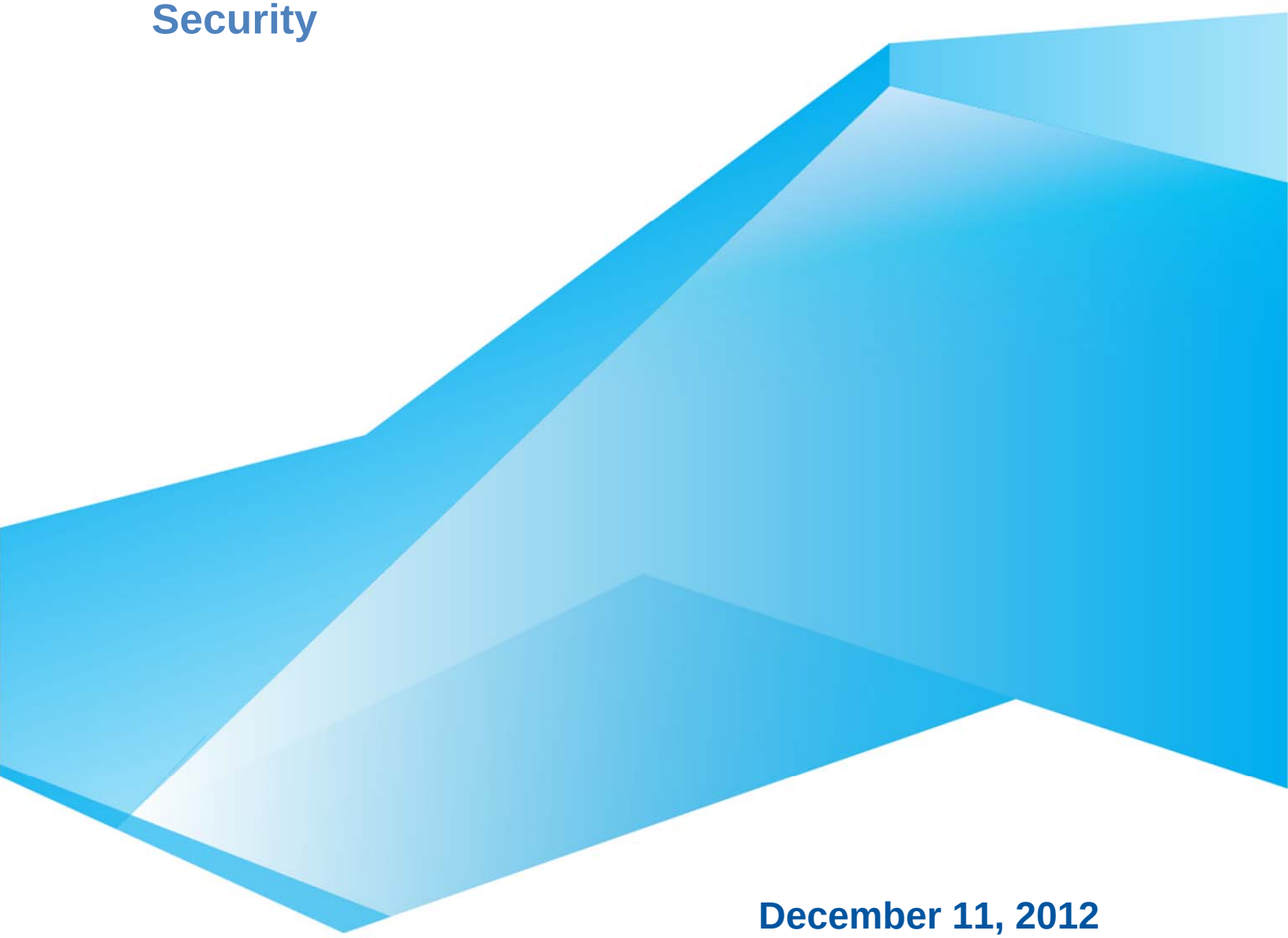
Deltek Know more. Do more.™

Deltek Costpoint® 7.0

Security



December 11, 2012

While Deltek has attempted to verify that the information in this document is accurate and complete, some typographical or technical errors may exist. The recipient of this document is solely responsible for all decisions relating to or use of the information provided herein.

The information contained in this publication is effective as of the publication date below and is subject to change without notice.

This edition published December 2012.

Contents

Overview

Security is a critical part of any application. Applications must be secured against disclosure of confidential information, modification, or destruction of data, misappropriation of resources, and compromise of accountability. Implementing security measures, such as authentication, authorization, integrity, confidentiality, and non-repudiation, can secure applications.

This document details how the preceding principles are designed and implemented in Deltek Costpoint®. Costpoint uses both Oracle WebLogic® and Java™ Authentication and Authorization Service (JAAS) frameworks to authenticate and authorize clients who are interactive users, Web service clients, and application clients.

Authentication

Authentication verifies the identity of an application user. Costpoint performs authentication using a login process during which the user supplies credentials, such as a username and password combination. When the user has been authenticated, Costpoint associates a set of identities (also known as *principals*) with that user. For example, the user's identities can include his or her username and group membership.

Multiple Authentication Methods Available

Costpoint supports multiple authentication methods so that:

- Internet users and local area network users can access Costpoint simultaneously.

  Generally, the following categories of users access Costpoint:

  - **In-house users** — Users who are registered in the company network (Windows Active Directory) and who typically log into Costpoint only after passing through local network authentication
  - **In-house users who travel occasionally and *Consultants*** — In-house users who occasionally log into Costpoint from remote sites without being authenticated in the company network
  - **Remote Office users** — Users who are not registered in the company network and who typically log into Costpoint via remote sites only
- Companies using Costpoint often have unique security requirements. For example, one company's rules may require users to be authenticated on their network before accessing Costpoint. Likewise, a company's rules may require Windows Active Directory to log users onto Costpoint.

Description of Authentication Methods

Costpoint Database

This authentication method supports all Costpoint users—it is the default authentication method. With this method, all passwords are checked against the Costpoint database.

Costpoint Database verification requires no extra configuration efforts. To log into Costpoint using this method, go to the costpoint.htm login page (for example, http://costpointswebserver/costpoint.htm).

Single Sign-on

This authentication method supports only In-house users who are currently logged into the company network (via Windows Active Directory). This method allows users to log into Costpoint without providing a user ID and password on the Costpoint Login screen.

Single Sign-On verification requires special WebLogic Server and Windows Active Directory configuration steps. To log into Costpoint using this method, go to the costpoint.htm login page (for example, http://costpointswebserver/costpoint.htm).

Active Directory

This authentication method supports In-house users who are registered in the company network (via Windows Active Directory) but not necessarily logged into the company network. A user is required to provide a user ID and password on the Login screen to access Costpoint. This authentication method verifies passwords against the Windows Active Directory.

Active Directory verification requires special WebLogic Server and Windows Active Directory configuration steps. To log into Costpoint using this method, go to the costpoint.htm login page (for example, http://costpointswebserver/costpoint.htm).

Single Sign-on or Active Directory

This authentication method supports In-house users and Consultants. It gives users two options for accessing Costpoint:

- **When a user is already logged into the company network (Single Sign-on)** — The user can access Costpoint without providing a user ID and password on the Costpoint **Login** screen.
- **When a user is not logged into the company network (Active Directory)** — The user can access Costpoint by entering a user ID and password on the Costpoint Login screen. This method verifies passwords against the Windows Active Directory.

Single Sign-on or Active Directory authentication requires special WebLogic Server and Windows Active Directory configuration steps. To log into Costpoint using this method, go to the costpoint.htm login page (for example: http://costpointswebserver/costpoint.htm).

Windows Domain and Active Directory

This authentication method supports In-house users who are currently logged into the company network (Windows Active Directory). A user is required to provide a user ID and password on the Costpoint Login screen. This method verifies passwords against Windows Active Directory.

Windows Domain and Active Directory authentication requires special WebLogic Server and Windows Active Directory configuration steps. To log into Costpoint using this method, go to the costpoint.htm login page (for example, http://costpointswebserver/costpoint.htm).

Windows Domain and Costpoint Database

This authentication method supports In-house users who are currently logged into the company network (Windows Active Directory). A user is required to enter a user ID and password on the Costpoint Login screen. This method verifies passwords against the Costpoint database.

This authentication method requires special configuration steps to be performed on WebLogic Server and Windows Active Directory. To log into Costpoint using this method, go to the costpoint.htm login page (for example, http://costpointswebserver/costpoint.htm).

Client Certificate

This authentication method is used for identifying the user by his or her X.509 certificate. This is a special use of the Secure Sockets Layer (SSL), where both the server and the user are identified by their own certificates. As a result, this is a very strong form of authentication which guarantees that a user can log into Costpoint only from a machine that has a valid certificate installed. All communication between server and client is encrypted.

This method is targeted to support all Costpoint users. It requires special configuration steps to be performed on WebLogic Server and on the client machine. To log into Costpoint using this method, go to the costpoint.htm login page and log in via https protocol (for example, https://dltkas44:7002/costpoint.htm).

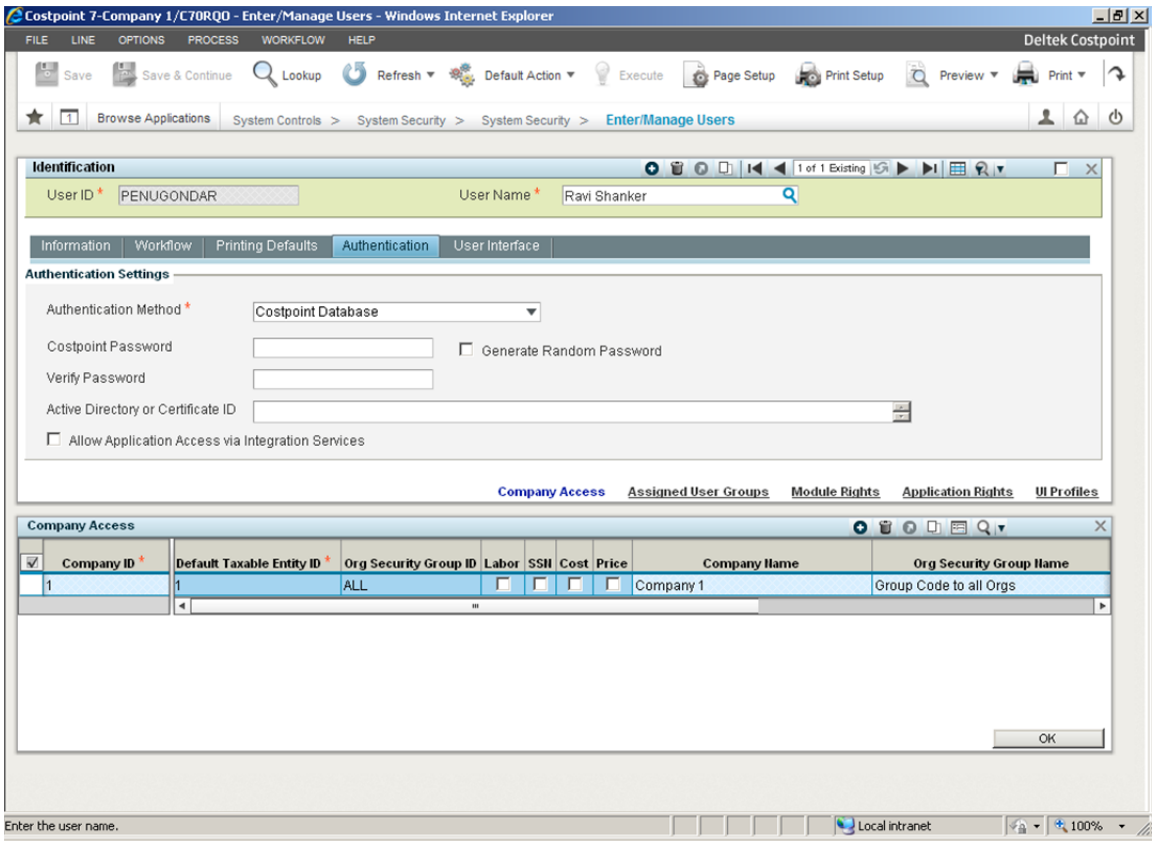
Assign Authentication Methods to Users

Each Costpoint user has an assigned authentication method. You can assign authentication methods to users using the Authentication tab of the Manage Users screen (SYMUSR).

