Installing the Core Access Assurance Suite

Release 9.1

Core Security SDI Corporation

1000 Holcomb Woods Parkway
Suite 401
Roswell, GA 30076
Phone: (678) 304-4500
Fax: (770) 573-3743

Trademarks

Table of Contents

List of Figures

List of Tables

Chapter 1: Overview of the Core Access Assurance Suite

This chapter provides an overview of the Core Access Assurance Suite. It includes the following sections:

- *“Core Access Assurance Suite Platforms and Applications” on page 12*
- *“Single Server and Distributed Server Installation” on page 16*

Core Access Assurance Suite Platforms and Applications

The Core Access Assurance Suite includes two platforms which each support different applications:

- The provisioning platform, including the following applications:

  Core Provisioning® user provisioning solution

  Core Compliance™ policy verification solution

  RoleCourier® role management solution

  Core Password® password provisioning solution.

  Profile Management® profile management solution

- The classic platform, including the following applications:

  PasswordCourier Classic and PasswordCourier Support Staff Classic password provisioning solutions

  ProfileCourier Classic profile management solution

You can configure these applications to work in conjunction with one another or separately. They are installed simultaneously during a single installation process, described in this guide.

Core Provisioning

Core Provisioning (formerly AccountCourier) allows you to cost-effectively automate the creation and management of user accounts and access rights while ensuring compliance. Core Provisioning accelerates the provisioning of users and the management of user access to corporate resources in keeping with business policies. The administrator specifies the end users, the accounts available for management, and other functions by defining workflows using the Core Access Assurance Suite Administration Manager. Each workflow defines a specific configuration of Core Provisioning.

For each workflow, the administrator can

- Set up authentication criteria to select which end users can access the workflow.
- Delegate user identity management functions to the end user by enabling one or more of the following actions for a workflow:

  Add an account
  Change an account
  Create an account
  Enable an account
  Disable an account
  Delete an account
  View an account

- Set up ticketing that creates and updates tickets for selected events through integration with a Help Desk application.
- Set up notification that generates SMTP-based e-mail for selected events.

Core Compliance

Certify and manage employee access privileges, so you can identify, validate, and effectively enforce least-privileged access across the enterprise. Today's businesses have a lot of regulatory requirements that say only certain people should have access to certain information. With Core Compliance, your company can automate the process of gathering all information and setting provisions around access.

Core Compliance Certification Review Cycles

Core Compliance Access Certification Review Cycles provide worksheets where business users or IT resource owners perform certification. These worksheets are grids which users can sort and filter to display the data in the most efficient way.

Core Password

Core Password (formerly PasswordCourier) offers self-service password reset capabilities for the enterprise and enables your users to use a single password to access multiple systems.

Core Password enables end users to reset their own passwords on a wide range of systems and applications from a web browser or log in screen. You can configure Core Password to create and update trouble tickets for a password reset request; this ensures the tracking of service quality information and creation of audit trail of Help Desk and support activity.

Instead of providing a separate Help Desk/problem management system, Core Password integrates with leading Help Desk management systems.

Core Password automates the following procedures:

- Authenticating end users
- Creating an audit trail in the Help Desk system and/or via e-mail
- Recording end user password reset request status and service quality statistics
- Reporting security incidents: if someone enters incorrect authentication information, Core Password can create a security incident trouble ticket in the Help Desk system and send an e-mail.

Profile Management

Profile Management enables end users to create and update their profiles securely using a web browser. As with Core Password, Profile Management integrates with leading Help Desk management systems. Profile Management automates the following procedures:

- Authenticating end users
- Updating profiles
- Creating an audit trail in the Help Desk system and/or via e-mail
- Recording end user profile creation and updates

- Reporting security incidents: if someone enters incorrect authentication information, ProfileCourier can create a security incident trouble ticket in the Help Desk system and send an e-mail

Core Access

Core Access (formerly Access Request Manager) offers an intuitive Web-based interface that provides detailed information about requests that is easy for business managers and other approvers to use.The Access Request Manager solution is a component of the Access Assurance Suite that resides within the Access Assurance Portal. The Access Request Manager is a complete, highly functional access request management system that enables:

- An individual, whether in IT or in a line of business, to request access to resources, such as an online application system.
- Designated approvers to approve or reject access requests.

The Identity Mapping Solution

The Identity Mapping Solution is a component of the Access Assurance Suite that resides within the Access Assurance Portal. It provides a user-friendly interface for an administrator to establish an enterprise-wide Data Mapping process including the following:

- Identifying sources of account, profile, and entitlement data including HR systems or Active Directory
- Configuring data collection rules or data feeds that collect and filter account, profile, and entitlement data
- Creating identity mapping rules to match user accounts with the user profile information, either automatically or manually.
- Executing mapping rules on data feeds periodically using the scheduler, or immediately.
- Reviewing the information to take timely action to resolve any discrepancies.
- Providing auditing and reporting on the Identity Mapping process.

Data Security Utility

The Core Access Assurance Suite supports two methods to secure data in the profile management and authentication data source: hashing and encryption. The Data Security Utility is an Core Access Assurance Suite administrative tool that allows selection of the fields within a data source whose data is to be hashed or encrypted. Core Security products can use secured data fields for end user validation, authentication, profile management, and query operations.

Marking a field hashed or encrypted in the Data Security Utility does *not* hash or encrypt data stored in the selected data source. Marking a field only tells the Core Server to treat the value in that field as hashed or encrypted. The field values for all existing records remain in their current format—only values entered after a field is marked are stored as hashed or encrypted.

For this reason, the best practice for securing data is for an administrator to plan and implement hashing and/or encryption before users begin to add values into the database. Once a field is marked as secure, the Core Server hashes or encrypts end user input for that field and compares it to stored data.

Any records with existing cleartext data or records secured with a different algorithm do not match the secure value generated by the Core Server and the authentication or query fails.

Note: This means that all end users *must* update their profiles by entering a value for the marked field even if this field already exists and has a stored value. Furthermore, end users must enter a *new* value.

To begin using the Data Security Utility, refer to the manual *Using the Core Access Assurance Suite Administration Manager Utilities*.

Single Server and Distributed Server Installation

During the installation, you choose whether to install the Core Access Assurance Suite on a single server (a “Complete” installation) or to install different components of the suite on different servers in the network (a “Custom” installation). You make this selection on the Setup Type dialog (see page 33).

Components of the Core Access Assurance Suite

There are four main components that are installed as part of the Core Access Assurance Suite.

- **Connector Framework** — The Connector Framework (CF) web service provides connectivity to the various targets in your enterprise.
- **Connector Framework Manager** — The Connector Framework Manager (CFM) web service manages the target configuration of all supported connectors. This service also manages communication between the Core Server and one or more Connector Frameworks.
- **Core Server** — The Core Server allows you to configure workflows for Password Reset and Provisioning actions.
- **Publisher and Web Access Options** — The Publisher web service supports an interface that allows end users to communicate with the Core Server.

Note: These components must be started in a particular order due to the dependencies they have with each other.

- CourProfiler service should be started first, as it is responsible for logging for all other components.
- CourConfigSvr service should be started next, as it is responsible for reading/ writing configuration information from the configuration repositories.
- Each Connector Frameworks should be started subsequently. Minimally, there should be one Connector Framework, usually residing on the same server as the Connector Framework Manager.
- The Connector Framework Manager should be started after all the Connector Framework(s) are started. The CFM discovers all the Connector Frameworks that it is configured to manage.
- The CourionService should be started after the Connector Framework Manager finishes initializing.
- Once the above are started in the presented order, the following do not require a specific order for starting.
  - CourATLService services the provisioning platform and classic platform end user ASP pages. IIS and CourionService should be running. This service starts automatically, if it is needed and is not running. You may need to disable the service during maintenance windows to prevent it from auto-starting.
  - CourATLAdmin services the provisioning platform’s Administration Manager. IIS and CourionService should be running. This service starts automatically, if it is needed and is not running. You may need to disable the service during

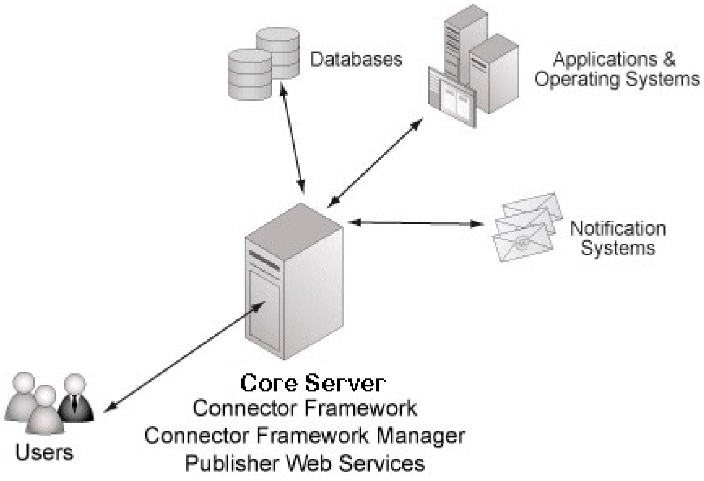
maintenance windows to prevent it from auto-starting. For security precautions you might want to disable this and only enable this service when needing to access the Administration Manager.

- All the services that start with “Courion ” are part of the Access Assurance Portal and are used for the portal authentication, Access Request Manager, Access Certification, Identity Mapping, etc. They can be started in order.
- CourGATService and CourTSAOService are part of the PasswordCourier Transparent Synchronization access option and can be started after the CourionService is started.
- CourProtocolService services the Enable Users Utility and the Configuration Migration Utility.
- CourPublisher service can be set to disabled. It is not used for anything at this time.

Single Server Installation

Figure 1 shows the components of the Core Access Assurance Suite installed onto a single server.

Figure 1: Single Server Installation



Distributed Installation

With a distributed installation, you can install the individual components on different servers. Installation is flexible. You can choose to separate out only one of the components, grouping the others onto a single server, break out all components into different servers, or other combinations. Additionally, you can install multiple instances of the Connector Framework and Connector Framework Manager on multiple servers. (Only one instance of a specific component may be installed on a server).

Note: Distributed installation is only available for the provisioning platform. If you are using the classic platform, install the Core Access Assurance Suite on a single server.

The following are examples of distributed installation in a corporate network:

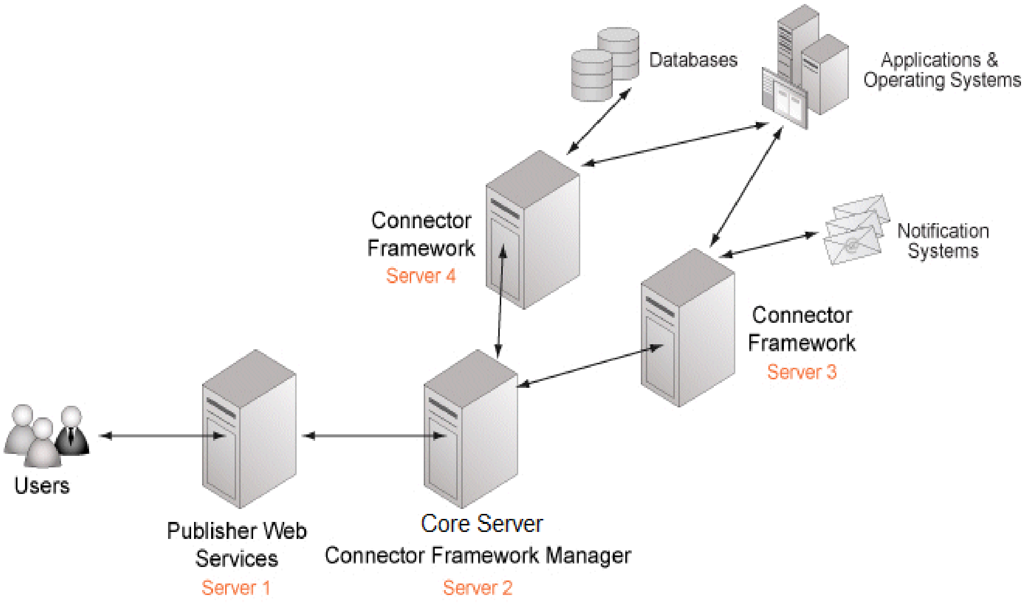
- **Multiple Domains** — If your company has multiple domains, and you install the Core Server and the Connector Framework Manager in Domain A, you may need to access targets in Domain B. If you install a Connector Framework on Domain B, the administrator who configures the connectors in Domain B only needs privileges in Domain A, because connectors are configured on the server in which the Connector Framework Manager is installed. In single server installation, the administrator would need privileges on both domains.
- **Supporting Client Software** — Some connectors (IBM Lotus Notes, Microsoft Exchange, Oracle E-Business, *etc.*) require supporting client software. By installing the Connector Framework on a server on which those tools are already installed, you avoid having to install those tools on the Core Server.

