the Configuration Wizard (see page 19)

Chapter 7: Troubleshooting and Messages

This section contains the following topics:

Users Challenged by PeopleSoft after Authenticating with CA SSO (see page 31)
Verify CA SSO Policies (see page 32)
Check the Web Agent Log (see page 32)
Examining PeopleCode Logs (see page 33)
Examining the Library Logs (see page 35)

Users Challenged by PeopleSoft after Authenticating with CA SSO

Valid on Weblogic server

Symptom:

Users are challenged by for their credentials, even after successfully authenticating with CA SSO. This occurs when a default setting in the config.xml file of Weblogic server is not changed when CA SSO is used.

Solution:

Change the default setting with the following steps:

1. Locate the following tag in the config.xml file:

   ```
   <enforce-valid-basic-auth-credentials>true</enforce-valid-basic-auth-credentials>
   ```

   Note: If the previous tag does not exist, add it *before* the following tag in the config.xml file:

   ```
   </security-configuration>
   ```

2. Change the value to false, as shown in the following example:

   ```
   <enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credentials>
   ```

3. Restart the Weblogic server to apply the changes.

Verify CA SSO Policies

Perform the following procedure to verify the CA SSO policies.

Follow these steps:

1. Click Start, Programs, CA SSO, CA SSO Test Tool.
2. Enter the correct Agent name and host configuration file (SmHost.conf).
3. Click Connect.
4. Enter the correct resource (For PeopleSoft 8.5 onwards, /psp/ps/?cmd=start), the action GET, and click IsProtected.
5. Enter a valid CA SSO username and password, and click IsAuthenticated.
6. Click IsAuthorized.

Note: If a red indicator appears or the PSUSERNAME and/or NPSSessionLinker responses do not appear in the Attributes box, examine the Policy Server configuration and logs. The PSUSERNAME response should hold a valid PeopleSoft username.

Check the Web Agent Log

If the web browser shows a 500 Server Error page, or if the web browser continuously returns to the login page, check the Web Agent log.

For a solution to these problems, see the CA SSO documentation.

Further diagnosis is beyond the scope of this document, but an examination of the Web Agent log file might reveal the solution.

Examining PeopleCode Logs

The PeopleCode logs six stages during a successful authentication, as shown in the following table. The stage numbers are not actually called out in the log.

Stage	Log Text	Description
1	Calling SmSSOSetLogging	PeopleCode is making its first call to the Validation Library (SMPSLoginLib). This is intended as a checkpoint to indicate that PeopleCode has been called, and is attempting to do its work.
2	Done calling DLL	PeopleCode has succeeded in calling the first DLL entry point, indicating that it has located, loaded, and used the DLL.
3	Authentication request received for DEFAULT_USER...	A login attempt has occurred for CA SSO integration, thus the remaining PeopleCode should be run.
4	Checking user KC0003	PeopleCode is checking the validity of the user who just attempted to log in.
5	User KC0003 OK with CA SSO	The Validation Library has indicated that the information provided matches the expected values and that the login attempt should be allowed.
6	Finished processing request	Final checkpoint before PeopleCode returns control to the PeopleSoft application server. PeopleSoft controls authorization only after a successful authentication attempt from PeopleCode.

Not Reaching Stage 1 (No log file)

If no log file is created, check the following:

- Is PeopleCode in the correct record? See the following steps:
  - Step 4 in Installing PeopleCode to the Application Designer (see page 25)
  - Step 3 in Registering PeopleCode for Authentication (see page 25)
- Is PeopleCode enabled in Signon PeopleCode? See Step 3 in Registering PeopleCode for Authentication (see page 25).
- Does PeopleCode open an appropriate log file? If not, check the log file path and make sure it is valid and the folder exists.

Not Reaching Stage 2

Reaching this stage means that PeopleCode is being called when a user logs in. If no further entries appear in the log, the PeopleSoft application server probably cannot locate the Validation Library. Verify that the Validation Library is installed in the correct directory.

Not Reaching Stage 3

Stage 3 indicates that an authentication attempt has been received for the user DEFAULT_USER. If the Stage 3 entry does not appear in the log file, make sure the default user is enabled.

More information:

Enable DEFAULT_USER on Web Server (see page 24)

Not Reaching Stage 4

If the text failed to get header PSUSERNAME appears instead of the text Checking user, the CA SSO HTTP header response PSUSERNAME does not appear in the HTTP request. Verify the CA SSO response through the use of the CA SSO Test Tool and the Web Agent log files.

Not Reaching Stage 5

If the following text appears, Login Library is unable to verify the user's session information:

```
User XXX, session … not acceptable… rejecting
```

Examine the Login Library log file for any indication of configuration problems.