- The default authentication method is Costpoint Database.
- An Active Directory ID must be entered for the following authentication methods: Single Sign-on, Active Directory, Single Sign-on or Active Directory, Windows Domain and Active Directory, and Windows Domain and Costpoint Database.
- Using certain authentication methods (including Single Sign-on, Active Directory, Single Sign-on or Active Directory, Windows Domain and Active Directory) requires special configuration steps to be performed by your company's IT team on the WebLogic Server and Windows Domain Controller machine.

To assign an authentication method to a user, complete the following steps:

1. Click **Administration » Security » System Security » Manage Users**.



2. Select a user.
3. Click the Authentication tab.
4. Enter the following Authentication Settings:
   - **Authentication Method** — Select the user authentication method (for example, **Costpoint Database** or **Single Sign-On**).
   - **Costpoint Password** — Enter the user password (required for Costpoint Database and Windows Domain and Costpoint Database authentication methods).

- **Verify Password** — Re-enter the same password to verify its accuracy (required for Costpoint Database authentication and Windows Domain and Costpoint Database authentication methods).
- **Active Directory or Certificate ID** — Select the user ID for login to the Windows Domain (required for Single Sign-on, Active Directory, Single Sign-on or Active Directory, Windows Domain and Active Directory, Windows Domain, and Costpoint Database authentication methods).
- **Allow Application Access via Integration Service** — Select this check box to allow integration clients (Web service clients, application clients, and any other programs or services) to log into Costpoint with a Costpoint user ID (required).

Security Realm and Authentication Providers

The WebLogic Server System Administrator configures the security realm (for example, **CPRealm**) to support Costpoint authentication. **CPRealm** is a chain of authentication providers in which each provider or set of providers is responsible for authenticating users of certain types, including the following: SSO, Windows Active Directory, and Costpoint Database.

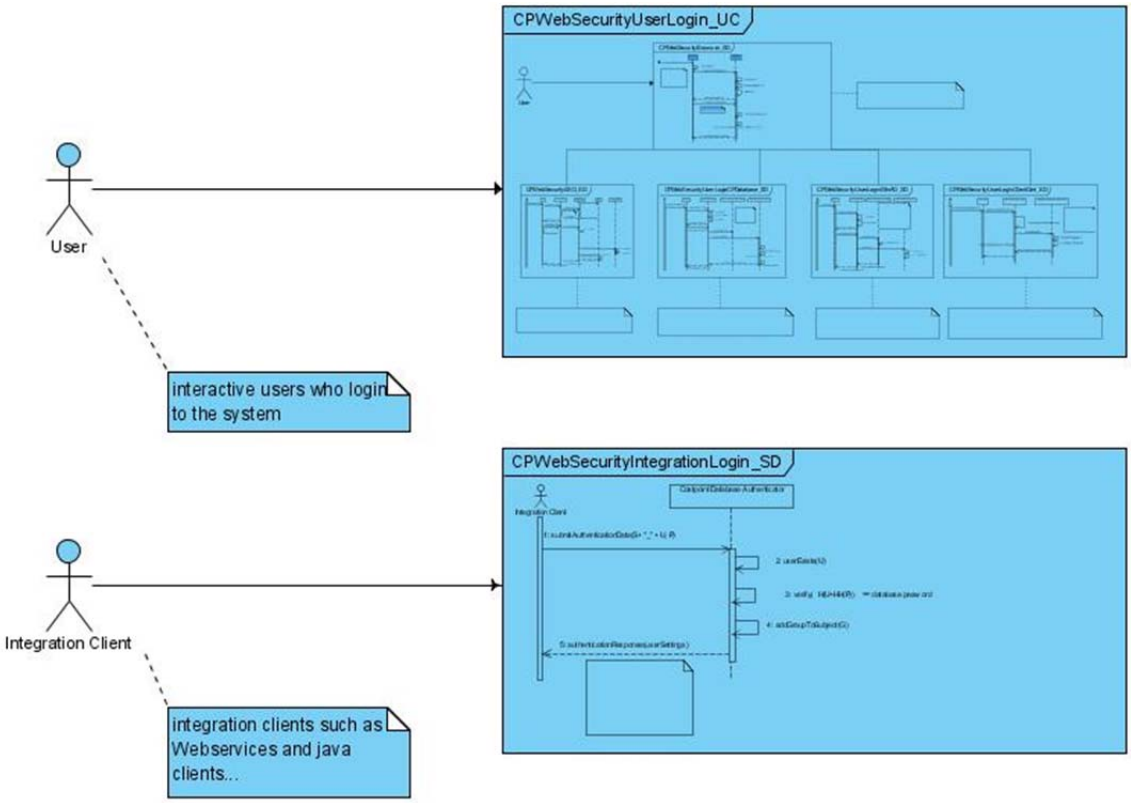
The following providers are preconfigured in following order to support all Costpoint authentication methods:

1. CPRDBMS Authenticator (Control Flag: Optional)
2. Negotiate Identity Assertor (Control Flag: Optional)
3. CPRDBMS SSO Helper Authenticator (Control Flag: Optional)
4. Default (Embedded LDAP) Authenticator (Control Flag: Optional)

Authentication data or identity information (for example, user ID and password) passes through each provider. Authentication providers verify the identity information and make it available to other providers in the chain and to other components in Costpoint.

Authentication Process at Login

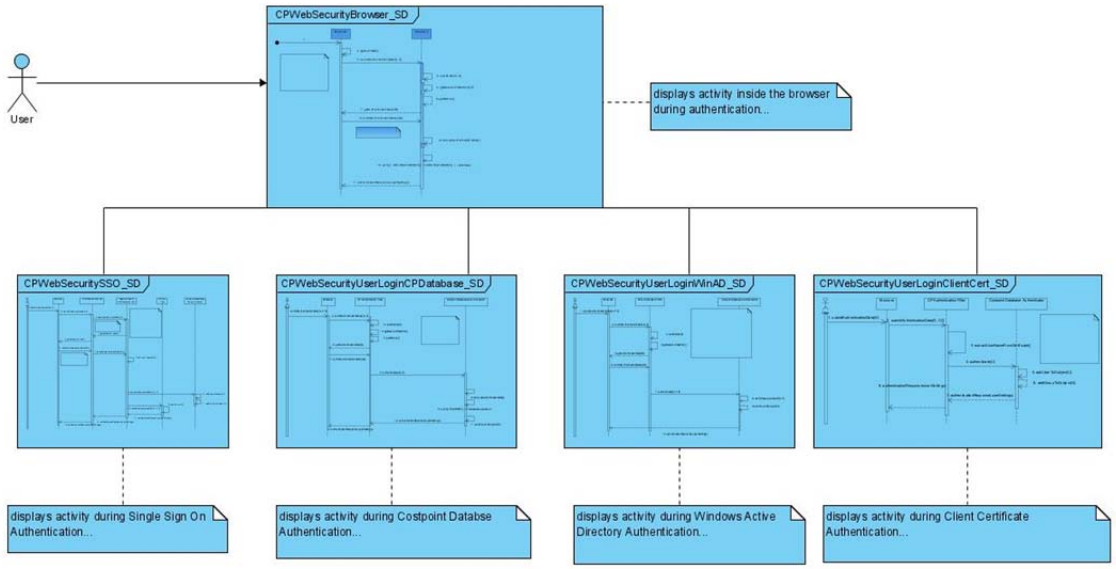
The Authentication process starts with client login requests. In Costpoint, we distinguish two types of clients: **real** (interactive) users and **integration** clients (Web service clients, application clients, and any other programs or services).



Interactive User Login

Four major processes occur during real user login:

- Costpoint Database authentication
- Client Certification authentication
- Windows Active Directory authentication
- Single Sign-On authentication.

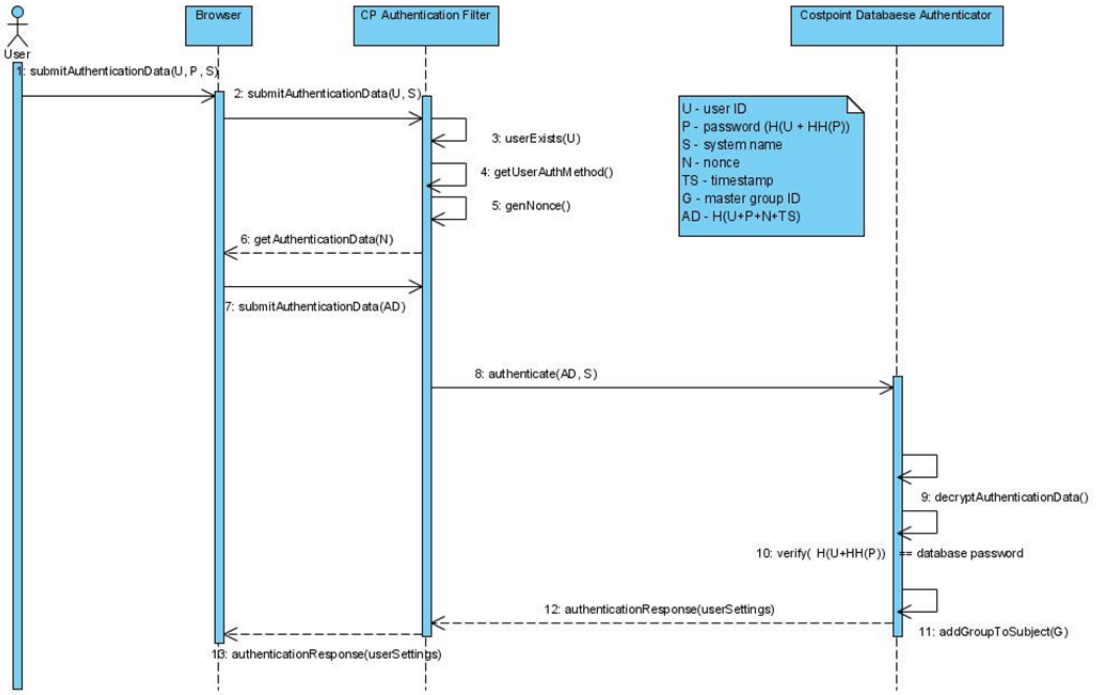


Costpoint Database Authentication

1. On the Costpoint Login page, the user enters authentication data (user ID/password/system), and clicks **Login**.
2. The Login page sends the user's authentication data to the server.
3. The Costpoint Security Filter checks the validity of the user ID.
4. The Costpoint Security Filter verifies that the user's authentication data matches the Costpoint Database.
5. The Costpoint Security Filter generates a random challenge number (nonce).
6. The Costpoint Security Filter sends a challenge nonce back to the Login page.
7. The Login page builds and encrypts an expression that contains the user ID, password, nonce, and timestamp and then sends the encrypted expression to the server.
8. The Costpoint Security Filter calls the WebLogic framework to authenticate the data. The request goes to the CPRDBMS Authenticator.
9. The CPRDBMS Authenticator decrypts the authentication data.
10. The CPRDBMS Authenticator checks the user ID/password combination against the database.