  Note: The supporting software also needs to be installed on the server hosting the Connector Framework Manager, so this example only applies when the Core Server and CFM are installed on different machines.
- **Load Balancing** — If you have a target that receives heavy usage, you can install Connector Frameworks on two or more servers and configure them for the same target. The Connector Framework Manager contacts them in a round-robin manner, to distribute the load.

  Not all targets can be configured for load balancing. Some targets can only communicate with a single Connector Framework. For more information, see *“Target Assignment” on page 86*.
- **High Availability** — You can install the Connector Framework Manager on two or more servers. If the primary CFM becomes unreachable, or fails in any manner, the Core Server begins communication with the next available CFM in the list. The Core Server then keeps communicating with that CFM until it becomes unreachable, at which point it tries the next CFM in the list. If there are no further CFMs in the list, it tries the primary CFM again. When the CourionService is started or restarted, the Core Server starts at the top of the CFM list, attempting communication with the primary CFM.
- **Securing the Core Server** — The Publisher and Web Access option includes all services used by both end-users and administrators for contact with the Core Server. By installing this option on a separate server, you can put the Core Server in a very secure segment of your network, while the Publisher is placed in a more public part of the network.
- **Improving End User Access Performance** — If your business is spread among different facilities, you can improve performance for the end user by placing the Publisher on a server where end users are located. This provides quicker loading of web pages since only transaction data is sent between the Core Server and the Publisher.

Figure 2 shows an example of a distributed installation.

Figure 2: Distributed Server Installation



In this example, the Publisher Service is installed on server one, the Core Server and Connector Framework Manager are installed on server two, and two separate Connector Frameworks are installed on servers three and four. Each Connector Framework communicates with a specific target (a notification system for server 3 and a database for server 4) and they also both share communication with a third target.

Order of Installation and Configuration for a Distributed Installation

When you install the different components on separate servers, you must enter the server name and other information for specific components when configuring other components. For this reason, you should install and configure components in the following order:

1. Connector Framework
2. Connector Framework Manager
3. Core Server
4. Publisher Manager

When you configure the Connector Framework Manager, you assign each target to one or more Connector Frameworks. When you configure two or more CFs, you designate one as the default CF. If you add a target in the Connector Configuration Manager after configuring the CFM, that target is automatically assigned to the default CF. If you want to access that target from a different or additional CF, you need to re-configure the CFM to change the target assignment.

Best Practice Examples for a Distributed Installation

If you are installing a single instance of the Connector Framework Manager (CFM), Core Security recommends that you install it on the same machine as the Core Server. Since all connector configuration is done on the machine where the CFM is installed, keeping the CFM on the same machine as the Core Server allows for easier configuration and maintenance.

However, if you are accessing connectors that require you to install additional client software, you may want to install the CFM on a different server. This is because you must install the client software on both the CFM and all CFs that access that target. This allows you to avoid having to install the client software on the machine hosting the Core Server.

The main reason for installing more than one CFM is to provide high availability. In case the primary CFM becomes unreachable or fails in any manner, the Core Server begins communicating with the secondary CFM. For this reason, if you install more than one instance of the CFM, those instances should be installed on separate machines from the Core Server.

Standard and Distributable Connectors

There are two categories of connectors:

- **Standard Connector** — You can assign a target associated with a standard connector to one Connector Framework.
- **Distributable Connector** — You can assign a target associated with a distributable connector to multiple Connector Frameworks.

Moving a Standard Connector to the List of Distributable Connectors

Certain connectors are distributable by default when you install the Core Access Assurance Suite. The Target Assignment list, configured in the Web Service Configuration Manager, indicates whether a connector is distributable with an icon:

- The icon indicates a standard connector
- The icon indicates a distributable connector.

You can move a standard connector to the list of distributable connectors if your installation requires this.

Note: Standard connectors that are moved to the distributable list and run in a distributed environment may not function correctly in all distributed environments. Please contact Core Security prior to using a standard connector in a distributed configuration.

To move a standard connector to the distributable list, edit the following XML file:

cntr_target_constraints_override.xml

It is located in: Program Files\Core Security\CourionService\conf

Follow the instructions in the XML file to move the connector to the distributable list. When you do, the connector appears as distributable (with a icon) in the Target Assignment list. See *“Target Assignment” on page 86* for more information on the Target Assignment list.

Chapter 2: Installing the Core Access Assurance Suite

This chapter describes how to install the Core Access Assurance Suite. It includes the following sections:

Before Installing the Suite

Before installing the product, confirm that all necessary conditions have been met as described in *Product Requirements for the Core Access Assurance Suite*.

Core Security recommends exiting all running applications before installing the Core Access Assurance Suite.

Installation requires one or more access keys, obtained from Core Security.

Installing Microsoft Message Queuing and Microsoft Internet Information Service and Microsoft .NET Framework 4.7.1

The Core Access Assurance Suite requires installation of the following Windows system components:

- Microsoft Message Queuing

  Note: If you plan to replicate the server used to host the Core Access Assurance Suite in your environment using virtualization, after you replicate the server, install Microsoft Message Queuing, then install the Core Access Assurance Suite.

  If you have already installed Microsoft Message Queuing and subsequently replicated the servers, contact Core Security customer support for more information. See *“Problem Reports” on page 135* for instructions on how to do this.

- Microsoft Internet Information Service
- Microsoft .NET Framework 4.7.1 - Download and install Microsoft .NET Framework 4.7.1 from this location: https://www.microsoft.com/en-us/download/details.aspx?id=56116

If these system components are not already installed when you start the Core Access Assurance Suite installation, a warning dialog box appears and the installation stops. Refer to Microsoft documentation for information about how to install these components.

Java Runtime Environment

To use the Customization Manager for PasswordCourier Classic, PasswordCourier Support Staff Classic, and ProfileCourier Classic, you need to install a current version of the Java Runtime Environment on the administrators’ client machines that run these applications and the utility. You can download it from the following location:

www.java.com

.NET Framework 3.5

Before you install the Core Access Assurance Suite on Windows Server 2012, follow these steps to download and install the version of .NET Framework that is required:

1. Download Microsoft .NET Framework 3.5 from this location:

   http://www.microsoft.com/en-us/download/details.aspx?id=21

   Place it in any location. Example: C:\dotnet35

Note: If you have the .NET 3.5 ISO, mount the ISO and run the following command to extract the source files to any location. In the following example, c:\dotnet35 is the location.

xcopy d:\sources\sxs\*.* c:\dotnet35 /s

2. Open Server Manager and install the following Server Role if it is not already installed:

   a. Web server (IIS)

   b. Under Web server (IIS)\Web Server\Application development

      net extensibility 3.5

      Application Initialization

      ASP

      ASP.NET 3.5

      ISAPI Extensions

      ISAPI Filters

3. Continuing in Server Manager, install the following Server Features:

   a. .NET Framework 3.5 Features

      .NET Framework 3.5(includes .NET 2.0 and 3.0)

      HTTP Activation

      Non-HTTP Activation

   b. Message Queuing

   c. Windows Identity Foundation(3.5)

4. Launch Microsoft Powershell as an Administrator. Run the following commands in PowerShell after navigating to the location of the .NET 3.5 setup file from step 1:

   a. $IISFeatures = @("Web-Static-Content", "Web-Default-Doc", "Web-Http-Errors", "Web-Asp-Net", "Web-Asp-Net45", "Web-Net-Ext", "Web-ISAPI-Ext", "Web-ISAPI-Filter", "Web-Http-Logging", "Web-Log-Libraries", "Web-Request-Monitor", "Web-Http-Tracing", "Web-Windows-Auth", "Web-Filtering", "Web-IP-Security", "Web-Stat-Compression", "Web-Dyn-Compression", "Web-Mgmt-Console", "Web-Scripting-Tools", "Web-Metabase", "Web-WMI", "Web-Lgcy-Scripting","NET-Framework-Core")

   a. Add-WindowsFeature -Name $IISfeatures -logPath "$Env:ComputerName.log" –Source C:\dotnet35

      Note: The source for this step is the location of the .NET framework from Step 1.

5. Install the Core Access Assurance Suite. After the installation completes, restart the server.

6. In IIS, change the Default Application Pool to v2.0:

   a. Open Internet Information Services (IIS) Manager.

   b. On the Connections pane, expand the server node and click Application Pools.

   c. On the Application Pools page, select the “DefaultAppPool”, and then click Basic Settings in the Actions pane.

d. In the Edit Application Pool dialog box, in the .NET Framework version list, select the version ".NET Framework v.2.0.50727".

e. Click OK.

7. Restart IIS.

Installing the Loopback Server

If you are licensed for the Access Request Manager and the Identity Mapping Solution, you must follow these steps to create a linked server with the name Loopback.

1. Using Microsoft® SQL Management Studio, open the *DBServer.sql* file (which is located in the Courion installation folder).
2. Select SQL CMD Mode under **QUERY** in Microsoft® SQL Management Studio.
3. Configure the variables explained in Table 1. The variables, which are specific to Access Insight (AI) only can be skipped while configuring the Access Assurance Suite (AAS) related schema changes.
4. Execute the *DBServer.sql* script to create a linked server with the name Loopback.

Table 1: Variables in the DBServer.sql file (Sheet 1 of 4)

Variable Name	Application	Description
CustID	AI	Customer specific ID used to identify individual customer. For example its value can be "L1" or any other unique value.
InstallLogins	AAS and AI	It is used to specify whether or not any logins should be created. Set this value as "yes" in both installation types (AAS & AI). If the value of this variable is other than the value "yes", then no login is created/installed.
Install_AAS	AAS	It is used to specify whether to install the AAS specific configurations. By default, its value is set as "yes" while installing AAS. If its value is other than the value "yes", then no AAS specific configurations are performed.
Install_AI_Cust_Logins	AI	It is used to specify whether or not the AI specific logins should be created. By default, its value is set as "no" while installing the AAS configurations. Keep its value as "no" while installing the AAS related configurations. If its value is set as "yes" then the AI related configurations are also be installed.
DatabaseInstanceName	AAS	This is the name of the database server to which the loopback linked server points to as the source server. This value cannot be kept blank. By default, its value is set as "(local)". Specify the value with the instance name, if required. For example, "(local)\instanceName".

Table 1: Variables in the DBServer.sql file (Sheet 2 of 4)

Variable Name	Application	Description
DataCollectionCredentialName	AAS	This is the name of the Credential used in the data collection process. By default, its value is set as "DataCollectionCred". Do not change the value of this variable.
WindowsIdentityUserName	AAS	This is the username of the Windows Identity used while creating the credential with the name set in the variable DataCollectionCredentialName. This value should be in the format of domain\username. Also, this user should be at least a domain user, which is the minimum privilege required. The user is not required to have any permissions on the SQL Server or the database. This account is used as a SQL Server credential, which is required to run data collection rules using SQL jobs. **Note:** The variable values WindowsIdentityUserName and WindowsIdentityPassword are only required for executing the script. As soon as the execution is done, you must remove the username and password from the script.
WindowsIdentityPassword	AAS	This is the password of the Windows Identity used while creating the credential with the name set in the variable DataCollectionCredentialName.
DataCollectionProxyName	AAS	This is name of the proxy created for the data collection process in order to execute the PowerShell script through collectors. By default, its value is set as "CourionDataCollectionProxy". Do not change the value of this variable.
DataCollectionEntitiesUserName	AAS	This is the User ID of DataCollectionEntities connection string in the CustomerConnStrings.config file. If it is a windows user, specify it in the format of domain\username.
dbConnectionStringUserName	AAS	This is the User ID of the dbConnectionString connection string in the CustomerConnStrings.config file. If it is a windows user, specify it in the format of domain\username.
LoopbackLinkedServerName	AAS	This is the name of the linked server used in AAS. By default its value is set as "loopback". Do not change the value of this variable.
UAdmin	AI	This is the username of a user with Admin role privilege. Ignore it while installing the AAS.

Table 1: Variables in the DBServer.sql file (Sheet 3 of 4)

Variable Name	Application	Description
PAdmin	AI	This is the password for the user set in the UAdmin variable above. Ignore it while installing AAS.
UBasicReadOnly	AAS and AI	This is the username for the user with BasicReadOnly role privilege. This user is common in both AAS and AI installation. This can be a Windows or a SQL Server user and should only have the ReadOnly permission on the Courion database. If Data Feeds in Identity Mapping are planned to be referenced from databases other than the Courion database, then the same ReadOnly access needs to be provided to this user on those databases. No additional access should be granted to this read only account. **Note:** The variable values UBasicReadOnly and PBasicReadOnly are only required for executing the script. As soon as the execution is done, you must remove the username and password from the script. You need to provide a blank password such as "" in the PBasicReadOnly variable if you are using a Windows user for the UBasicReadOnly variable.
PBasicReadOnly	AAS and AI	This is the password of the user set in the variable UBasicReadOnly. In case of Windows authentication, keep the value as default or blank.
UBasicReadWrite	AI	This is the username of the user with the BasicReadWrite role privilege. Ignore it while installing AAS.
PBasicReadWrite	AI	This is the password of the user set in the variable UBasicReadWrite. Ignore it while installing AAS.
USecureReadOnly	AI	This is the username of the user with the SecureReadOnly role privilege. Ignore it while installing AAS.
PSecureReadOnly	AI	This is the password of the user set in the variable USecureReadOnly. Ignore it while installing AAS.
USecureReadWrite	AI	This is the username of the user with the SecureReadWrite role privilege. Ignore it while installing AAS.
PSecureReadWrite	AI	This is the password of the user set in the variable USecureReadWrite. Ignore it while installing AAS.