More information:

Examining the Library Logs (see page **35**)

Not Reaching Stage 6

If the PeopleSoft application server does not reach Stage 6, it may have crashed between Login Library's successful response and the end of the SITEMINDER_SSO function. Examine PeopleCode for a potential cause, which is likely to be something external to any of the PeopleSoft agent binaries.

Examining the Library Logs

PeopleCode contains the function SmSSOSetLogging, which takes two parameters:

- An integer indicating the logging level (from 0 to 4) with higher numbers indicating larger amounts of information
- The path to a log file

Log Levels

The following table shows the log level parameter values along with their meanings and indicators.

Level	Log Indicator	Meaning
0	*No indicator; no log file*	None; the log file is off
1	ERR	Errors only; errors in initialization and communication are logged
2	INF	Informational; at this level information indicating the root cause of the problem is shown. The specific cause of the problem will probably not appear.

Level	Log Indicator	Meaning
3	DBG	Debug; information not typically useful in production environments.
4	XXX	Extra; the information shown is intended as an aid in locating problems in the Login Library code itself, and intended for CA Technical Support.

Determine the Level to Set

Perform the following procedure to determine the level to set.

Follow these steps:

1. Set the log level to 2 and examine the logs. The most common problems reported to CA SSO are the result of errors in the configuration of agent name and host configuration object. At log level 2 many of these errors will appear and the solution will be obvious— for example a typical error is “Failed to connect agent - check host configuration object and agent name,” implying that the agent name or host configuration object is the likely root cause of the problem.
2. Increase the log level to 3 and examine the logs. When the problem’s cause does not appear under log level 2 and the only error text appearing is “Session not valid - returning –1,” log level 3 will reveal additional information, including the cause.

Appendix A: NPSEncrypt and NPSVersion Tools

This section contains the following topics:

NPSEncrypt Tool (see page 37)
NPSVersion Tool (see page 38)

NPSEncrypt Tool

Sometimes, *secret* values must be stored in a configuration file. For security purposes, you might want to encrypt and store the encrypted form of these secret values. To do this, use the NPSEncrypt tool. When a setting allows encrypted values to be used, this product decrypts it before use. If the setting is not encrypted, the value entered will be used as is.

The NPSEncrypt utility takes plain text entered on the command line, encrypts it, and prints the result on the screen. The resulting encrypted text can be copied and pasted wherever it is needed.

A product that allows an encrypted value automatically decrypts it when needed.

Run the NPSEncrypt utility from the directory where it exists. The default location of the utility is as follows:

```
<PeopleSoft_agent_install_folder>\tools\
```

To encrypt a value, navigate to the following directory and type the NPSEncrypt command followed by a space and followed by the text to be encrypted:

```
/<agent_install_dir>/tools
```

```
C:\>npsencrypt secret
[NPSEncrypt Version 1.1 - NPSEncrypt Revision 1]
[NDSEnc-B]CKtyevyWkrF24Aj9Ly+xEQ==
```

In this case the encrypted form of secret is:

```
[NDSEnc-B]CKtyevyWkrF24Aj9Ly+xEQ==
```

When you copy and paste, grab the entire line, including [NDSEnc-].

NPSEncrypt will encrypt the same text to many different cipher text values. Use any of the values, for example:

```
C:\>npsencrypt secret
[NPSEncrypt Version 1.1 - NPSEncrypt Revision 1]
[NDSEnc-C]iQO2KVyRN2fB4tMwjtgRYQ==
C:\>npsencrypt secret
[NPSEncrypt Version 1.1 - NPSEncrypt Revision 1]
[NDSEnc-C]FWhVC+MiA7aNnA87szw76g==
C:\>npsencrypt secret
[NPSEncrypt Version 1.1 - NPSEncrypt Revision 1]
[NDSEnc-B]PD24A2Iz6H+KeDh7j4zUIg==
```

NPSVersion Tool

Use the NPSVersion tool to extract version information from many CA products. To use this tool, type the NPSVersion on a command line followed by a space and the name of the executable whose version information you want, for example:

```
C:\> NPSVersion sessionlinkd
[NPSVersion Version 1.0 - NPSVersion Revision 1]
sessionlinkd         -   Package: NPSSessionLinker V1.3
sessionlinkd        - Component: SessionLinker daemon V1.3.2 (Jul 14 2003 20:26:16)
sessionlinkd         -  Platform: AIX
C:\>
```

Package

Refers to the version of Product, in this case the SessionLinker version 1.3 product.

Component

Refers to the actual part of the product that is enclosed within this specific file. It is not uncommon for this version number to be larger than the *Package* version. This is usually due to *Component* having one of more bugs repaired or minor enhancements added that did not require the entire Package to be rebuilt or renumbered.

You may use the NPSVersion tool on one platform to extract information for a product built for any other platform. The actual information displayed might differ in format and content from what is shown above, but the relevant lines when discussing any issues with Support are Package and Component. Each line includes a version number.