11. If authentication succeeds, the CPRDBMS Authenticator creates a JAAS subject and passes it back to the Costpoint Security Filter. If authentication fails, the CPRDBMS Authenticator will return an error.
12. The Costpoint Security Filter generates and sends either an “Ok” response or an “Error” response to the Login page.

    If authentication succeeds, the user receives access to Costpoint.

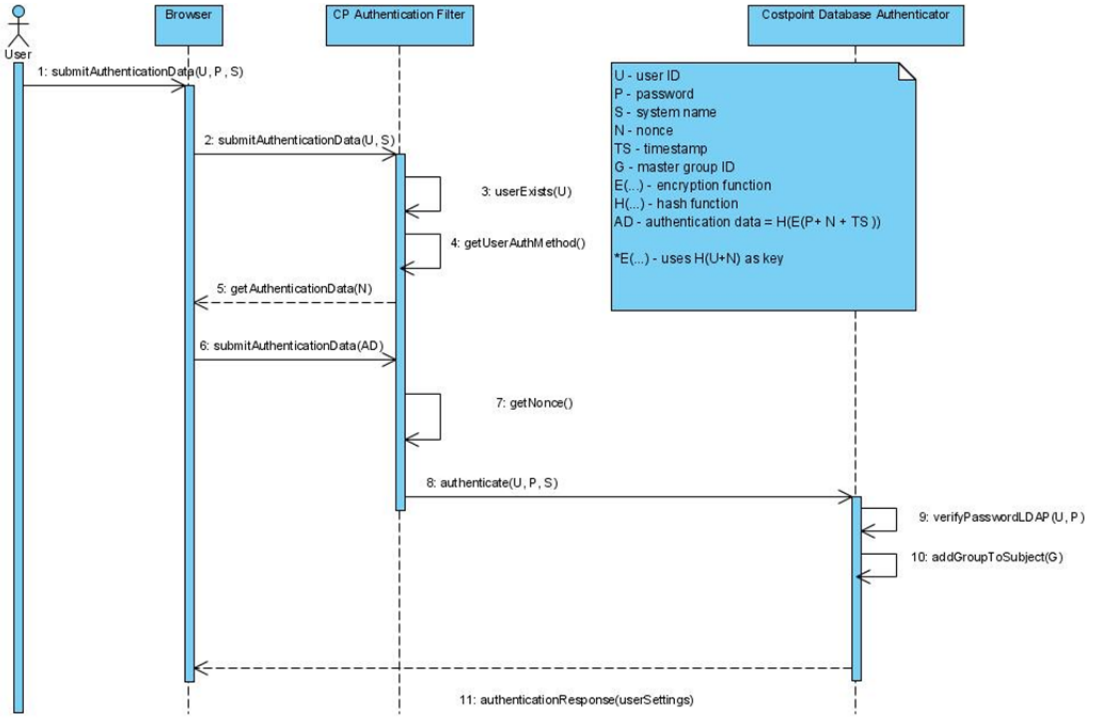


Windows Active Directory Authentication

1. On the Costpoint Login page, the user enters authentication data (user ID/password/system), and clicks **Login**.
2. The Login page sends the authentication data to the server.
3. The Costpoint Security Filter checks the user ID validity.
4. The Costpoint Security Filter verifies that the user authentication method is Windows Active Directory.
5. The Costpoint Security Filter requests the authentication data from the client. The filter generates a random challenge number (nonce) and sends the nonce to the client.
6. The Login page builds and encrypts an expression that contains the password, nonce, and timestamp and sends the encrypted expression to the server.
7. The Costpoint Security Filter decrypts the authentication data and verifies that the nonce received from the client matches the nonce previously generated on the server and then calls the WebLogic framework to authenticate the data.
8. The request goes to the CPRDBMS Authenticator.
9. The CPRDBMS Authenticator verifies the user ID/password combination in Windows Active Directory Server.

10. If authentication succeeds, a JAAS subject is created and passed back to the Costpoint Security Filter. If authentication fails, an error is generated.
11. The Costpoint Security Filter generates and sends either an “Ok” response or an “Error” response to the Login page.

    If authentication succeeds, the user receives access to Costpoint.



Single Sign-On Authentication

1. On the Costpoint Login page, the user enters authentication data (the system name only, no user ID or password needed), and clicks **Login.**

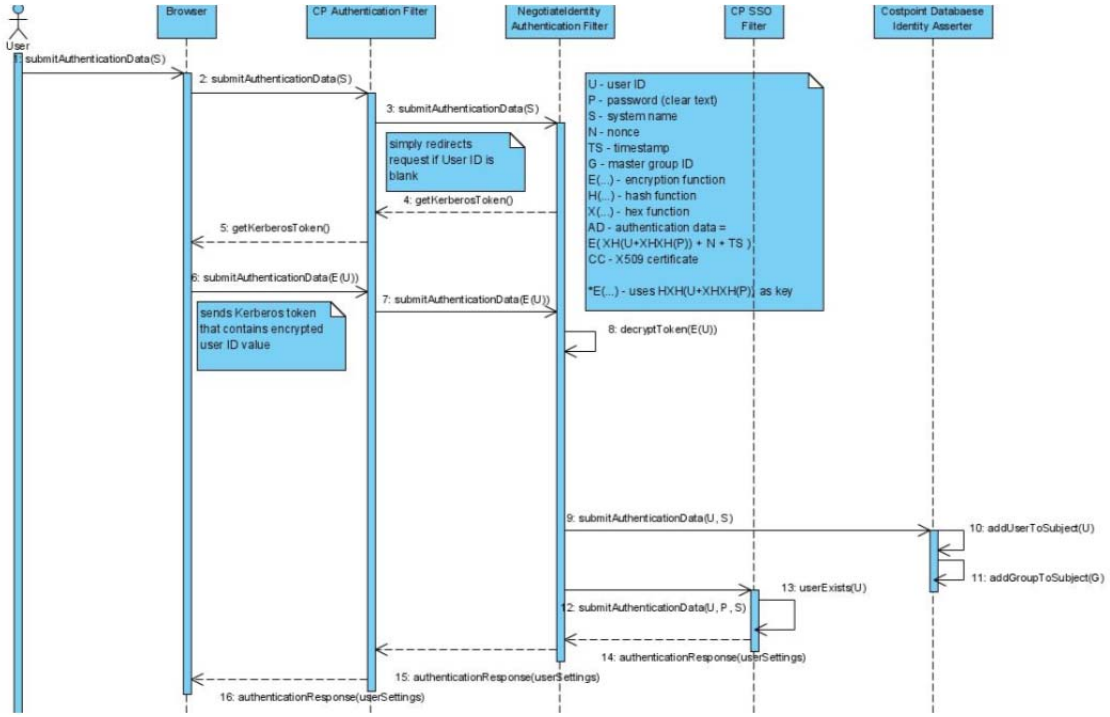
   For Single Sign-On authentication, the user must be logged into the Windows Domain (company network).

2. The Login page sends authentication data (system name) to the server.
3. The Costpoint Security Filter redirects the request to the Negotiate Identity Filter (WebLogic’s SSO filter).
4. The Negotiate Identity Filter communicates back to the browser to obtain an encrypted token (Kerberos string token) containing a user ID. The request goes through the Costpoint Security Filter.
5. The Costpoint Security Filter redirects the request to the browser.
6. The browser sends the encrypted token to the Negotiate Identity Filter. The request goes through the Costpoint Security Filter.
7. The Costpoint Security Filter redirects the request to the Negotiate Identity Filter.
8. The Negotiate Identity Filter decrypts the token and extracts the user ID.

9. The Negotiate Identity Filter calls the WebLogic framework to authenticate (assert identity) the user ID without a password.

> In identity assertion mode, no password is needed because successful decryption establishes trust between the server and the client.

10. The request passes to the CPRDBMS Authenticator.
11. The CPRDBMS Authenticator trusts the identity extracted by the Negotiate Identity Filter and adds the identity to the JAAS subject.
12. The CPRDBMS Authenticator adds the master Costpoint Group “ApplicationUserGroup” to the JAAS subject.
13. The request passes to the CPRDBMS SSO Filter, which verifies whether or not the JAAS subject contains a principal. If yes, the SSO Filter checks if the user ID is valid in the Costpoint database.
14. The SSO Filter checks the Costpoint database to see if the user ID from the JAAS subject is a valid user and if the user’s authentication method is SSO.
15. The CPRDBMS SSO Filter generates and sends either an “Ok” response or an “Error” response to the Login page.
16. If authentication succeeds, the user receives access to Costpoint.



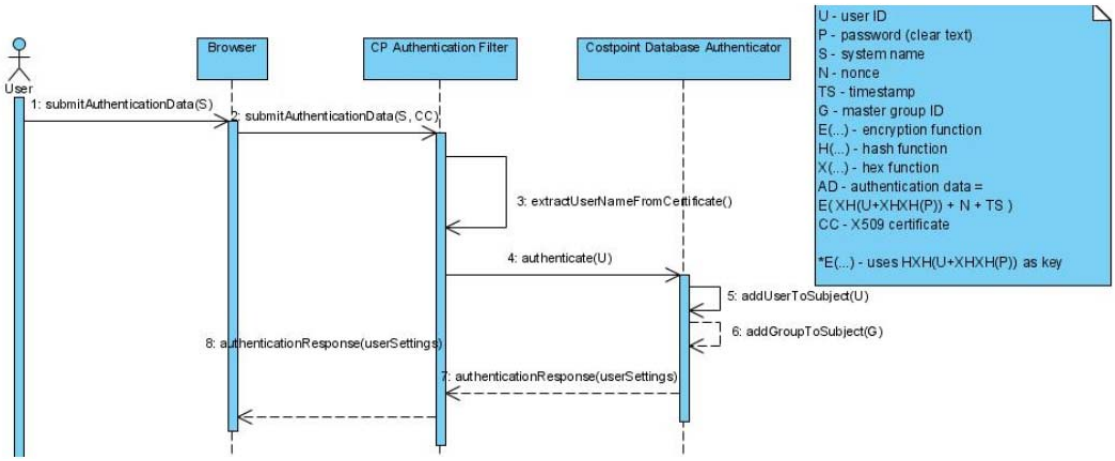
Certificate SSO Authentication

1. On the Costpoint Login page, the user enters authentication data (the system name only, no user ID or password needed), and clicks **Login**., The user must have a valid X509 certificate installed on the machine he or she is using to log in.
2. The Login page sends authentication data (system name and X509 client certificate) to the server.
3. The Costpoint Security Filter extracts the user ID from the X509 certificate. It then verifies that the user exists in the database and that the user authentication method is Client Certificate.
4. The Costpoint Security Filter calls the WebLogic framework to authenticate (assert identity) the user ID without a password.