Table 1: Variables in the DBServer.sql file (Sheet 4 of 4)

Variable Name	Application	Description
DatabaseName	AAS	This is name of the database in which the linked server with the name set in the variable LoopbackLinkedServerName is to be created. By default its value is set as "master". Its value should always be "master". Do not change its default value in any case.
DefaultDataPath	AAS and AI	This is a default variable. Ignore this variable and keep its default value as "".
DefaultLogPath	AAS and AI	This is a default variable. Ignore this variable and keep its default value as "".

Installing the Courion Database

If you are licensed for the Access Request Manager, Access Certification, and Identity Mapping, you must follow these steps to create a database with the name Courion. This database contains the IdentityMap and other output tables produced by the Identity Mapping process, including the other working tables. Data feeds containing user accounts and identity data must reside in this database, or in other databases on the same database server as this database.

1. Using Microsoft® SQL Management Studio, open the *Courion.sql* file (which is located in the Courion installation folder).
2. Select SQL CMD Mode under **QUERY** in Microsoft® SQL Management Studio.
3. Configure the variables explained in Table 2.
4. Execute the *Courion.sql* database script.

Table 2: Variables in the Courion.sql File (Sheet 1 of 2)

Variable Name	Description	Example
AASServer	Name of the Application Server (in square brackets) where AAS is installed	:setvar AASServer "[MyServer.Dom]"
LoopbackLinkedServerName	The default value is “loopback”. This should match with the value of the variableLoopbackLinked ServerName" in the DBServer.sql file	:setvar LoopbackLinkedServerName "loopback"
UBasicReadOnly	The value for this variable should match with the value of the variable UBasicReadOnly in the DBServer.sql file	:setvar UBasicReadOnly "BasicReadOnly"
DatabaseName	The Courion database name	:setvar DatabaseName "Courion"

Table 2: Variables in the Courion.sql File (Sheet 2 of 2)

Variable Name	Description	Example
DefaultDataPath	This is a default variable. Ignore this variable and keep its default value as "".	:setvar DefaultDataPath ""
DefaultLogPath	This is a default variable. Ignore this variable and keep its default value as "".	:setvar DefaultLogPath ""

Installing the Core Access Assurance Suite

This section describes how to install the Core Access Assurance Suite.

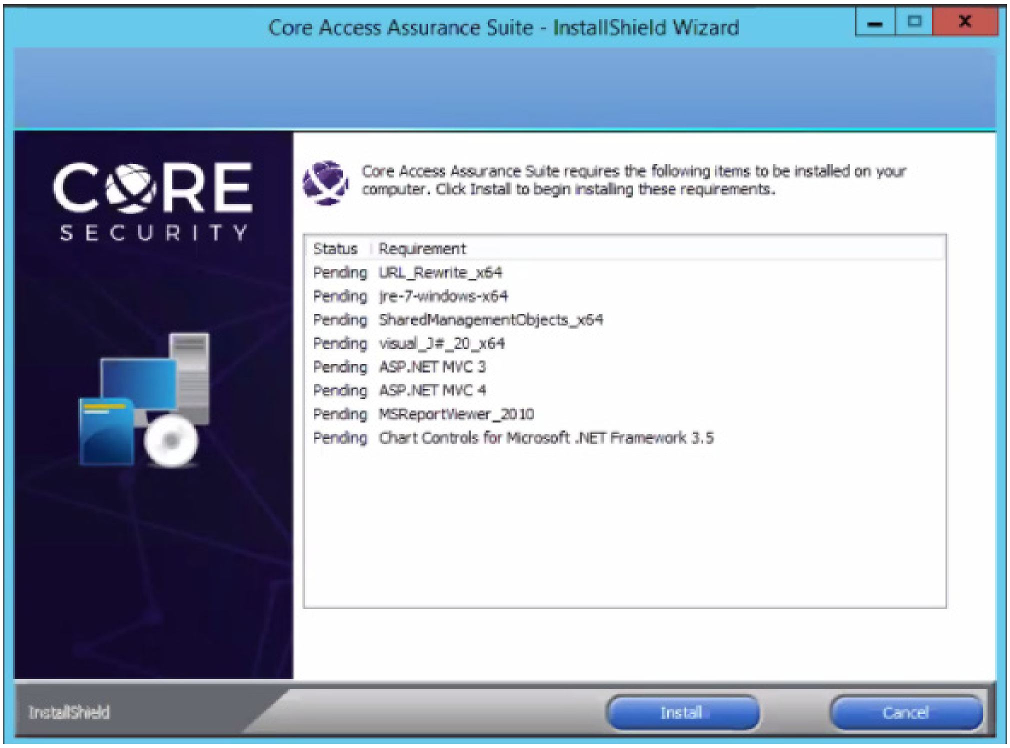
Release Notes

After completing the installation, you can view the Release Notes associated with this release of Core Access Assurance Suite software. The Release Notes include important information about how this release affects any features you have configured in a previous release. Core Security strongly recommends that you consult the Release Notes.

Starting the Installation

1. Log on to the Windows system as a user with administrator privileges.
2. Download file `CoreInstall.exe` from the Core Security Support website and then execute it by double-clicking the filename. The Access Assurance Suite installation checks for required components and offers to install them as shown in *Figure 3*.

Figure 3: Required Components



Core Access Assurance Suite - InstallShield Wizard

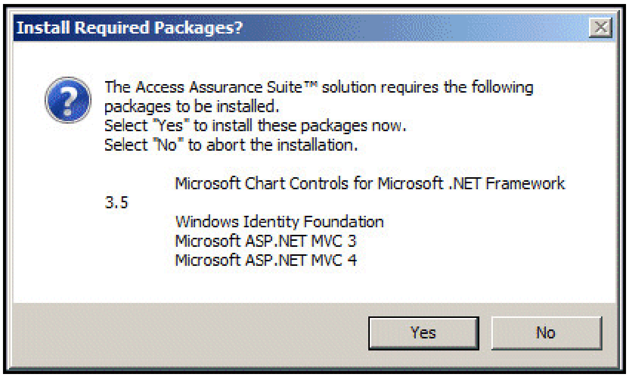
Core Access Assurance Suite requires the following items to be installed on your computer. Click Install to begin installing these requirements.

Status	Requirement
Pending	URL_Rewrite_x64
Pending	jre-7-windows-x64
Pending	SharedManagementObjects_x64
Pending	visual_J#_20_x64
Pending	ASP.NET MVC 3
Pending	ASP.NET MVC 4
Pending	MSReportViewer_2010
Pending	Chart Controls for Microsoft .NET Framework 3.5

3. Click **INSTALL**.

   The Core Access Assurance Suite requires several third-party applications as described in the product requirements. If these applications are not already installed on the server, a dialog box similar to *Figure 4* appears.

Figure 4: Install Required Packages



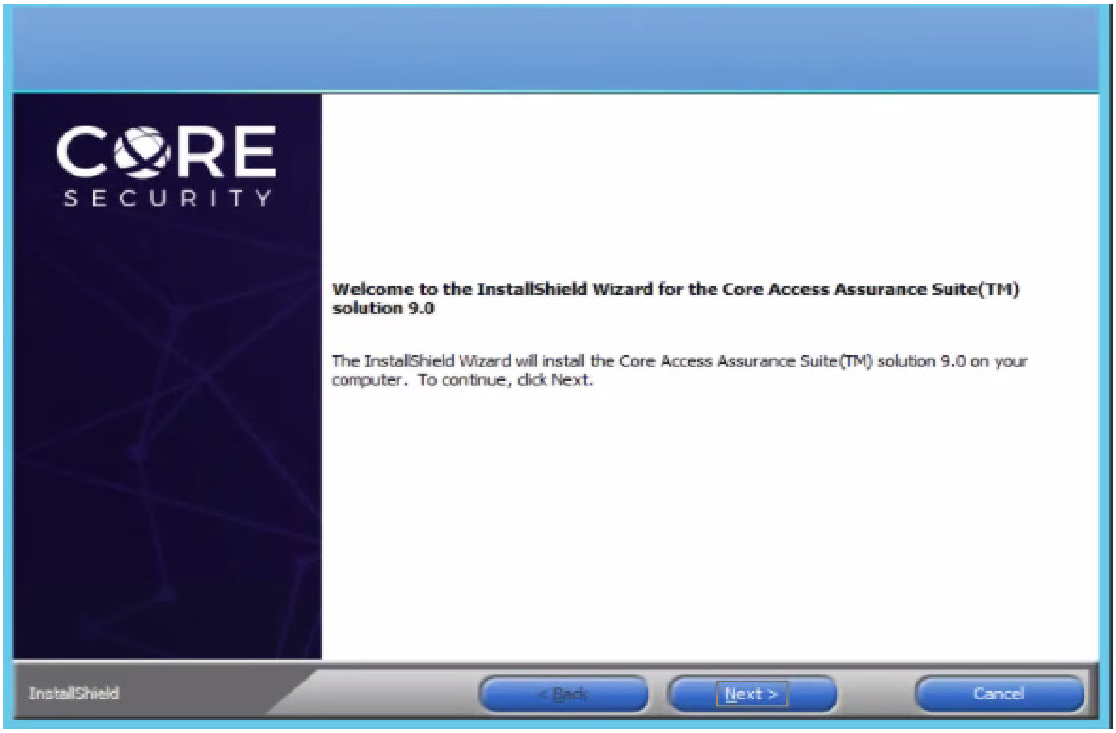
Install Required Packages?

The Access Assurance Suite™ solution requires the following packages to be installed.
Select "Yes" to install these packages now.
Select "No" to abort the installation.

Microsoft Chart Controls for Microsoft .NET Framework
3.5
Windows Identity Foundation
Microsoft ASP.NET MVC 3
Microsoft ASP.NET MVC 4

4. Click Yes.The applications are installed and then the Welcome dialog box appears as shown in *Figure 5*.

Figure 5: Welcome

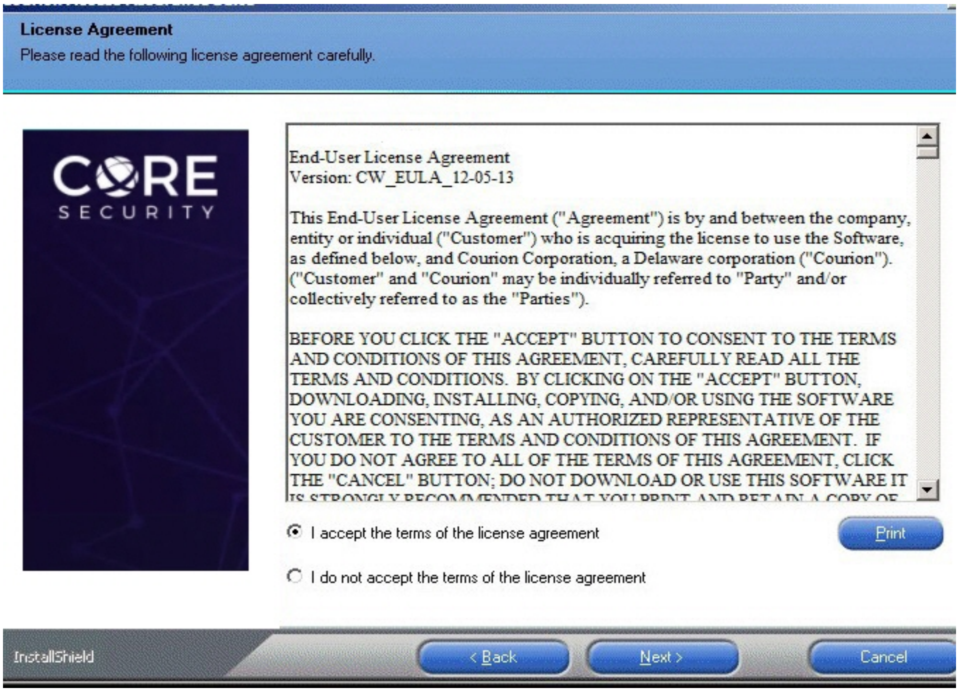


Welcome to the InstallShield Wizard for the Core Access Assurance Suite(TM) solution 9.0

The InstallShield Wizard will install the Core Access Assurance Suite(TM) solution 9.0 on your computer. To continue, click Next.

5. Click the **Next** button.The installation program displays the Core Access Assurance Suite license agreement as shown in *Figure 6*.

Figure 6: License Agreement



License Agreement

Please read the following license agreement carefully.