> In identity assertion mode, no password is needed because successful decryption establishes trust between the server and the client.

5. The request passes to the CPRDBMS Authenticator.
6. The CPRDBMS Authenticator trusts the identity extracted by the Costpoint Security Filter and adds the identity to the JAAS subject.
7. The CPRDBMS Authenticator adds the master Costpoint Group “ApplicationUserGroup” to the JAAS subject.
8. The CPRDBMS Authenticator returns to the Costpoint Security Filter.
9. The Costpoint Security Filter generates and sends either an “Ok” response or an “Error” response to the Login page.

   If authentication succeeds, the user gains access to Costpoint.



Integration Client Login

Integration clients, such as Web services and Java application clients, log into Costpoint programmatically.

- Integration clients must use a real Costpoint user ID to log in.
- An integration client user identity must use the Costpoint Database authentication method. Additionally, the **Allow Application Access via Integration Service** check box must be selected.

Integration Client Authentication

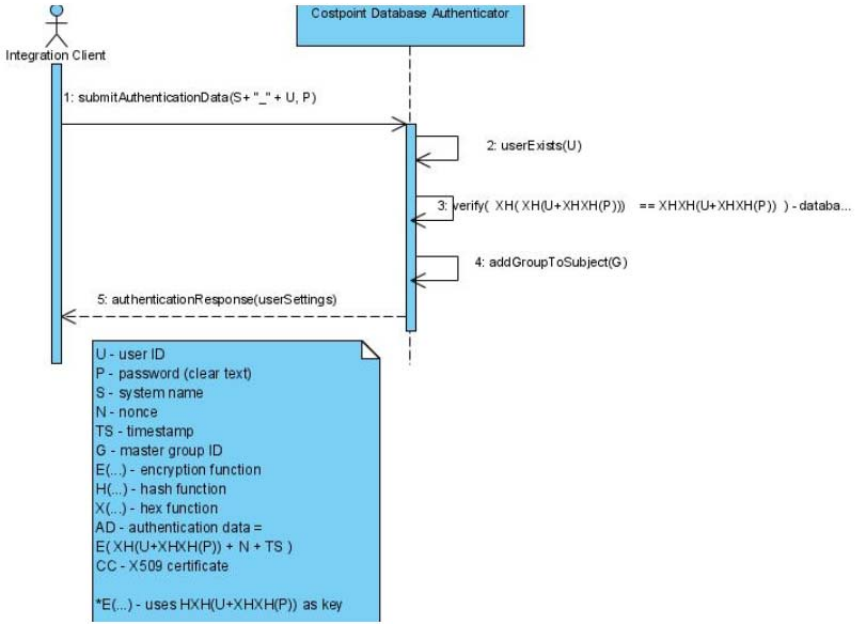
1. The client submits authentication data such as a user ID, password, and system name.

   Due to limitations in the Web services framework, the system name must be sent concatenated with the user ID (for example, O60QRD__SMITH).

   Use two underscore characters "__" as the delimiter between the system name and user ID.

2. The CPRDBMS Authenticator parses out the system name, user ID, and password and checks if the user exists in the database.
3. The CPRDBMS Authenticator verifies the user's password in the database.
4. If authentication succeeds, the CPRDBMS Authenticator creates a JAAS subject for the user and adds the master Costpoint Group "ApplicationUserGroup" to the created subject. If authentication fails, the CPRDBMS Authenticator generates an error.
5. The CPRDBMS Authenticator generates and sends either an "Ok" response or an "Error" response to the integration client.

   If authentication succeeds, the user gains access to Costpoint.



Set-Up Steps Required for Each Authentication Method

If you want to use the Costpoint Database authentication method, you do not need to perform any extra configuration steps. However, all other authentication methods require some special configuration.

Each of the configuration steps is described later in this guide.

Authentication Method	Configuration Steps Required
Costpoint Database	None; configuration provided by default
Active Directory	Windows Active Directory Setup
Single Sign-On	Single Sign-On Setup
Single Sign-on or Active Directory	Single Sign-On Setup + Windows Active Directory Setup
Windows Domain and Active Directory	Single Sign-On Setup + Windows Active Directory Setup
Windows Domain and Costpoint Database	Single Sign-On Setup
Client Certificate	Client Certificate Setup

Windows Active Directory Setup

If you are upgrading from Costpoint 6.1 and use the standard WebLogic Active Directory authentication provider, you have to remove it prior to using Active Directory authentication in Costpoint 7.0.

Log into the WebLogic console, and remove the Active Directory provider under the list of CPRealm providers.

To enable authentication of Costpoint users with Windows Active Directory, complete the following steps:

1. Configure the Windows Domain Controller and Active Directory.

   The Active Directory service is the distributed directory service that is included with the Microsoft® Windows Server 2000-2007 operating system. It enables centralized, secure management of an entire network. A domain controller is a server that is running a version of the Windows Server 2000-2007 operating system and has Active Directory installed.

   For more information on how to set up the Domain Controller and Active Directory, refer to Microsoft documentation.

2. Update the Windows Active Directory settings in the enterprise.properties file.

   The settings should be updated using the Costpoint Configuration Utility, which makes changes to the enterprise.properties file.

   For information about the Costpoint Configuration utility, see the *Deltek Costpoint 7.0 Configuration Utility Guide*.

   The following properties in the enteprise.properties files are updated:

   - **ldap.ldapServers** — Comma-separated list of logical names of LDAP/Windows Active Directory Servers. For example, if users will be authenticated against the esdtest.com domain, then the whole domain name or part of its name can be used as the logical name of the LDAP server (for example, **esdtest)**.
   - **ldap.<LDAP Server Name>.host** — Host name/IP address of the Windows Active Directory domain controller (for example, **esdtest** or **10.4.34.2**)
   - **ldap.<LDAP Server Name>.domain** — Domain name inside the Windows Active Directory domain controller (for example, **esdtest.com**)
   - **ldap.<LDAP Server Name>.port** — Port number of the Windows Active Directory Server (for example, **389**)
   - **ldap.<LDAP Server Name>.useSSL** — Whether or not to use SSL between the WebLogic Server and the Windows Active Directory Server (**true** or **false;** the default is **false**)

   If multiple LDAP servers/domains exist, a user will be authenticated against each server/domain until authentication succeeds. The order of LDAP Servers should be specified in the **ldap.ldapServers** property, which should contain a comma separated list of LDAP servers/domains (for example, **ldap.ldapServers**=esdtest,esdtest2).

This is what the enterprise.properties file should look like after your edits:

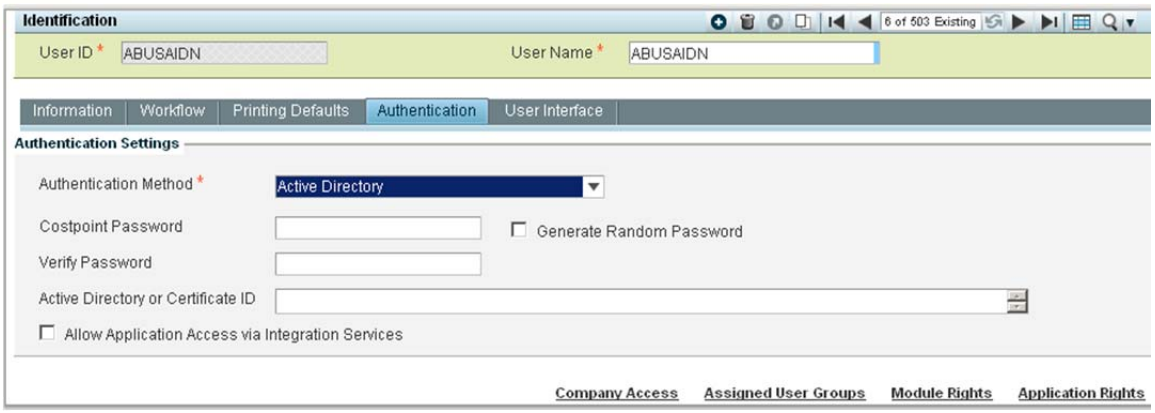
```
# ldap configuration example – single server/domain
ldap.ldapServers=esdtest
ldap.esdtest.host=10.4.34.4
ldap.esdtest.domain=esdtest.com
ldap.esdtest.port=389
ldap.esdtest.useSSL=false

# ldap configuration example – multiple servers/domains
ldap.ldapServers=esdtest,esdtest2

ldap.esdtest.host=10.4.34.4
ldap.esdtest.domain=esdtest.com
ldap.esdtest.port=389
ldap.esdtest.useSSL=false

ldap.esdtest2.host=10.4.34.6
ldap.esdtest2.domain=esdtest2.com
ldap.esdtest2.port=389
ldap.esdtest2.useSSL=false
```

Update User Setup

The Costpoint Administrator must also assign the Active Directory authentication method to each user who will use it. Use the Manage Users (SYMUSR) application to make this assignment.

To assign the Active Directory authentication method to a user, complete the following steps:

1. Click **Administration » Security » System Security » Manage Users**.
2. Select a user.
3. Click the Authentication tab.
4. Enter the following Authentication Settings:
   - **Authentication Method** — Enter **Active Directory**.
   - **Active Directory or Certificate ID** — Enter the Active Directory user ID.
5. Save your changes.
6. Repeat these steps for any users who should have Active Directory authentication.


Identification
6 of 503 Existing
User ID *
ABUSAIDN
User Name *
ABUSAIDN
Information
Workflow
Printing Defaults
Authentication
User Interface
Authentication Settings
Authentication Method *
Active Directory
Costpoint Password
Generate Random Password
Verify Password
Active Directory or Certificate ID
Allow Application Access via Integration Services
Company Access
Assigned User Groups
Module Rights
Application Rights

Single Sign-On Setup

To use Single Sign-On with the Windows 2008 R2 platform, the following requirements must be in place:

- AES-128, AES-256, and RC4 supported Kerberos encryption for WebLogic Server Active Directory user account
- Internet Explorer 8.0 and later for client workstations

Setting up Single Sign-On is a three-step process:

1. Configure the Active Directory Server.
2. Configure the WebLogic Server.
3. Configure IE browsers to work with this configuration.

In the sample procedure below, the following names are used:

- WebLogic Server Active Directory user account — sso_weblogic
- Active Directory Domain Controller host name — dc
- Active Directory Domain name — esdtest1.com
- WebLogic Server host name — serv2

Configure the Active Directory Server

The first step to implementing Single Sign-On with Windows Authentication is to configure the active directory server, which has two steps:

- Create a new user account in Active Directory.
- Use **setspn** to create the Service Principal Name (SPN).

Create a New User Account in Active Directory

To create a new user account in Active Directory for the host computer on which WebLogic Server runs, complete the following steps:

1. Start the Active Directory Users and Computers program on the Active Directory server.
2. Click **New User**.
3. Name the new user account.
4. Under **Account Options**, select the **This account supports Kerberos AES 128 bit encryption** option.

Enabling AES encryption can corrupt the user's password. Reset the password after this step.

5. Under **Account Options**, clear the **Do not require Kerberos preauthentication** option.

Use setspn to Create the Service Principal Name (SPN)

Use the **setspn** utility to create the Service Principal Name (SPN) for the user account created in the previous steps. You use SPNs to locate a target principal name for running a service. **setspn** allows you to view the current SPNs, reset the account's default SPNs, and add or delete supplemental SPNs. Some services and applications may require manual modification of a service account's SPN information to authenticate correctly.

For more information about setspn.exe, refer to the following Webpage:

http://technet.microsoft.com/en-us/library/cc731241(v=ws.10).aspx

To use the setspn utility to create the Service Principal Name (SPN) for the user account, complete the following steps:

1. Locate and execute the **setspn** utility in the Windows 2008 Resource Kit.
2. Enter the following commands at the DOS prompt:

   ```
   setspn -a HTTP/serv2.ESDTEST1.COM sso_weblogic
   ```

   Where

   serv2 is the server machine name running WebLogic.

   The domain name (ESDTEST1.COM) in the WebLogic Server host name must be UPPERCASE.

   The following output displays:

   ```
   Registering ServicePrincipalNames for
   CN=sso_weblogic,CN=Users,DC=esdtest1,DC=com
   HTTP/serv2.ESDTEST1.COM
   Updated object
   ```

3. Use the following command to identify the SPNs associated with your user account:

   ```
   setspn -L sso_weblogic
   ```

   The following output displays:

   ```
   Registered ServicePrincipalNames for
   CN=sso_weblogic,CN=Users,DC=esdtest1,DC=com
   HTTP/serv2.ESDTEST1.COM
   ```

   This step is critical. If the same service is linked to a different account in the Active Directory server, the client does not send a Kerberos ticket to the server.

Configure the WebLogic Server

The second step to implementing Single Sign-On with Windows Authentication is to configure the WebLogic server, which includes the following steps:

- Create the Keytab file with **ktab.**
- Process the Keytab file with **kinit.**
- Configure WebLogic to use Negotiate Identity Asserter.
- Create a JAAS login file for WebLogic to connect with Kerberos.
- Modify the WebLogic startup command.
- Update the user setup in Costpoint to use SSO.

Create the Keytab file with ktab.exe

Prior to creating the keytab file, complete the following steps:

1. Create a file named **krb5.ini** in your C:\Windows directory. (Use the WordPad text editor to create the file.)

   The Windows environment variable PATH must contain the folder C:\WINDOWS.

2. Add the following parameters to the **krb5.ini** file:

```
[libdefaults]
default_realm = ESDTEST1.COM
default_tkt_enctypes = aes128-cts
default_tgs_enctypes = aes128-cts
ticket_lifetime = 600

[realms]

ESDTEST1.COM = {
kdc = dc
admin_server = dc
default_domain = ESDTEST1.COM
}

[appdefaults]
autologin = true
forward = true
forwardable = true
encrypt = true
```

   In some cases, Active Directory is configured to use RC4 encryption as the default encryption algorithm. In those situations, both **default_tkt_enctypes** and **default_tgs_enctypes** should be set to **rc4-hmac**.

To create the keytab file, complete the following steps:

1. At the DOS command prompt, enter the following command to process your keytab file:

```
C:\Oracle\Middleware12.1.1\jdk1.7.0_02\bin\ktab -k
C:\deltek\costpoint\70\delteke\cp_keytab -a sso_weblogic@ESDTEST1.COM -n
0
```

The domain name (ESDTEST1.COM) in the username must be UPPERCASE.

For the last option **-n 0, 0** stands for digit 0.

You create the **cp_keytab** file in your **delteke** domain folder (for example, **C:\deltek\dostpoint\70\delteke**).

2. You will be prompted for a password. Enter the password for the **sso_weblogic** user created in Active Directory earlier:

```
Password for sso_weblogic@ESDTEST1.COM:Password1
```

3. You should see the following output:

```
Done!
Service key for sso_weblogic@ESDTEST1.COM is saved in
C:\deltek\costpoint\70\delteke\cp_keytab
```

Process the keytab File with kinit.exe

After creating the keytab file, you need to run the **kinit** utility. You use this utility to obtain and cache Kerberos ticket-granting tickets.

To process the keytab file with kinit, complete the following steps:

1. Run the **kinit.exe** utility to process your keytab file:

```
C:\Oracle\Middleware12.1.1\jdk1.7.0_02\bin\kinit -k -t
C:\deltek\costpoint\70\delteke\cp_keytab sso_weblogic@ESDTEST1.COM
```

The domain name (ESDTEST1.COM) must be UPPERCASE.

The following output displays:

```
New ticket is stored in cache file
C:\Users\Administrator.ESDTEST1\krb5cc_administrator
```

2. Run the **klist** program to verify that the **kinit** program succeeded. **klist** displays the entries in the local credentials cache and key table. After you modify the credentials cache with **kinit** or modify the **cp_keytab** with **ktab**, the only way to verify the changes is to view the contents of the credentials cache and/or **cp_keytab** using **klist**. This program does not change the Kerberos database.

   Use this command to run the **klist** program:

```
C:\Oracle\Middleware12.1.1\jdk1.7.0_02\bin\klist
```

   The following output displays:

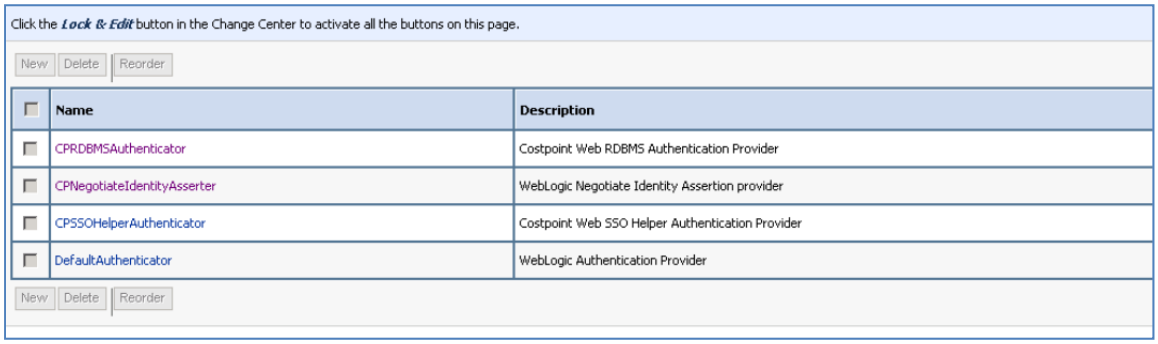
```
Credentials cache:
C:\Users\Administrator.ESDTEST1\krb5cc_administrator


Default principal: sso_weblogic@ESDTEST1.COM, 1 entry found.
```

```
[1]  Service Principal:  krbtgt/ESDTEST1.COM@ESDTEST1.COM
     Valid starting:  Feb 09, 2012 13:28
     Expires:         Feb 09, 2012 23:28
```

Configure WebLogic to Use Negotiate Identity Asserter

To configure WebLogic to use Negotiate Identity Asserter, complete the following steps:

1. Log into the WebLogic console.
2. In the top left corner of the screen, click **Lock & Edit**.
3. In the Domain Structure, click **Security realms**.
4. Click **CPRealm**.
5. In the **Settings for CPRealm** window, click the Providers tab. The Authentication tab displays.
6. Click **New** and enter any name (for example, CPNegotiateIdentityAsserter).
7. From the list of available authentication providers, select **NegotiateIdentityAsserter**.
8. Click **OK**.
9. Click the **Reorder** button, and move **CPNegotiateIdentityAsserter** up so that it becomes second in the list of providers (just after **CPRDBMSAuthenticator**, but before **CPSSOHelperAuthenticator**).



Create a JAAS Login File for WebLogic to Connect with Kerberos

The JAAS login file tells the WebLogic security framework to use Kerberos authentication and defines the location of the keytab file that contains the Kerberos identification information for the WebLogic server.

To create the JAAS login file, complete the following steps:

1. Use the WordPad text editor to create a JAAS login file with the name **krb5login.conf**.
2. Add the following contents to the file:

```
com.sun.security.jgss.krb5.initiate {
 com.sun.security.auth.module.Krb5LoginModule required
 principal="sso_weblogic@ESDTEST1.COM" useKeyTab=true
 keyTab="C:\\deltek\\costpoint\\70\\delteke\\cp_keytab" storeKey=true;
};
com.sun.security.jgss.krb5.accept {
 com.sun.security.auth.module.Krb5LoginModule required
 principal="sso_weblogic@ESDTEST1.COM" useKeyTab=true
 keyTab="C:\\deltek\\costpoint\\70\\delteke\\cp_keytab" storeKey=true;
};
```

3. Copy the file into your **delteke** domain folder (for example, C:\deltek\costpoint\70\delteke).

The files **krb5login.conf** and **cp_keytab** must be in the **delteke** domain folder (for example, **C:\deltek\costpoint\70\delteke**).

Modify the WebLogic Server Command files

Edit the C:\Deltek\Costpoint\70\Bin\CPWebSetEnv.cmd file using a text editor like Wordpad. Add a KERB_ARGS variable and set value to it. Append the new KERB_ARGS value to the already existing BEA_ARGS variable as shown below.

Remember to substitute the correct values for the domain name and file paths as required.

```
set KERB_ARGS=-Djava.security.krb5.realm=ESDTEST1.COM
-Djava.security.krb5.kdc=ESDTEST1.COM
-Djava.security.auth.login.config="C:\\deltek\\costpoint\\70\\delteke\\krb5login.conf"
-Djavax.security.auth.useSubjectCredsOnly=false
-Dweblogic.security.enableNegotiate=true -Dsun.security.krb5.debug=false



Set BEA_ARGS=-Dbea.home=%BEAHOME% -Dweblogic.RootDirectory=%CP_DOMAIN% -
Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE% -
Dweblogic.system.StoreBootIdentity=true %KERB_ARGS%
```

Costpoint Windows Service has to be reinstalled for these changes to be effective. It can be done by running InstallCPWebasService.cmd.

Update User Setup in Costpoint to Use Single Sign-On

To update the user setup in Costpoint to use Single Sign-On, complete the following steps:

1. Log into Costpoint as the system administrator (CPSUPERUSER).
2. Click **Administration » Security » System Security » Manage Users**.
3. Select a user who should be assigned the Single Sign-On authentication method.
4. Click the Authentication tab.
5. In the **Authentication Method** field, select **Single Sign-On**.
6. In the **Active Directory or Certificate ID** field, enter the Active Directory user ID.
7. Save your changes.