End-User License Agreement
Version: CW_EULA_12-05-13

This End-User License Agreement ("Agreement") is by and between the company, entity or individual ("Customer") who is acquiring the license to use the Software, as defined below, and Courion Corporation, a Delaware corporation ("Courion"). ("Customer" and "Courion" may be individually referred to "Party" and/or collectively referred to as the "Parties").

BEFORE YOU CLICK THE "ACCEPT" BUTTON TO CONSENT TO THE TERMS AND CONDITIONS OF THIS AGREEMENT, CAREFULLY READ ALL THE TERMS AND CONDITIONS. BY CLICKING ON THE "ACCEPT" BUTTON, DOWNLOADING, INSTALLING, COPYING, AND/OR USING THE SOFTWARE YOU ARE CONSENTING, AS AN AUTHORIZED REPRESENTATIVE OF THE CUSTOMER TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE "CANCEL" BUTTON; DO NOT DOWNLOAD OR USE THIS SOFTWARE IT IS STRONGLY RECOMMENDED THAT YOU PRINT AND RETAIN A COPY OF

● I accept the terms of the license agreement

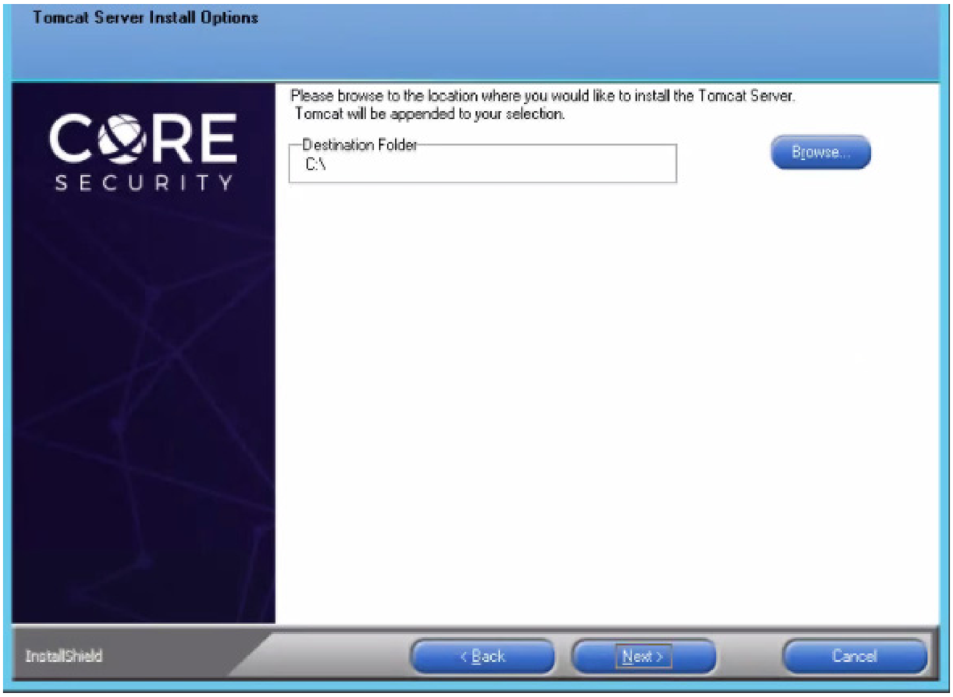
○ I do not accept the terms of the license agreement

6. Read and accept this agreement by clicking **I ACCEPT...** and then click **NEXT** to continue installing the Core Access Assurance Suite.

   If you click **CANCEL**, installation stops.

   The Tomcat Server Install Options dialog box appears as shown in *Figure 7*. Tomcat is required on the same system as the Web server in order to run ETL operations.

Figure 7: Location for Tomcat Server Files



Tomcat Server Install Options

Please browse to the location where you would like to install the Tomcat Server. Tomcat will be appended to your selection.

Destination Folder: C:\

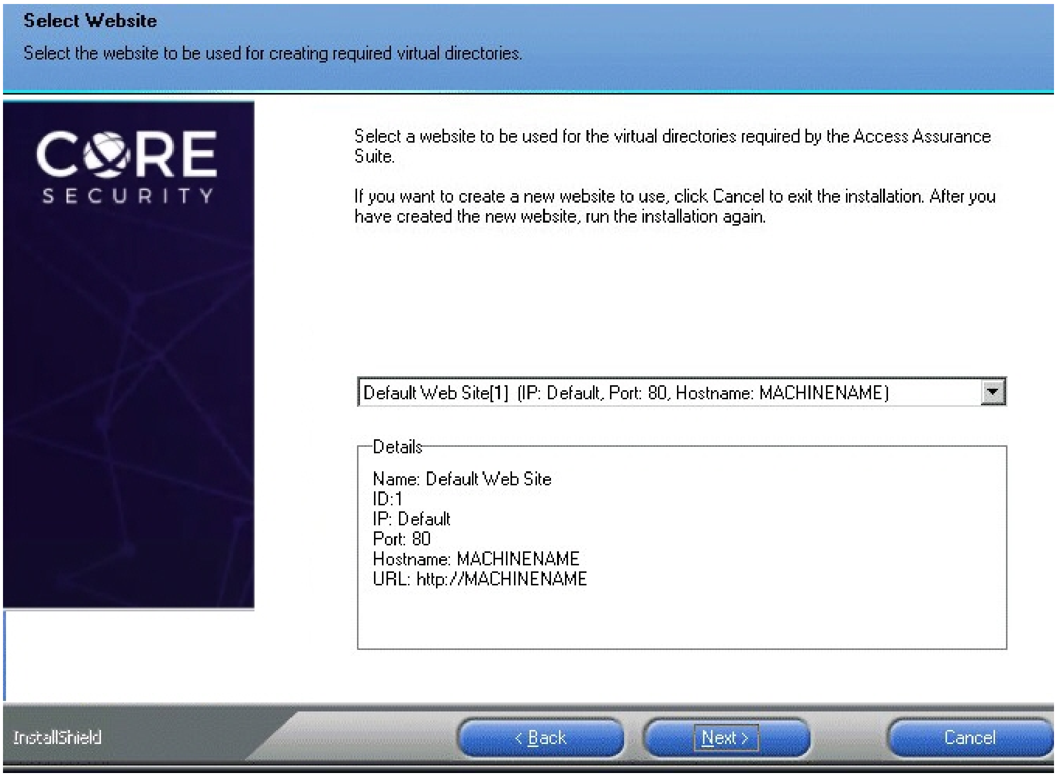
7. Click the **BROWSE...** button and browse to the location where the Tomcat Server files should be installed. When prompted to create a new directory, click **YES**.

8. To accept the default path, click **NEXT.** To choose a different location, click the **CHANGE** button and from the folder selection dialog box select the appropriate folder.

   The software then warns you about any services that are stopped during the installation. To continue, click **YES.**

   If this is an initial installation of the Core Access Assurance Suite, the Select Website dialog box appears, as shown in *Figure 8*. (This dialog box does not appear during update installations.)

Figure 8: Select Website



Select Website

Select the website to be used for creating required virtual directories.

Select a website to be used for the virtual directories required by the Access Assurance Suite.

If you want to create a new website to use, click Cancel to exit the installation. After you have created the new website, run the installation again.

Default Web Site[1] (IP: Default, Port: 80, Hostname: MACHINENAME)

Details

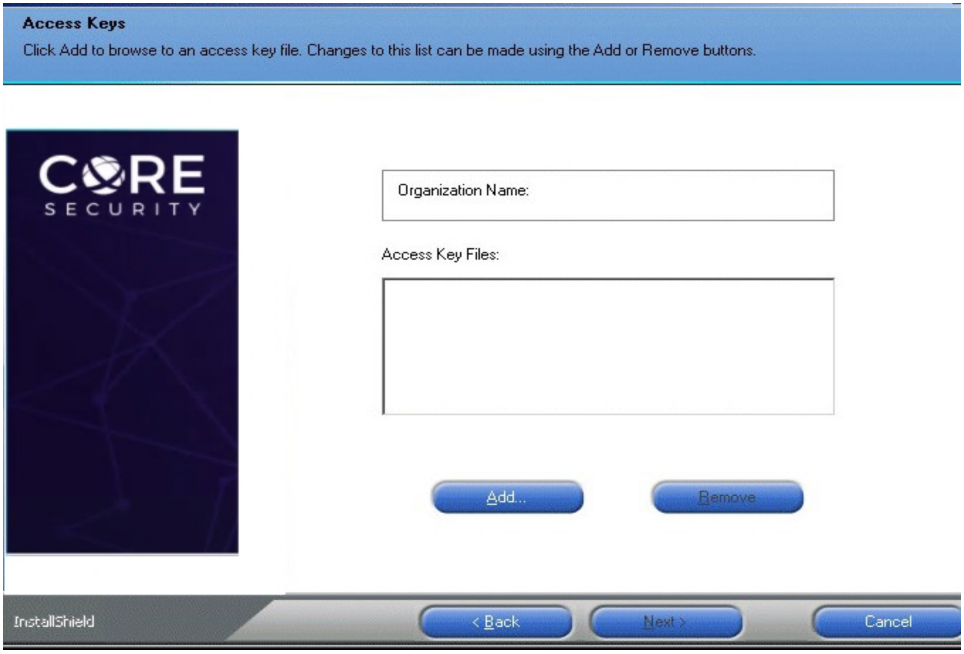
Name: Default Web Site
ID:1
IP: Default
Port: 80
Hostname: MACHINENAME
URL: http://MACHINENAME

9. The Select Website dialog box enables you to choose a non-default web site for installation of the Courion virtual directories and for hosting the Connector Framework Manager, Connector Framework, and Publisher web services.

   To accept the default, click **NEXT**. You can select a non-default web site from the drop-down list or create a new non-default web site. To create a new non-default web site, click **CANCEL** to exit from the installation, create the new web site, and run the installation again. The new web site appears in the drop-down list.

10. The Access Keys dialog box appears, as shown in *Figure 9*.

Figure 9: Access Keys



Access Keys

Click Add to browse to an access key file. Changes to this list can be made using the Add or Remove buttons.

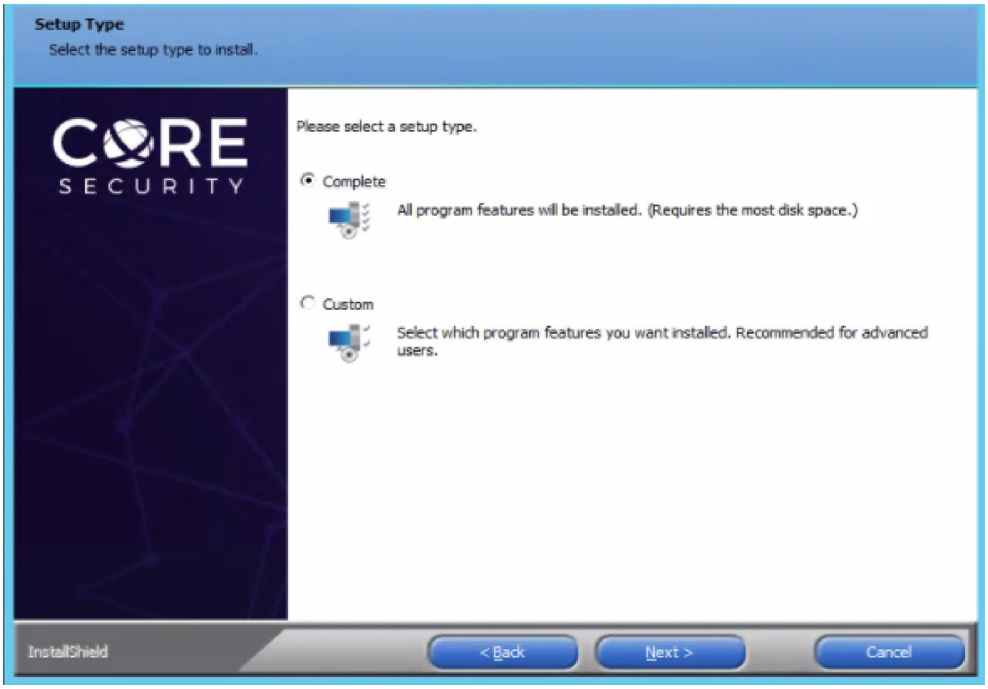
Organization Name:

Access Key Files:

11. Click the **ADD...** button and browse to find and select the key file (supplied by Core Security), and then click **OPEN.** You can add more than one key file.

    Click **NEXT**. The Setup Type dialog box appears, as shown in *Figure 10*.

Figure 10: Setup Type



Setup Type

Select the setup type to install.

Please select a setup type.

● Complete

All program features will be installed. (Requires the most disk space.)

○ Custom

Select which program features you want installed. Recommended for advanced users.

12. On the Setup Type dialog box, choose a setup option to determine which features are installed:

Complete — This option installs all features of the Core Access Assurance Suite on a single server.

Custom — This option allows you to choose individual features of the Core Access Assurance Suite to install, such as the Core Server or individual web services. Choose this option if you plan on implementing a distributed installation. For details, see *“Single Server and Distributed Server Installation” on page 16*.