8. Repeat these steps for other users who should be assigned the Single Sign-On authentication method.

Configure Internet Explorer to Work with Single-Sign On

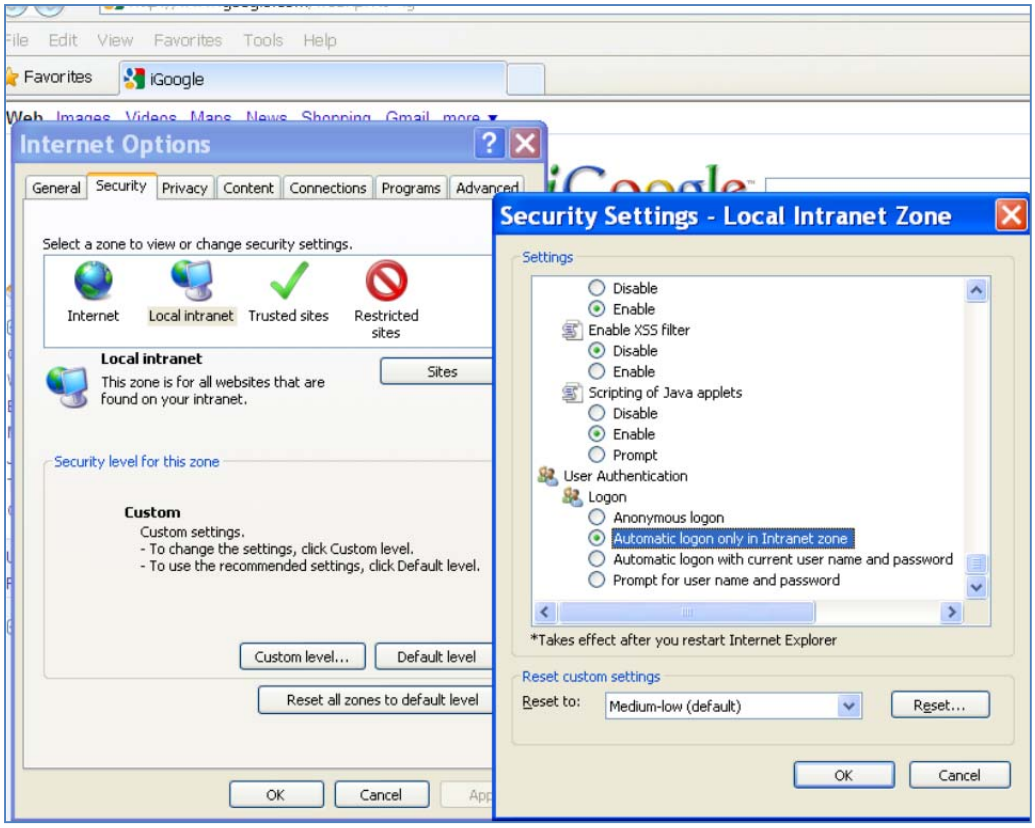
To enable clients to use Single Sign-On with Internet Explorer browsers, complete the following steps:

1. Add the Costpoint URL to your Local Intranet zone sites. From Internet Explorer, click **Tools » Security » Local intranet » Sites » Advanced** and add the Costpoint URL to the list of local intranet sites.

   Use only the fully qualified WebLogic machine name in the Costpoint URL.

   This should be the name that was configured on the Active Directory machine using **setspn** (for example, **http://serv2.esdtest1.com:7001**).

2. On the Security tab, click the **Custom Level** button. Under **User Authentication/Logon**, verify that the **Automatic Logon Only in Intranet zone** option is selected.



3. Click **Tools » Internet Options » Advanced**, and ensure that the **Enable Integrated Windows Authentication** option is selected.

Single Sign-On Troubleshooting

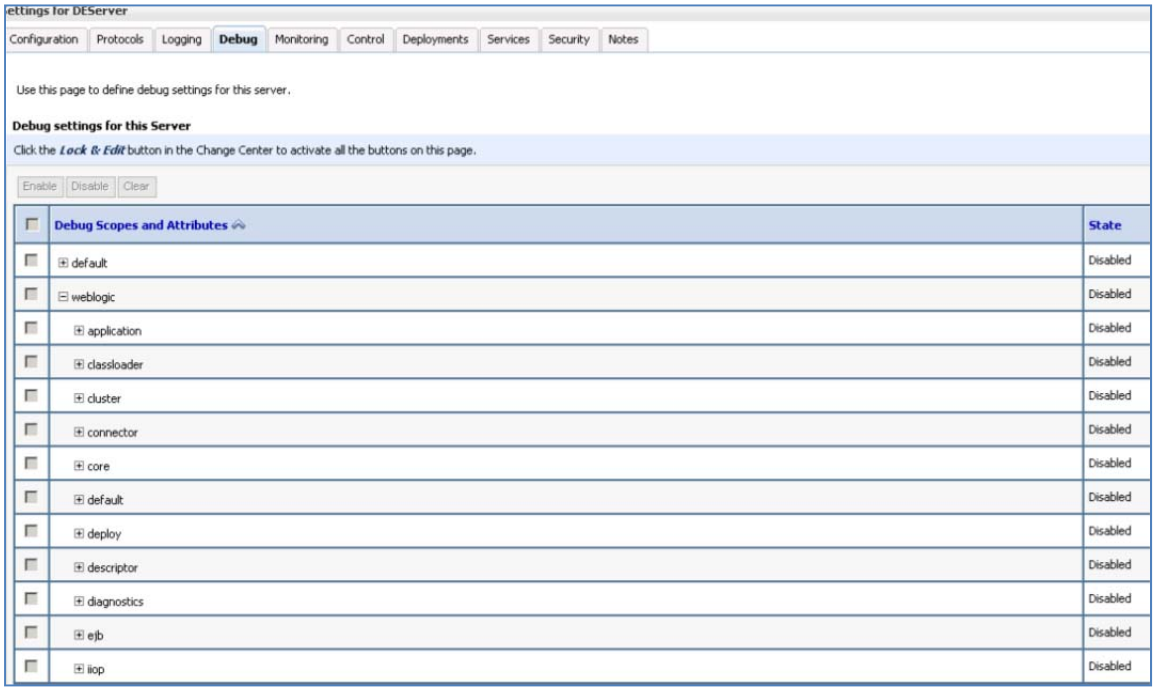
Deltek recommends that you take the following steps to diagnose problems with Single Sign-On authentication:

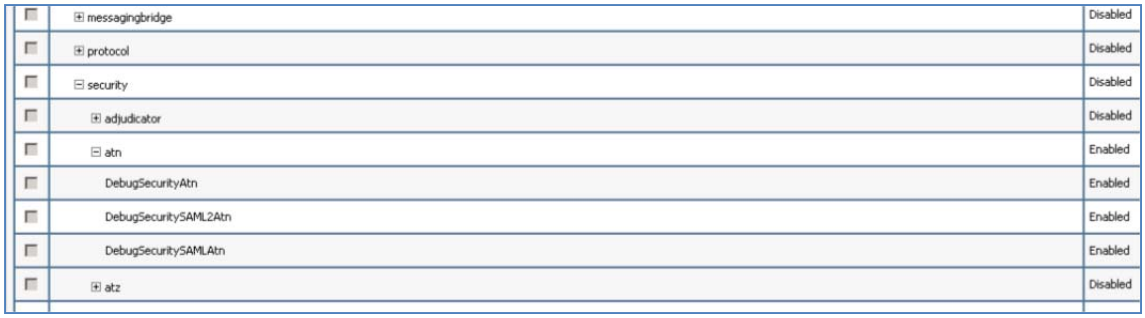
- Turn on verbose debug logging during authentication attempts.
- Use an additional run-time command line switch.
- Stop using Windows Service and use the StartCPWeb.cmd file to start the server.

Turn on Verbose Debug Logging

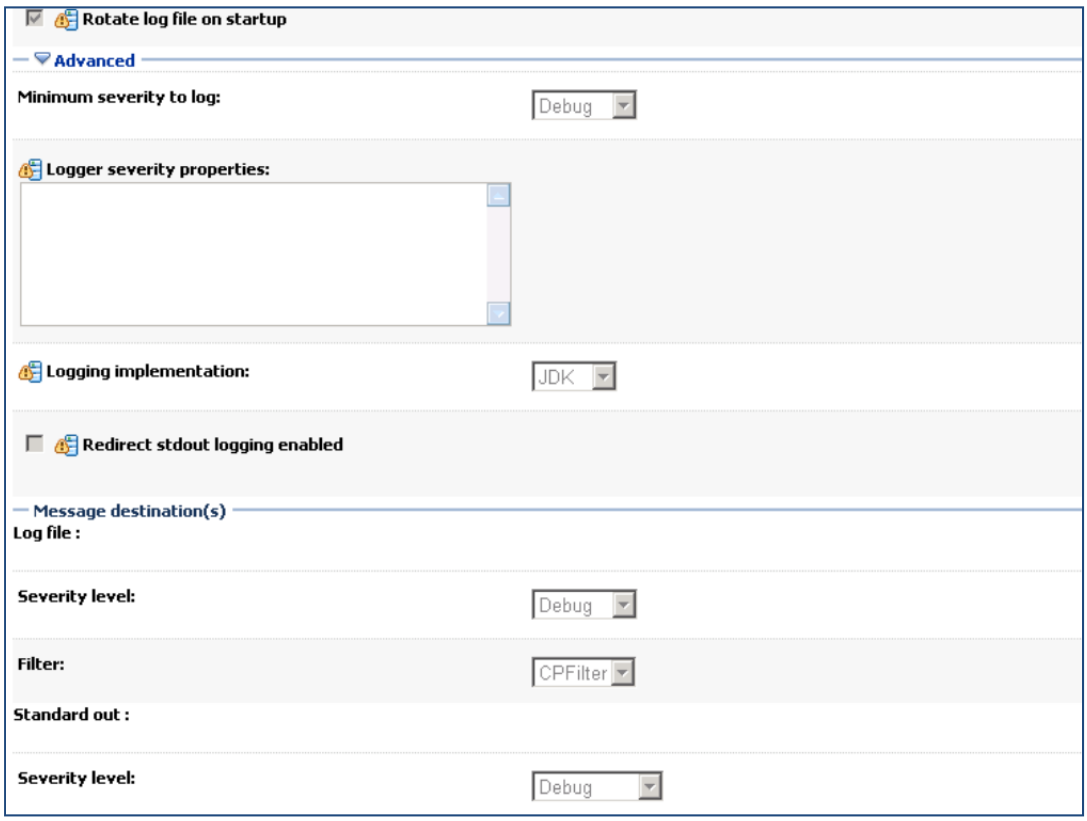
To turn on verbose logging, complete the following steps:

1. In the WebLogic console, click **Lock & Edit** in the top left corner.
2. In the Domain Structure on the left, click **Environment – Servers**.
3. On the Summary of Servers page, click **DEServer(admin)**.
4. Click the Debug tab.
5. Drill down to **weblogic – security – atn**.
6. Select the **DebugSecurityAtn** option.
7. Click **Enable** at the top or bottom of the page.