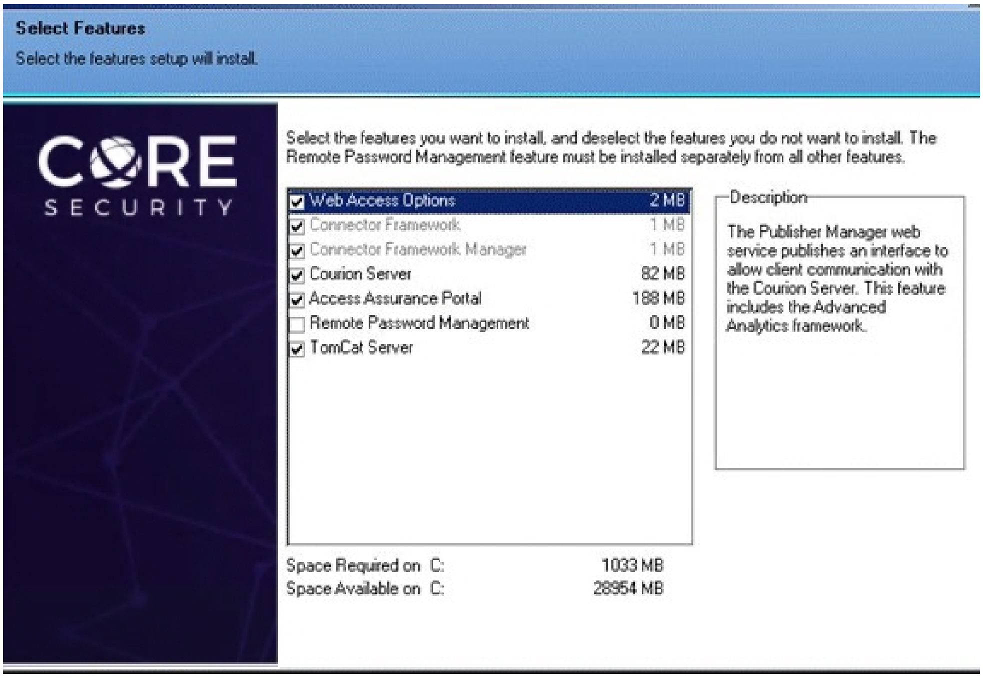
Note: Distributed installation is only available for the provisioning platform.

You should also choose this option if you are installing the Remote Password Management feature for the classic platform.

Select the appropriate setup option and click **Next**. If you select the **Custom** option, the Select Features dialog box appears, as shown in *Figure 11*. If you select the **Complete** option, the Start Copying Files dialog box appears, as shown in *Figure 12*.

Note: The installation prompts you to click Yes to stop the World Wide Web Publishing Service during the installation. Click Yes to continue.

Figure 11: Select Features



Select Features

Select the features setup will install.

Select the features you want to install, and deselect the features you do not want to install. The Remote Password Management feature must be installed separately from all other features.

☑ Web Access Options	2 MB
☑ Connector Framework	1 MB
☑ Connector Framework Manager	1 MB
☑ Courion Server	82 MB
☑ Access Assurance Portal	188 MB
□ Remote Password Management	0 MB
☑ TomCat Server	22 MB

Description

The Publisher Manager web service publishes an interface to allow client communication with the Courion Server. This feature includes the Advanced Analytics framework.

Space Required on C:	1033 MB
Space Available on C:	28954 MB

13. On the Select Features dialog box, choose which features you want to install:

Web Access Options — The Publisher web service publishes an interface to allow clients to communicate with the Core Access Assurance Suite.

Connector Framework — The Connector Framework web service provides connectivity to the various targets in your enterprise.

Connector Framework Manager — The Connector Framework Manager web service manages the target configuration of all supported connectors. This service also manages communication between the Core Access Assurance Suite and the Connector Framework.

Core Server — The Core Server allows you to configure workflows for Password Reset and Provisioning actions.

Access Assurance portal - portal-based interfaces, including the Core Access solution and the Core Compliance policy verification solution.

DATA COLLECTION - identifies sources of account, profile, and entitlement data including HR systems or Active Directory.

IDENTITY MAPPING - provides a user-friendly interface for an administrator to establish an enterprise-wide Data Mapping process.

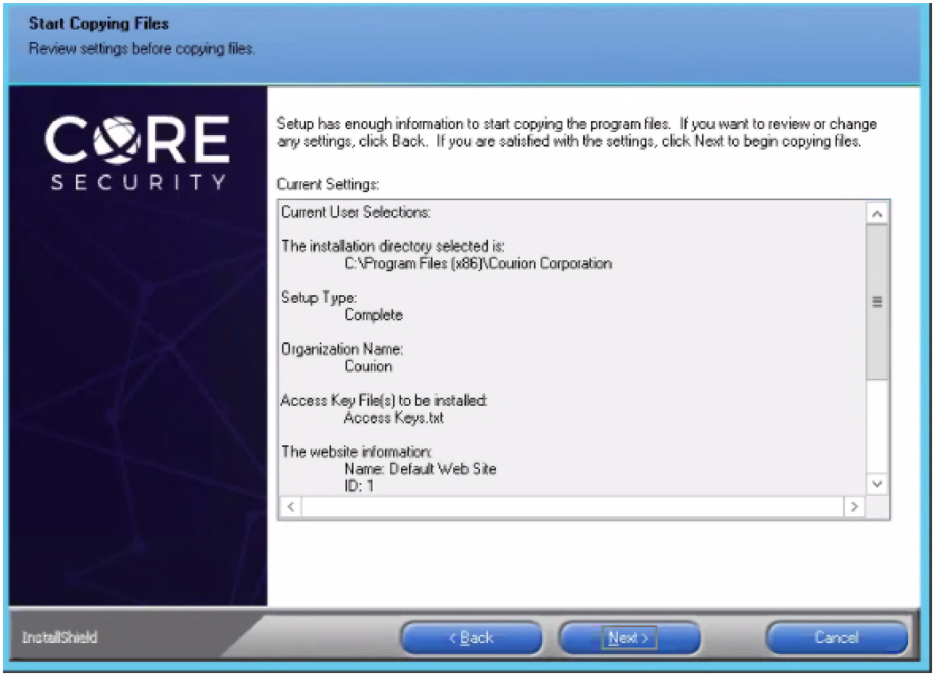
REMOTE PASSWORD MANAGEMENT — The Remote Password Management feature allows users to execute remote password reset requests. It is only available for use with PasswordCourier Classic.

The ability to perform remote resets removes requirements such as needing a domain trust relationship between the target domain and the domain in which the Core Server is operating. The Remote Password Management feature is intended for installation on a separate server from the other Courion services.

If you select the Remote Password Management feature, you cannot select any other features on this server. If you select this feature with any of the other features and click **NEXT**, an error message appears, requiring you to deselect either Remote Password Management or the other features.

Select the desired features and click **NEXT**. A warning appears informing you of any services that are stopped during installation. Click **YES** to continue. The Start Copying Files dialog box appears, as shown in *Figure 12*.

Figure 12: Start Copying Files



Start Copying Files

Review settings before copying files.

Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

Current User Selections:

The installation directory selected is:
C:\Program Files (x86)\Courion Corporation

Setup Type:
Complete

Organization Name:
Courion

Access Key File(s) to be installed:
Access Keys.txt

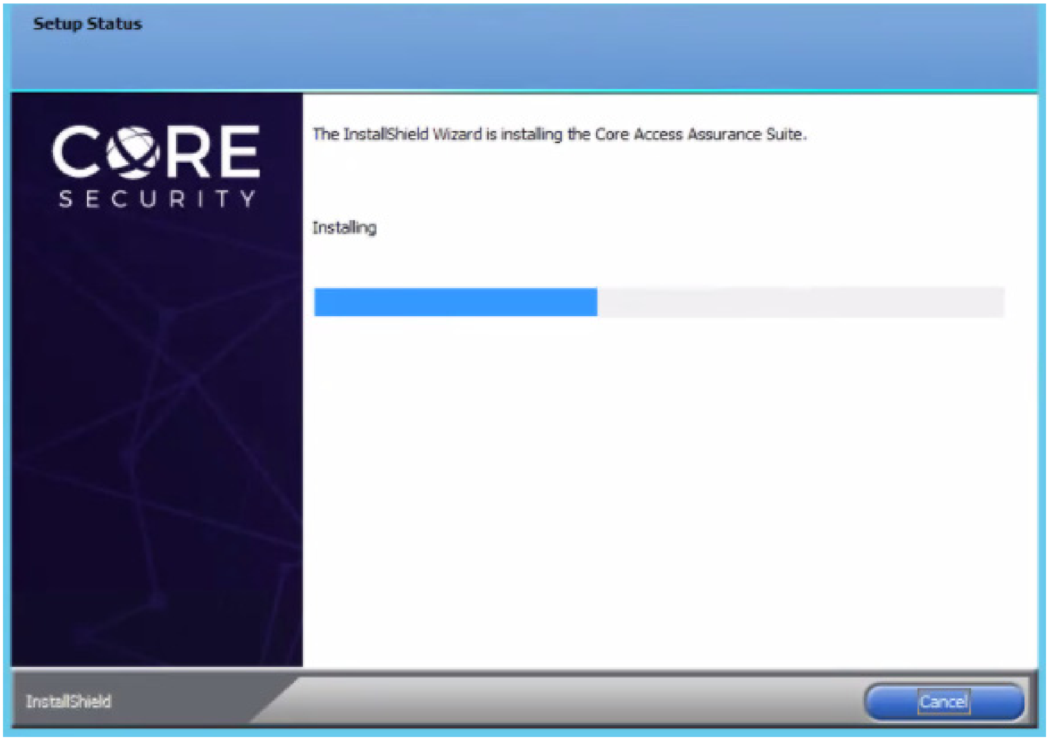
The website information:
Name: Default Web Site
ID: 1

14. If you want to change any install settings, click **BACK**; then make the changes you want. To quit the install process, click **CANCEL**.

    To begin copying files, click **NEXT.** Status messages appear that indicate the progress of the installation.

    The Setup Status dialog box, shown in *Figure 13*, indicates that installation is proceeding.

Figure 13: Setup Status



The InstallShield Wizard is installing the Core Access Assurance Suite.

Installing

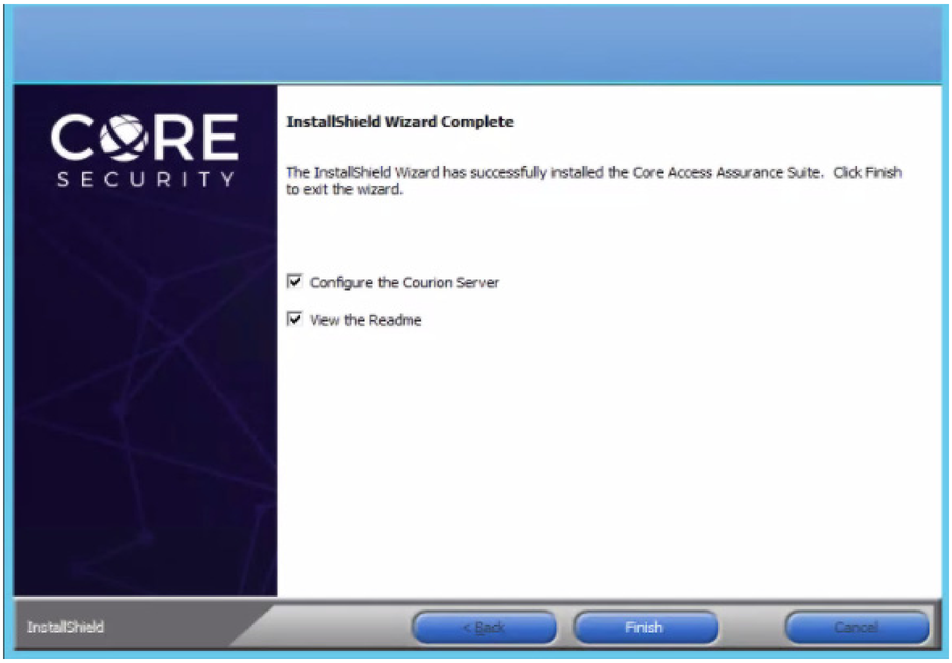
The software installs files, including sample web pages. It also registers services, creates a start menu, and creates a virtual directory. Installation takes a few minutes. In some instances, your computer is required to reboot before you proceed.

15. If you do not have a current version of JVM™ (Java Virtual Machine), a dialog box appears. Click **YES** to install JVM.

16. When the installation is complete, an InstallShield Wizard Complete dialog box appears. The options available on the dialog box vary depending on what you have chosen to install.

    Figure 14 shows the dialog box that appears if you have specified a Complete installation or if you have selected the Core Server option.

Figure 14: InstallShield Wizard Completed



InstallShield Wizard Complete

The InstallShield Wizard has successfully installed the Core Access Assurance Suite. Click Finish to exit the wizard.