8. Click the Logging tab, and then click **Advanced**.
9. Change the **Logging Level** to **Debug** for:
   - **Minimum severity to log**
   - **Log file**
   - **Standard out**



10. Click **Activate changes** in the top-left corner. This will change the config.xml file. The code below is an example of the file after the changes have been activated.

```
<file-
name>C:/oracle/Middleware/wlserver_10.3/delteke/servers/DEServer/logs/DES
erver.log</file-name>
  <rotation-type>bySize</rotation-type>
  <number-of-files-limited>true</number-of-files-limited>
  <file-count>7</file-count>
  <file-min-size>500</file-min-size>
  <rotate-log-on-startup>true</rotate-log-on-startup>
  <logger-severity>Debug</logger-severity>
  <log-file-severity>Debug</log-file-severity>
  <stdout-severity>Debug</stdout-severity>
  <log-file-filter>CPFilter</log-file-filter>
  <stdout-filter xsi:nil="true" />
  <domain-log-broadcast-severity>Notice</domain-log-broadcast-severity>
  <domain-log-broadcast-filter xsi:nil="true" />
  <memory-buffer-severity>Debug</memory-buffer-severity>
  <memory-buffer-filter xsi:nil="true" />
  <log4j-logging-enabled>false</log4j-logging-enabled>
  <redirect-stdout-to-server-log-enabled>false</redirect-stdout-to-
server-log-enabled>
  <domain-log-broadcaster-buffer-size>1</domain-log-broadcaster-buffer-
size>
  </log>
  <stuck-thread-max-time>36000</stuck-thread-max-time>
  <listen-port>7009</listen-port>
- <web-server>
  <name>DEServer</name>
- <web-server-log>
  <logging-enabled>false</logging-enabled>
  <elf-fields xsi:nil="true" />
  </web-server-log>
  <default-web-app-context-root>/DEWebApp</default-web-app-context-root>
  </web-server>
- <server-debug>
  <name>DEServer</name>
- <debug-scope>
  <name>weblogic.security.atn</name>
  <enabled>true</enabled>
  </debug-scope>
  <debug-security-atn>true</debug-security-atn>
  <debug-security-saml-atn>true</debug-security-saml-atn>
  <debug-security-saml2-atn>true</debug-security-saml2-atn>
  <web-module>true</web-module>
  </server-debug>
```

This high level of verbose logging can impose a performance penalty on the operation of your production system. After you solve your authentication problems, Deltek recommends that you restore the settings in these files to settings that allow for adequate server performance.

11. Make changes to the C:\Deltek\Costpoint\70\bin\CPWebSetEnv command file.
12. Locate the following command:

    ```
    -Djava.security.krb5.realm=ESDTEST.COM
    ```

13. Add the following command just before the command in step 2:

    ```
    -Dsun.security.krb5.debug=true
    ```

    ```
    Locate -Dsun.security.krb5.debug=false and change value to true.
    ```

    ```
    12. Stop Weblogic Server and restart using
    C:\Deltek\Costpoint\70\bin\StartCPWeb.cmd file.
    ```

Client Certificate Setup

Follow the steps below to enable Client Certificate authorization.

Two Setup Methods

You can use either of these approaches to Client Certificate authorization.

- **Browser » WebLogic Server**

  With this authentication method, all the communication between the browser and the WebLogic Server is over https protocol. Therefore, WebLogic Server must be configured to support two-way SSL.

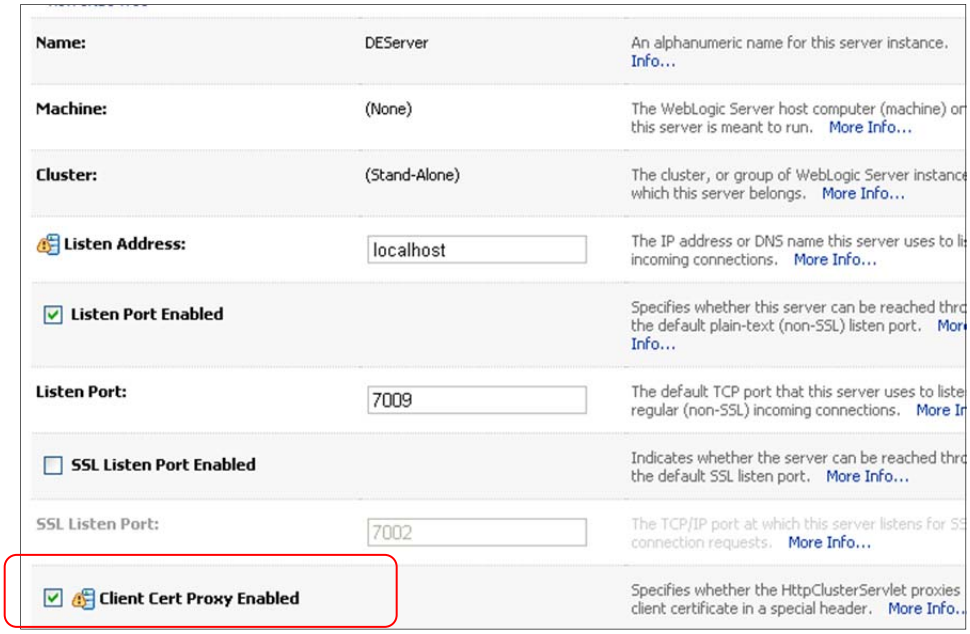
- **Browser » IIS » WebLogic Server (Cluster of WebLogic Servers)**

  With this configuration, all communication between the browser and IIS occurs over https protocol. The communication between IIS and the WebLogic Server can be implemented over https or just http. If the WebLogic Server (or WebLogic Server Cluster) and IIS sit inside the same local area network, Deltek recommends that you use http (not https) between IIS and WebLogic Server, because the SSL encryption and decryption routine creates unnecessary overhead between two (or more that two, if you use a WebLogic Server Cluster) trusted peers.

  In addition, you have to configure the IIS proxy to forward client certificates to the WebLogic Server (or the WebLogic Server Cluster).

  To configure the IIS proxy to forward client certificates, complete the following steps:

  1. Log into the WebLogic Server console, and navigate to **Environment » Server » General**.
  2. Select the **Client Cert Proxy Enabled** check box. If you are using a WebLogic Server Cluster, make the change for each server node.



Requirement for Valid Certificate

Both approaches to Client Certificate authentication require that the client (browser) has a valid certificate that can be trusted by WebLogic Server or IIS. The certificate must be installed on each user's machine. The next steps assume that the client has a valid certificate. This certificate must be imported into Internet Explorer.

Note that you do not have to set up any additional authentication providers to support Client Certificate authentication. The only providers required are those that come by default with the Costpoint installation (for example, the CPRDBMSAuthenticator, the CPSSOHelperAuthenticator, and the DefaultAuthenticator).

To import a certificate into Internet Explorer, complete the following steps:

1. Obtain a signed personal certificate from your organization's IT department, VeriSign, or another trusted certificate authority.
2. In Internet Explorer, click **Tools » Internet Options » Content tab » Certificates » Personal**.
3. Click **Import**, and use the Certificate Import wizard to import the certificate. When prompted, enter the password associated with the certificate.
4. Open the Costpoint Login page; for example:
   - https://dltkas44:7002/costpoint.htm (**Browser » WebLogic Server**)
   - https://dltkas88/cpweb/costpoint.htm (**Browser » IIS » WebLogic Server**)

   You should be able to log in to Costpoint without providing a User ID and password on the Costpoint Login page.

User Access to Modules, Applications, Reports, Etc.

Authorization controls access to resources by answering the following question: “Does a user have rights to access a protected resource?”

In Costpoint, we identify two types of resources that require protection:

- Application business objects
- J2EE server components/services

A security policy must be implemented for each component in the previous lists. A security policy answers the question: “Who has access to a resource?”

Resource Type	Components	Security Policies Defined By:
Application business objects	Modules, applications, result sets, actions, and reports	Costpoint security applications such as the following: Manage Users, Module Rights, Application Rights, Report Rights, Report Archive Rights, Action Rights, and Result Set Rights
J2EE server components/services	Web applications, EJBs, JDBC connection pools, JMS servers, Java connectors, and mail sessions	Server administration tools (for example, the WebLogic Server console)

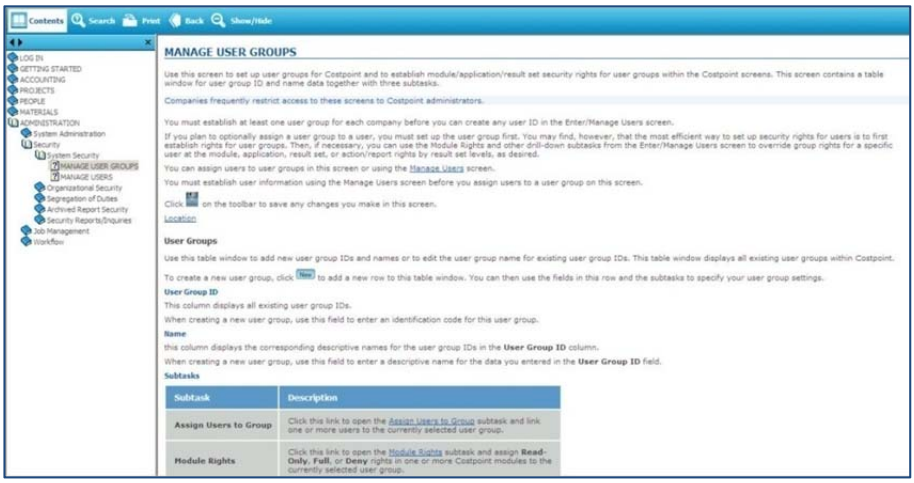
When Costpoint is installed, only one user account called **CPSUPERUSER** is created. This is a predefined administrative user in Costpoint that has full rights to all modules and applications. Deltek expects clients to log in to Costpoint under this account and set up additional user groups and users with appropriate privileges in the Manage Users and Manage User Groups applications. Keep in mind that those are regular Costpoint applications and you will need to provide rights for those applications to your Costpoint administrative users, who will be able to change other user’s privileges in Costpoint, create new Costpoint users and groups, or remove unneeded user accounts.

Also, for new installations and for upgrades from previous versions of Costpoint for which the **Apply Default User Groups and Permissions** option was selected, the install will add an out-of-the-box, predefined set of user groups and permissions. The idea is to give clients a template of what user groups they might want to have in the organization and what rights these user groups should typically have. For example, an **AP clerk** user group will be created, which has all the permissions AP clerks are expected to have. In total, the install will create 47 user groups that all start with a **STD**_ prefix (for example, STD_AP_MGR – Accounts Payable Manager, STD_CM_CLRK – Cash Management Clerk, and so on).

Assign Rights to Application Business Objects

You can control a user's access to application business objects using Costpoint screens and tables for entering and storing security information.

The Costpoint 7 online help provides detailed instructions for assigning rights to users. Look under **Administration » Security**.



Follow these guidelines:

In Costpoint 7, Organization security works the same way as it did in previous client/server versions of Costpoint, using the same screens and database tables.

- A user can be assigned to one or more user groups or to no user groups. In Costpoint 7, unlike previous versions of Costpoint.
- A user or user group can be given module-level security rights that control whether the user or group has **Full**, **Read-Only**, or **Deny** rights to a module in the Costpoint menu. If application security is not specified, module-level security also determines what, if any, access users have to applications within a module.
- A user or user group can be given application-level security rights that control whether the user or group has **Full**, **Read-Only**, or **Deny** rights to an application in the Costpoint menu. If a user has **Read-Only** rights for a maintenance application, he or she can only view result sets called from that application, regardless of what their result set rights may be. If a user's rights or any of their user group rights are set to **Deny**, the user will not be able to see that application in the Costpoint menu or access it directly.
- An application may be used in multiple modules. If application-level security is not specified for an application for a user and his or her user groups, the access rights of all the modules that contain that application are used to determine if that application can be accessed. If one or more of those module rights is set to **Deny**, the user does not have access to the application.
- Result set security is used to determine the specific activities a user can perform within a given result set. **No, Read, Update**, **Insert**, or **Delete** rights can be given for a maintenance result set. If a result set is used in more than one application, the result set security applies to all applications that call that result set. Result set security will not

override application security rights (or module rights if the application security rights are not defined).

- Action security specifies whether or not the user can execute actions for the result set. Rights for actions are either granted or denied. In general, unless **Action** rights are explicitly denied, the user may run actions associated with the result set.

If the result set by design is not Read-Only (in other words, one or more of **Insert**, **Update**, and **Delete** are available for that result set in the Design Tool), and the user has **Read-Only** rights for the result set in the W_RS_RIGHTS table, that user will not be able to execute any actions for that result set, unless they have been explicitly granted rights to those actions. Conversely, if the result set is set to **Read-Only** in the Design Tool, the default behavior is that the user can run actions for that result let unless action rights are explicitly denied.

- Report security specifies whether or not the user can run reports for the result set. Rights for reports are either granted or denied. By default, the user can run any report associated with the result set unless rights are explicitly denied.
- Report archive security specifies whether or not the user can view archived reports. Rights are set at different levels, based on a group of reports, a specific report, or a particular instance of a report. Rights can be set within one company or for all companies. Access to archived reports can be either granted or denied. You can also specify different levels of access, such as view reports, modify archive policy, and delete archived reports. In addition, organization security can be either ignored or taken into account when viewing archived reports. In the latter case, users with different organization security profiles can access the same archived reports.
- Security rights do not need to be explicitly specified in the database at each of the levels in order to fully view/access applications and result sets. If no application security is set up for a user and his or her user groups, module security can be used. If result set security is not defined for a user and his or her groups, application security determines what the user can do in that screen. If action and report rights are not specified, Costpoint allows the user to execute the actions or reports.
- Initially, Lookup result sets (result sets called from another result set using the **Lookup** button) are excluded from result set security because the Lookups do not allow users to modify data.

User and User Group Assignments

Because users can be assigned to multiple user groups and can have security rights of their own, the logic for determining what a user can access or modify is complex. To determine if a user has rights to access a module, application, or result set, data must be read from the user's own rights as well as the rights of all of the user groups to which the user belongs.

Module Security

The following rules are used to determine a user's module security rights:

- If there are no rows for a given module within the W_MODULE_RIGHTS table for a user or the user's assigned user groups, the user cannot view/access that module.
- If in one or more W_MODULE_RIGHTS rows for that user or his assigned user groups, the user is denied access to that module (ACCESS_FL = 9), the user cannot view/access that module.
- If one or more W_MODULE_RIGHTS rows exist for that module and none have the Deny setting (ACCESS_FL = 9) the user can view/access the module.

Application Security

The following rules are used to determine a user's application security rights:

- If there are no rows for that application within the W_APP_RIGHTS tables for the user or the user's assigned user groups, the application must determine security by checking the module rights for ALL modules that contain that application.
- If there are no rows for modules that contain the application, the user cannot access the application.
- If there are one or more module rows where access is denied (W_MODULE_RIGHTS. ACCESS_FL = 9) for the user or the groups the user belongs to, then the user cannot access the application.
- If there are one or more module rows selected for the user and the user's assigned user groups where access is denied (W_MODULE_RIGHTS. ACCESS_FL = 9).
- If one or more module rows selected for the user and the user's assigned user groups have **Full** access (W_MODULE_RIGHTS. ACCESS_FL = 5), the user can view and change data in the application.
- If one or more module rows exist, but all of the rows' access codes are set to **Read-Only** (W_MODULE_RIGHTS. ACCESS_FL = 1), the user can view the data in the application but cannot change it.
- If one or more W_APP_RIGHTS rows exist for the application and the user and the user's assigned user groups, use the following logic to determine the application rights:
  - If, in one or more W_APP_RIGHTS rows for the user and the user's assigned user groups, the user is denied access to the application (ACCESS_FL = 9), the user cannot view/access that application.
  - If, in one or more W_APP_RIGHTS rows for the user or the user's assigned user groups, the user is given **Full** rights to the application (ACCESS_FL= 5), for process applications, the user will be allowed to run processes that update the database.
  - If one or more W_APP_RIGHTS rows exist and they all have an access code of **Read-Only** (ACCESS_FL =1), the user can access and view the application, but not change data (even if the result set security would normally allow it). For process applications, the user can generate reports, but cannot perform processes that update the Costpoint database.

Result Set Security

The following rules are used to determine a user's result set security rights:

- If a user has **Full** access to an application, result set security is used to determine which result sets the user can view, add, change, or delete. If the user has **Read-Only** access to an application, result set security is used only to determine which result sets the user can view.
- If there are no rows for the result set within the W_RS_RIGHTS tables for the user or the user's assigned user groups, the application/module security determines the user's rights to result sets within a given application.
- If the user has **Full** rights to an application (or module if no application rights are defined), the user can select, insert, update, and delete rows within all result sets for that application.
- If, in one or more W_RS_RIGHTS rows for the user or the user's assigned user groups, the user is denied access to a result set (DENY_FL = Y), the user cannot view or update data in that result set.
- The user can view rows in the result set if one or more selected rows in the W_RS_RIGHTS table has the SELECT_FL = Y. The user can insert, update, and delete rows in that result set if one or more of the selected rows' INSERT_FL, UPDATE_FL, and DELETE_FL are set to **Y**, respectively (if they also have **Full** rights to that application).

Action Security

The following rules are used to determine a user's result set security rights:

- If a user has full access to a result set, result security is used to determine which actions the user can execute.
- If the result set is **Read-Only** by design (INSERT_FL, DELETE_FL, and UPDATE_FL are all **N** in S_RS_LIST), then, by default, the user can execute any action, regardless of data in the W_RS_RIGHTS table.
- If the result set is not **Read-Only** by design (one or more of INSERT_FL, DELETE_FL, UPDATE_FL are set to **Y** in S_RS_LIST) and the user has **Read-Only** access in W_RS_RIGHTS, the user will not be allowed to execute any actions on that result set unless rights are explicitly granted to him or her in W_ACTION_RIGHTS.
- In all other cases in which the user has rights to the result set, if the EXEC_FL is **N** in W_ACTION_RIGHTS for the action, the user cannot execute the action; if the EXEC_FL is **Y** in W_ACTION_RIGHTS or there are no rows in W_ACTION_RIGHTS for that result set, the user may execute the action.

Report Security

If the user has any access at all to the result set, he or she may run any report associated with that result set unless there is a row in W_REPORT_RIGHTS with EXEC_FL = 'N' for that report.

Hierarchy Diagrams

The diagrams below show the hierarchy of security settings for individual users and user groups.

Hierarchy of Security Settings for Users

Maintain Users (SYMUSR) [W_USER_UGRP_LIST, filter on TYPE = U]

- User Company Access [W_USER_COMPANY]
- Assign Groups to User [W_USER_GRP_USERS]
- Web Module Rights [W_MODULE_RIGHTS]
  - Web Application Rights by Module [W_APP_RIGHTS]
    - Result Set Rights by Application [W_RS_RIGHTS]
      - Action Rights by Result Set [W_ACTION_RIGHTS]
      - Report Rights by Result Set [W_RPT_RIGHTS]

Hierarchy of Security Settings for Users

Maintain User Groups (SYMGRP) [W_USER_UGRP_LIST, filter on TYPE = G]

- Assign Users to Group [W_USER_GRP_USERS]
- Web Module Rights [W_MODULE_RIGHTS]
  - Web Application Rights by Module [W_APP_RIGHTS]
  - Result Set Rights by Application [W_RS_RIGHTS]
    - Action Rights by Result Set [W_ACTION_RIGHTS]
    - Report Rights by Result Set [W_RPT_RIGHTS]

There are a few restrictions in the Application Security override of result set security:

- If a user has no access to an application, he or she cannot view any result sets from within that application, no matter what result set security access he or she has.
- If a user has **Read-Only** access to an application, he or she cannot modify data in any result set from within that application, even if the user has **Full** rights to the result set. However, the user may be able to view those result sets from other applications.

Implementing Security for J2EE Server Components and Services

The Costpoint application runs on a J2EE server (for example, a WebLogic Server) and uses the following J2EE components and services:

- Web application
- EJB

- Java connector
- JDBC service
- JMS service
- Mail service

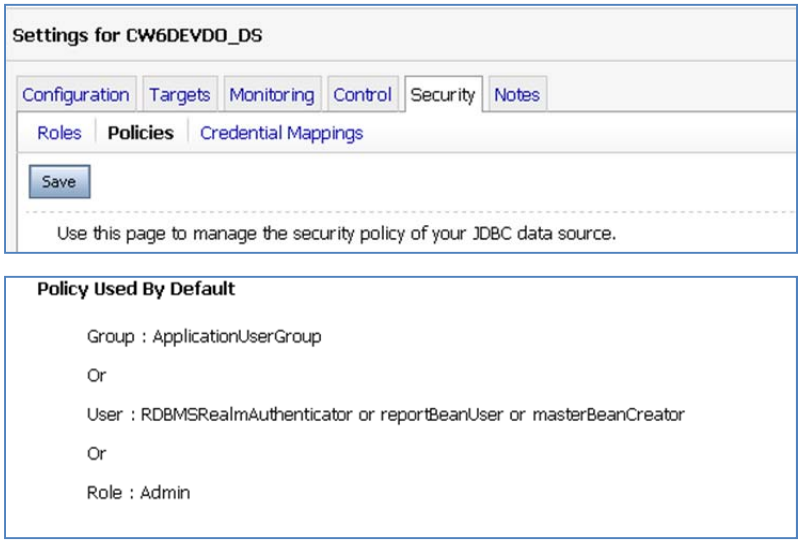
Each of these components and services must be protected; there are security polices implemented for each component. Implementation of these polices is vendor-specific.

WebLogic Server Implementation

Security policies for the WebLogic server are defined at the user level. Costpoint ships with some built-in users that support these security policies:

- **reportDataUser** — This user accesses the report bean during report generation.
- **reportBeanUser** — This user is used to run the report bean through the run-as property in the bean's Deployment Descriptor.
- **masterBeanCreator** — This user is used to create the master bean through the login bean.
- **asyncProcessUser** — This user is used for running processes and reports asynchronously or through the process server.
- **RDBMSRealmAuthenticator** — This user is used to access JDBC pools during the login process.

Security policies for these components and services are defined through the WebLogic console. For more details, log into the WebLogic console, select the targeted component or service, and go to the **Security/Policies** tab. For example, this is the security policy for a JDBC connection pool:



To achieve maximum security, Costpoint ships with WebLogic security policies pre-configured for built-in Costpoint users and user groups. Do not modify security policies to decrease the rights given to built-in users or user groups.

Deltek is the leading global provider of enterprise software and information solutions for professional services firms, government contractors, and government agencies. For decades, we have delivered actionable insight that empowers our customers to unlock their business potential. Over 14,000 organizations and 1.8 million users in approximately 80 countries around the world rely on Deltek to research and identify opportunities, win new business, optimize resource, streamline operations, and deliver more profitable projects. Deltek – Know more. Do more.®

deltek.com