☑ Configure the Courion Server

☑ View the Readme

By default, the options on this dialog box are always selected.

The first option depends on what you have chosen to install:

> **CONFIGURE THE CORE SERVER** — This is displayed if you selected a Complete installation or if you chose Core Server with or without any other option in a Custom installation.
>
> **CONFIGURE PASSWORD MANAGEMENT MODULES** — This is displayed if you selected the Remote Password Management option.

The second option is as follows:

> **VIEW THE README FILE** — The Readme File offers useful information about this release.

17. Click **FINISH**.

    Installation of the selected Core Access Assurance Suite features is complete.

    If you selected **CONFIGURE THE CORE SERVER**, the Configuration Manager starts automatically. See *“Running the Core Server Configuration Manager” on page 42* for details.

    If you install one or more of the web service components without installing the Core Server, a configure checkbox does not appear. Instead, the Web Service Configuration Manager launches automatically after you click **FINISH**. See *“Configuring the Web Service Options” on page 73* for details.

    If you left the Release Notes box checked, the software displays the Release Notes. If you want to read the Release Notes at later time, it is located in the default installation folder as `ReleaseNotes_ProvGov.rtf`.

Applet and ASP (Active Server Page) Installation

Before you can access Core Access Assurance Suite products via a browser, the appropriate files must be in a publishing folder for a web server. Normally, the installation software does this for you by creating a virtual directory named `Courion`. This section only applies if you specified a different virtual directory in Step 9 of *"Starting the Installation"* on page 32.

Java™ Pages

Note: PasswordCourier Classic and PasswordCourier Support Staff Classic only.

PasswordCourier Classic and PasswordCourier Support Staff Classic are the only applications in the Core Access Assurance Suite that use Java™ in addition to ASP.

The CourionService "www" folder created during the Core Access Assurance Suite installation contains the supporting files for Java™. Make this folder available on the web server.

If a non-default TCP/IP port was specified during setup, then you must update each applet's HTML file with the new port value. The HTML files to update are in the folder `javacode` folder `pwc_java`. The parameter to modify is `<PARAM name= "PORT" value="8189">`.

To redirect end users to a specific URL at the completion of using the Core Access Assurance Suite, PARAMs must be added to the HTML. Two different redirections may be specified based on the success or non-success of the action. For example, in PasswordCourier:

- `<PARAM=SuccessRedirectTo VALUE="http://www.company.com/pwdcour/pwdresetsucc.html">`
- `<PARAM=SuccessRedirectTarget VALUE="_self">`
- `<PARAM=NonSuccessRedirectTo VALUE="http://www.company.com/pwdcour/pwdresetnonsucc.html">`
- `<PARAM=NonSuccessRedirectTarget VALUE="_self">`

If these PARAMs are present, the end users are redirected to the specified HTML page upon clicking the **FINISH** button. This button appears after the action request is completed for the Core Access Assurance Suite.

Java™ technology-enabled applets are installed as part of the Core Access Assurance Suite setup. Core Security suggests that you restrict Web access to the administrator applets, *i.e.* the various configuration managers. Core Security recommends restricting access to the online administrator's documentation. Consult the HTTP server documentation for information about restricting access to web pages.

ASP Pages

Information on ASP configuration is available in the chapter on "Web Access (ASP) Configuration" in the *Core Access Assurance Suite Implementation Guide.*

Removing the Core Access Assurance Suite

If you use Windows Add/Remove Software function to remove the Core Access Assurance Suite from the system, the removal will be incomplete. All of the following steps are required for a complete removal:

1. Save file `cfgfile.db`, which is the configuration repository, and any other supplied files (such as scripts) that you have modified since installation to a folder outside the Core Access Assurance Suite installation folder. Later, if needed, you can restore the repository and any modified files.
2. Uninstall the Core Access Assurance Suite from the control panel **ADD/ REMOVE SOFTWARE** function.

   If you plan to install any version of the Core Access Assurance Suite, make sure the **REMOVE SOFTWARE** operation has completed. If you try to install before the **REMOVE** operation completes, the installation stops with an “Access Denied” error message.
3. Remove configuration data in the WMI repository. The Core Access Assurance Suite stores configuration data in both the `cfgfile.db` database and the Windows Management Instrumentation (WMI) repository. Core Security supplies `.mof` files that you can use to remove the data in the WMI repository. In a typical installation, the files containing this data are found in the C:\Program Files\Courion Corporation\WBEM folder.

   To remove WMI configuration data, open a command prompt and change directories to the C:\Program Files\Courion Corporation\WBEM folder.

   Execute the following command:

   ```
   mofcomp.exe CourWMIObjects_deleteNamespaces.mof
   ```

   If you want to remove only configuration data for specific features of the product, Core Security has supplied `.mof` files for each feature.

   To remove "Connector Framework" configuration data, execute the following command:

   ```
   mofcomp.exe CourWMIObjectInstances_CF2.mof
   ```

   To remove "Connector Framework Manager" configuration data, execute the following command:

   ```
   mofcomp.exe CourWMIObjectInstances_CFM2.mof
   ```

   To remove "Publisher" configuration data, execute the following command:

   ```
   mofcomp.exe CourWMIObjectInstances_pub2.mof
   ```

   To remove "Core Server" configuration data, execute the following command:

   ```
   mofcomp.exe CourWMIObjectInstances_plat2.mof
   ```
4. Delete the Core Security folder where the Core Access Assurance Suite was installed. (You may need to delete each folder within the Core Security folder first, then delete the folder itself.)
5. Clear your browser’s cache. To clear a Microsoft Explorer cache, use the sequence Tools > Internet Options. Then in section **TEMPORARY INTERNET FILES**, click the **DELETE FILES** button and confirm.
6. Delete the registry entries for Courion using the Registry Editor. Click **START > RUN > REGEDIT**.

7. Do one of the following:
   - For 32-bit systems, delete the HKEY_LOCAL_MACHINE\SOFTWARE\Courion Corporation subkey.
   - For 64-bit systems, delete the HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Courion Corporation subkey.
8. Shutdown Windows to apply the registry changes.
9. Restart Windows.

Next Steps

Once you have installed the Core Access Assurance Suite and configured the Core Server (and any necessary web services in a distributed installation), you can configure connectors, PMMs, and workflows, as described in the following manuals:

- For connectors and password management modules, *Configuring Password Management Modules (PMMs), Connectors, and Agents.*
- For workflow configuration on the provisioning platform, *Configuring Workflows with the Core Access Assurance Suite Administration Manager*.
- For PasswordCourier Classic and PasswordCourier Support Staff Classic, *Using PasswordCourier and PasswordCourier Support Staff Classic*.

Chapter 3: Configuring the Core Server

This chapter describes how to configure the Core Server, using standard configuration. It includes the following topics:

- *"Running the Core Server Configuration Manager" on page 42*
- *"Additional Configuration for the Access Assurance Portal Components" on page 67*

Running the Core Server Configuration Manager

The Core Server Configuration Manager is a wizard that walks you through server configuration.

If you selected the **CONFIGURE THE CORE SERVER** option at the end of installation, the Configuration Manager starts automatically. If the server rebooted and the Configuration Manager did not start automatically, you need to start it manually.

For a Windows server, start Configuration Manager as follows:

Start>All Programs>Core Access Assurance Suite>Configuration Manager

Note: Before using the Configuration Manager, make sure that you (or another administrator) have installed the Loopback server and the Courion database as described in the previous chapter.

You need the following information:

- The number of the TCP/IP port on which the CourionService listens.

You may also need the following if you are configuring the classic platform:

- For ODBC or LDAP configuration, a privileged username and password, and database and server names.
- For Help Desk or database system configuration, the server name, the Help Desk username and password, and the applicable information for any other fields specific to the Help Desk/database system.
- The SMTP server hostname or IP address and domain name that the Core Server uses for communication with connected systems.

If you have access keys for Core Provisioning, Core Compliance, RoleCourier, Core Password, or ProfileCourier, a separate Connector Configuration Manager configures connectors. A connector allows access to system resources, such as the profile data source and Help Desk application. Connectors are described in the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents*.

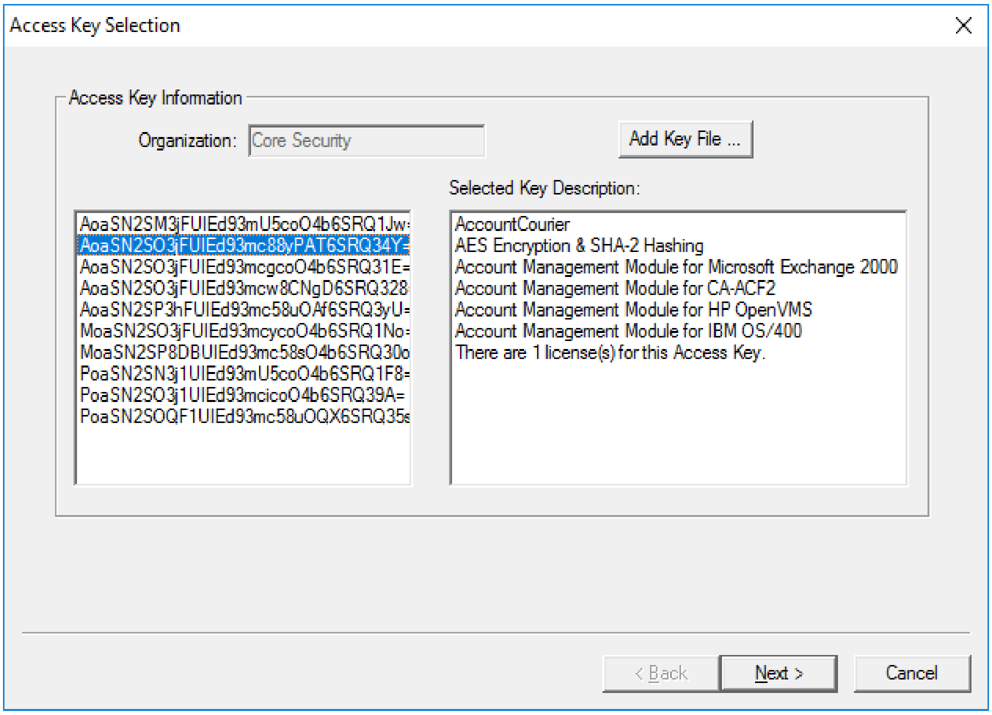
If you have access keys for Core Password, the Configuration Manager lets you configure Password Management Modules (PMMs).

Note: If you add or delete access keys, you need to restart IIS for the changes to take effect.

Access Keys

The Access Key Selection dialog box appears, as shown in *Figure 15*.

Figure 15: Access Key



Organization: Core Security

```
AoaSN2SM3jFUIEd93mU5coO4b6SRQ1Jw=
AoaSN2SO3jFUIEd93mc88yPAT6SRQ34Y=
AoaSN2SO3jFUIEd93mcgcoO4b6SRQ31E=
AoaSN2SO3jFUIEd93mcw8CNgD6SRQ328
AoaSN2SP3hFUIEd93mc58uOAf6SRQ3yU=
MoaSN2SO3jFUIEd93mcycoO4b6SRQ1No=
MoaSN2SP8DBUIEd93mc58sO4b6SRQ30o
PoaSN2SN3j1UIEd93mU5coO4b6SRQ1F8=
PoaSN2SO3j1UIEd93mcicoO4b6SRQ39A=
PoaSN2SOQF1UIEd93mc58uOQX6SRQ35s
```

Selected Key Description:

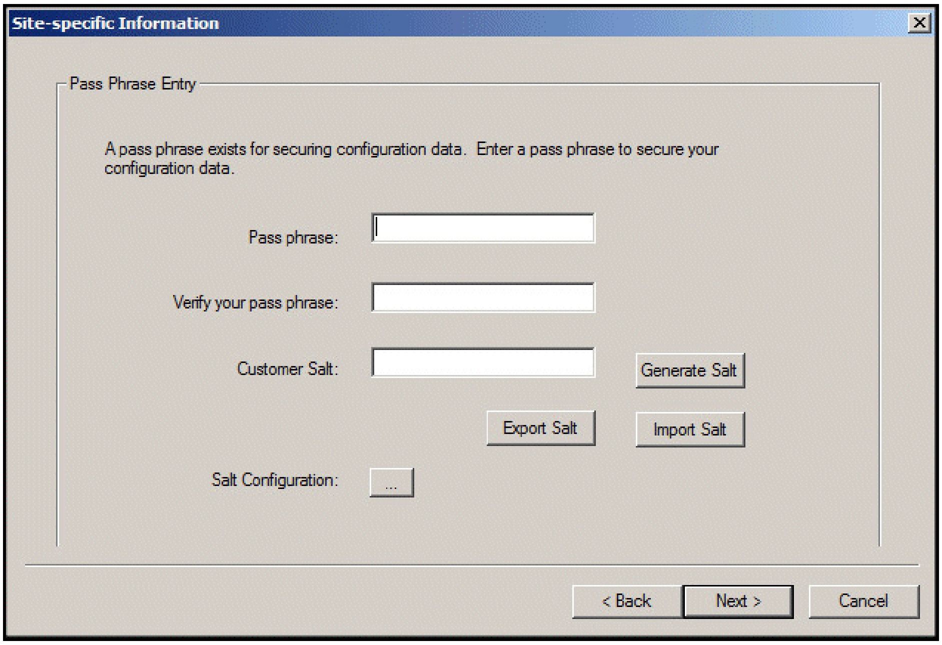
AccountCourier
AES Encryption & SHA-2 Hashing
Account Management Module for Microsoft Exchange 2000
Account Management Module for CA-ACF2
Account Management Module for HP OpenVMS
Account Management Module for IBM OS/400
There are 1 license(s) for this Access Key.

1. During installation, the administrator added access keys. Those keys appear in the list on the left. If you highlight a key, a description appears. To add more keys, click **ADD KEY FILE**.
2. Click the **NEXT** button.

Site-specific Information

The Site-specific Information dialog box helps keep your configuration data secure. If you enter a non-evaluation access key, the Site-specific Information dialog box appears, as in *Figure 16*.

Figure 16: Site-specific Information



Site-specific Information

Pass Phrase Entry

A pass phrase exists for securing configuration data. Enter a pass phrase to secure your configuration data.

Pass phrase:

Verify your pass phrase:

Customer Salt:

The pass phrase you enter in the Site-specific Information dialog box generates an encryption key. This key is used to encrypt configuration data. The key itself is encrypted using Microsoft CryptoAPI.

3. Enter a pass phrase.

4. **Follow your company's security policies to store this pass phrase in a secure location.**

   Note: The pass phrase is required to run the Encryption Update Utility for Shared Data. Shared data includes Data Security Utility (DSU) settings in the cfgfile.dbfile, data in the Transaction Repository, and data in the Courion database such as the challenge data and collection rules data. This utility is required as part of upgrading the Access Assurance Suite. If you change the pass phrase for your installation, you need to run this utility to re-encrypt the data.

   If there is a pass phrase mismatch while upgrading the Access Assurance Suite, the encrypted Shared Data will not be decrypted successfully. As a result, Data Collection rules and Provisioning Workflows will not execute successfully.

5. In the next field, enter the pass phrase again to verify it against the first field.

   Note: If you installed the Access Assurance Suite in a distributed server configuration, you must use the same pass phrase on all servers hosting the different components. If you decide to change the pass phrase at any time, be sure to change it on all servers as described in *"Configuring Access Keys and a Pass Phrase" on page 76*.

6. The Access Assurance Suite supports the SHA2 algorithm for hashing data. The algorithm computes a sixty-four (64) byte binary message digest from an arbitrary length preimage. To create a hash value from the clear text value, the Suite requires three inputs: a cleartext value, a customer salt value of at least 90 characters in length, and a random salt value.

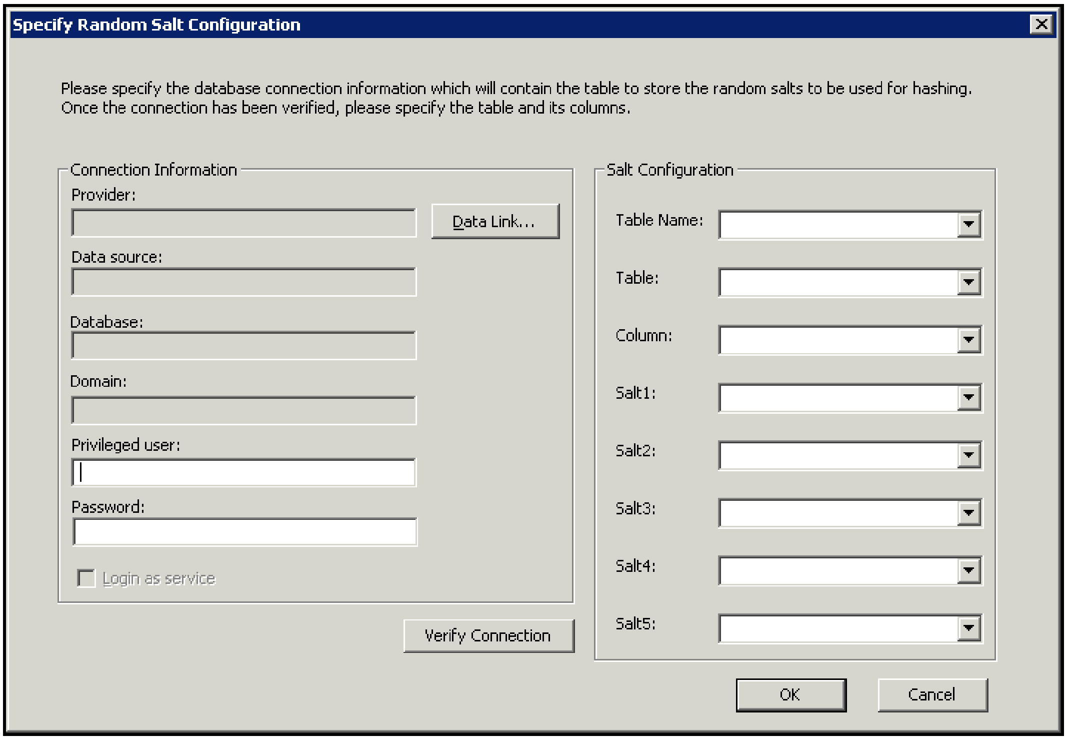
   In the **CUSTOMER SALT** field, enter a string of at least 90 characters. For ease of use, click **GENERATE SALT** to auto-generate a random salt.

Note: The customer salt should be set and never modified. If the customer salt is modified, all of the hashed data including password history will not be usable. The data will have to be created again through the workflows.

For distributed installations, you can use the **Export Salt** button to save the salt to a file and then copy that file to every Core Server. On each Core Server, use the **Import Salt** button to ensure that every server uses the same customer salt value.

7. Click the **Salt Configuration...** button.The Specify Random Salt Configuration dialog box appears.

Figure 17: Specify Random Salt Configuration Dialog Box



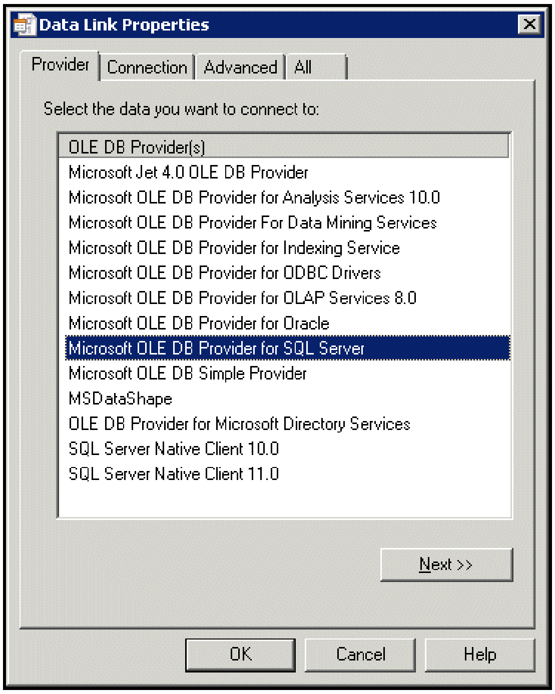
Specify Random Salt Configuration

Please specify the database connection information which will contain the table to store the random salts to be used for hashing. Once the connection has been verified, please specify the table and its columns.

8. Click **Data Link**. The Data Link Properties dialog box appears.

9. Select Microsoft OLE DB Provider for SQL Server as shown in *Figure 18*.

Figure 18: Data Link Properties for Salt Configuration - Provider

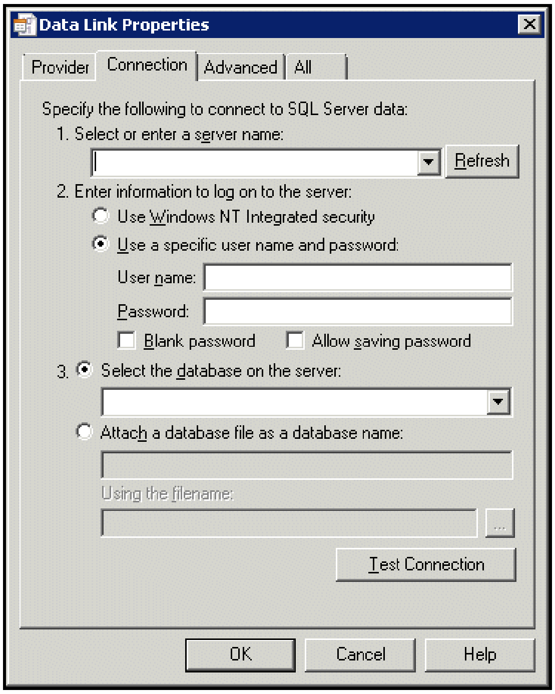


Select the data you want to connect to:

OLE DB Provider(s)
- Microsoft Jet 4.0 OLE DB Provider
- Microsoft OLE DB Provider for Analysis Services 10.0
- Microsoft OLE DB Provider For Data Mining Services
- Microsoft OLE DB Provider for Indexing Service
- Microsoft OLE DB Provider for ODBC Drivers
- Microsoft OLE DB Provider for OLAP Services 8.0
- Microsoft OLE DB Provider for Oracle
- Microsoft OLE DB Provider for SQL Server
- Microsoft OLE DB Simple Provider
- MSDataShape
- OLE DB Provider for Microsoft Directory Services
- SQL Server Native Client 10.0
- SQL Server Native Client 11.0

10. Click **NEXT**. The Connection tab appears.

Figure 19: Data Link Properties for Salt Configuration - Connection



Specify the following to connect to SQL Server data:

1. Select or enter a server name:

2. Enter information to log on to the server:
   ○ Use Windows NT Integrated security
   ● Use a specific user name and password:
   User name:
   Password:
   □ Blank password □ Allow saving password

3. ● Select the database on the server:
   ○ Attach a database file as a database name:
   Using the filename:

11. When hashing occurs, AAS establishes a connection to the database which contains the random salt configuration values. In the **SERVER NAME** field, specify the name of the database server.

12. The dialog box also prompts you to connect to the database by specifying connection parameters:

- For Windows NT Integrated authentication, select **USE WINDOWS NT INTEGRATED SECURITY.**
- For SQL authentication, specify the following:
  f. **USER NAME**: The username of the SQL Server Login user; the minimum SQL Server Role that this user needs is “public”.
  g. **PASSWORD**: The password of the SQL Server Login user

13. **SELECT THE DATABASE ON THE SERVER**: From the drop-down list, select the database created using the Courion.sql file
14. Click **TEST CONNECTION** to verify the credentials.
15. Click **OK** to close the Data Link Properties dialog box.
16. In the Specify Random Salt Configuration dialog box, specify the following connection parameters:
    - **DOMAIN**: For Windows NT Integrated authentication, specify the Active Directory domain.
    - **PRIVILEGED USER**: For SQL authentication, specify the username of the SQL Server login user. For Windows NT Integrated authentication, specify the Active Directory account name. The minimum role that this login requires is “public”.
    - **PASSWORD**: The password of the SQL Server Login, or the password of the Active Directory account
17. Click **VERIFY CONNECTION** to validate the connection from AAS to the Courion database hosted on the SQL server. If the verification succeeds, click **OK**.
18. If you specified the Courion database, the table and salt values will be detected automatically. If you are not using the Courion database, you must specify the table and columns which conform to the following sample:

```
CREATE TABLE [SaltValuesSample](
      [id_key] [bigint] IDENTITY(1,1) NOT NULL,
      [Table] [nvarchar](512) NOT NULL,
      [Column][nvarchar](512) NOT NULL,
      [Salt1] [nvarchar](24) NOT NULL,
      [Salt2] [nvarchar](24) NOT NULL,
      [Salt3] [nvarchar](24) NOT NULL,
      [Salt4] [nvarchar](24) NOT NULL,
      [Salt5] [nvarchar](24) NOT NULL
)
```

19. Click **OK**.The Specify Random Salt Configuration dialog box closes.
20. Click **NEXT**. The Site-specific Information dialog box closes; AAS creates a new encryption key, and automatically uses that key to re-encrypt all encrypted Core Server configuration information.

Platform Selection

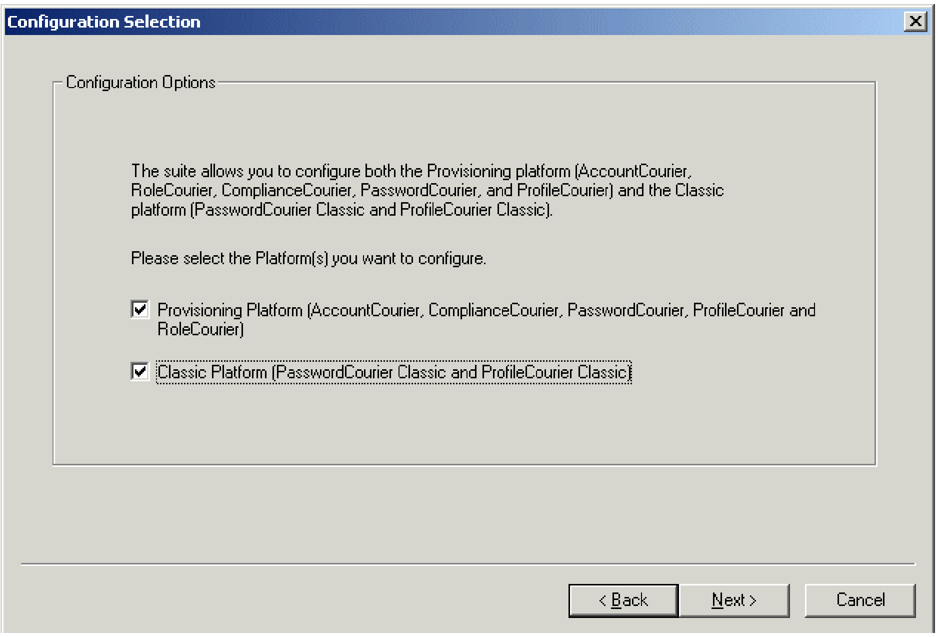
The Configuration Selection dialog box (as shown in *Figure 20*) prompts you to select one or both Access Assurance Suite platforms:

The **Provisioning Platform**, which includes the Core Provisioning, Core Compliance, Core Password, and ProfileCourier applications. You configure them using the Access Assurance Suite Administration Manager.

The **Classic Platform**, which includes Core Password and PasswordCourier Support Staff Classic and ProfileCourier Classic. You configure them using their own customization managers, explained in the manuals *Using Core Password and PasswordCourier Support Staff Classic* and *Using ProfileCourier Classic*. The classic platform lets you administer each application using its own customization manager.

Note: The classic platform is only available in a single server installation. This checkbox is only available for selection if you selected a Complete Installation, or selected Custom and then selected all four components.

Figure 20: Configuration Selection



Configuration Selection

Configuration Options

The suite allows you to configure both the Provisioning platform (AccountCourier, RoleCourier, ComplianceCourier, PasswordCourier, and ProfileCourier) and the Classic platform (PasswordCourier Classic and ProfileCourier Classic).

Please select the Platform(s) you want to configure.

☑ Provisioning Platform (AccountCourier, ComplianceCourier, PasswordCourier, ProfileCourier and RoleCourier)

☑ Classic Platform (PasswordCourier Classic and ProfileCourier Classic)

Select **Provisioning Platform** to install and configure the provisioning applications.

Select **Classic Platform** to install and configure the Classic versions of PasswordCourier and ProfileCourier.

Note: The choice you make — Provisioning, Classic, or both—determines which dialog boxes appear during the rest of the configuration process.

21. Choose either or both platforms and click **Next.**

    Note: For a new installation, select the provisioning platform because it provides more advanced features.

Express Configure Web Access and Install Baseline

From the Express Configuration Selection dialog box, shown in *Figure 21*, you can select **Express Configure Web Access** and **Install baseline**. These processes streamline installation and configure the sample workflow.

Note: Both Express Configure Web Access and Install Baseline are only available in a single server installation. This dialog box only appears if you selected a Complete installation, or selected Custom and then selected all components.

Express Configure Web Access streamlines the installation process but does not configure the sample workflow. This process resets the HTTPS key length to 0 (a possible security risk) and sets the session timeout period to 300 seconds. Later, you can change these settings, as explained in the “Web Access (ASP) Configuration” chapter in the *Access Assurance Suite Implementation Guide*.

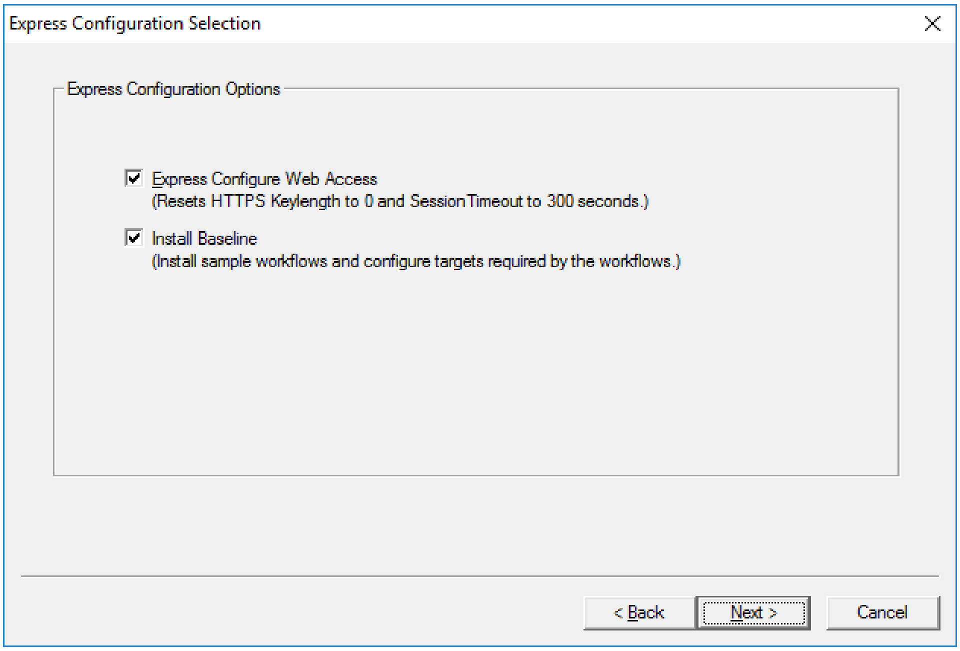
Install Baseline option installs default workflows and its associated connector targets.

The following workflows are installed as part of Baseline:

- **ProfileCreation** - Creates profile data for new user
- **ProfileManagement** – Manage profile of the user
- **PasswordReset_SelfService** – Used for resetting the password by end user
- **PasswordReset_SupportStaff** – Used for resetting the password of any account by support teams
- **XMLAOProvisioning** – Used for automated provisioning of the accounts
- **TransparentSynchronization** – Used for synchronizing the password of an account on different target
- **NonEmployeeExtend** – Used for changing the end date of the contractors/ Non-Employee.

22. By default both **Express Configure Web Access** and **Install Baseline** are selected. If either Express box is disabled, you see a message explaining which components are missing. Click **Next**.

Figure 21: Express Configuration Selection



Express Configuration Selection

Express Configuration Options

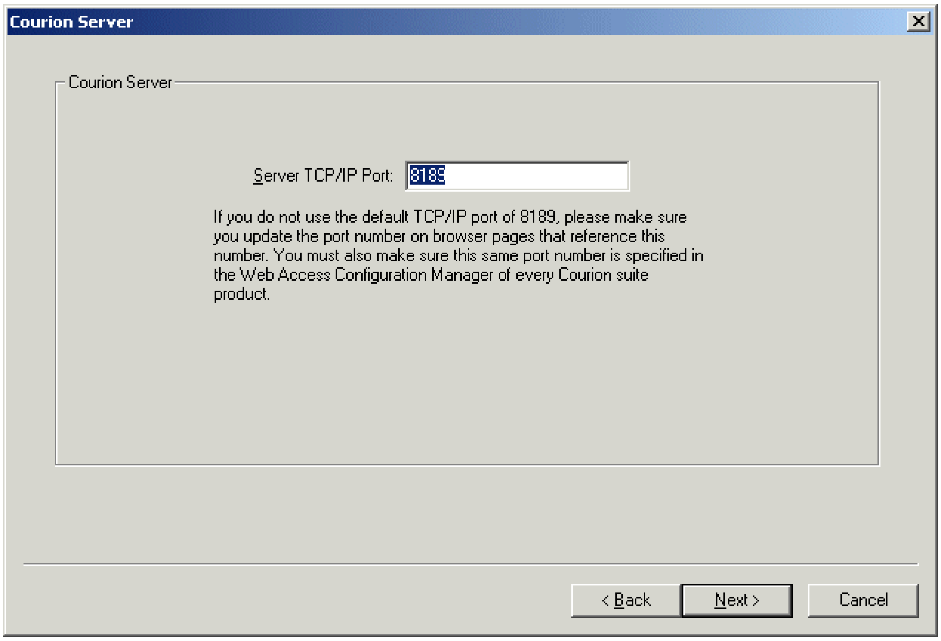
☑ Express Configure Web Access
(Resets HTTPS Keylength to 0 and SessionTimeout to 300 seconds.)

☑ Install Baseline
(Install sample workflows and configure targets required by the workflows.)

Courion Server TCP/IP Port

The configuration prompts for server TCP/IP information, *Figure 22*.

Figure 22: Courion Server TCP/IP



Server TCP/IP Port: 8189

If you do not use the default TCP/IP port of 8189, please make sure you update the port number on browser pages that reference this number. You must also make sure this same port number is specified in the Web Access Configuration Manager of every Courion suite product.

You can choose from the standard range of end user ports within the company's system. Consult the network administrator if you have questions about which port to use. The default value is 8189.

23. Specify the TCP/IP port to be used by the Core Server. Click the **NEXT** button.

If you change the port number from the default, then you must update any files installed within the Courion Corporation www folder, replacing the default value with your non-default value. Search for `8189` and update accordingly. Most but not all changes are to occurrences of `<PARAM name=PORT value="8189">`.

Additionally, you need to change the web access configuration options. There are five separate configurations that you need to change, depending on which platform you have installed. For the provisioning platform:

- Provisioning Platform End User, accessed from the following shortcut:

  Start>All Programs>Core Access Assurance Suite>Web Access Configuration

- Administration Manager Access, accessed from the following shortcut:

  Start>All Programs>Core Access Assurance Suite>Administration Manager Configuration

- Publisher Access, accessed from the following shortcut:

  Start>All Programs>Core Access Assurance Suite>Web Service Configuration Manager

For the classic platform:

- PasswordCourier Classic Access, accessed from the following shortcut:

  Start>All Programs>Core Access Assurance Suite>PasswordCourier ClassicWeb Access Configuration

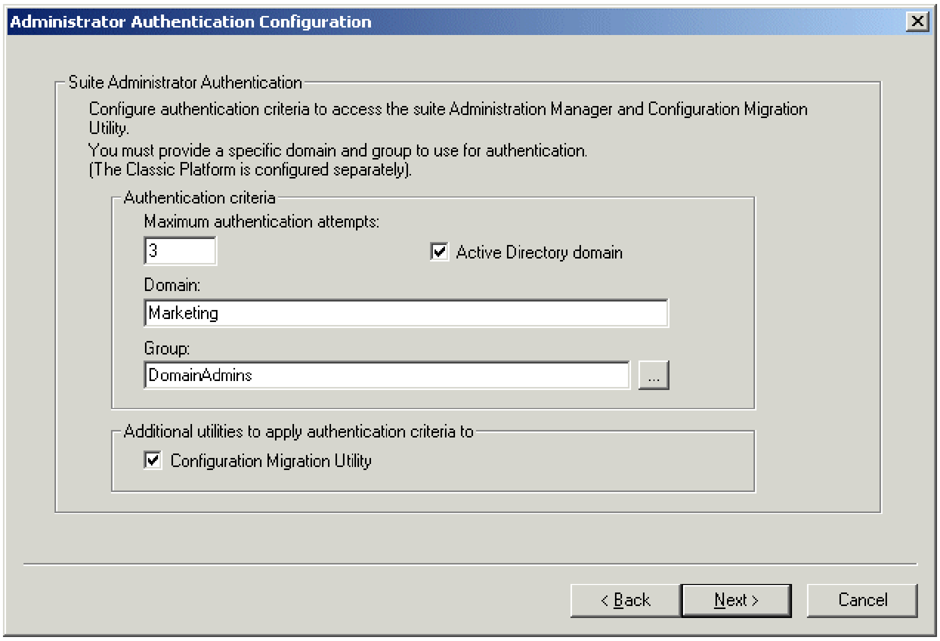
- ProfileCourier Classic Access, accessed from the following shortcut:

  Start>All Programs>Core Access Assurance Suite>ProfileCourier Classic>Web Access Configuration

Administrator Authentication

The Administrator Authentication dialog box appears, as shown in *Figure 23*.

Figure 23: Administrator Authentication Configuration



Administrator Authentication Configuration

Suite Administrator Authentication

Configure authentication criteria to access the suite Administration Manager and Configuration Migration Utility.

You must provide a specific domain and group to use for authentication.

(The Classic Platform is configured separately).

Authentication criteria

Maximum authentication attempts: 3

☑ Active Directory domain

Domain: Marketing

Group: DomainAdmins

Additional utilities to apply authentication criteria to

☑ Configuration Migration Utility

Use this dialog box to specify who can access the Access Assurance Suite Administration Manager interface.

Administrators can access the Administration Manager by authenticating to a domain or to the local workstation. The default values displayed are those of the domain that the server is registered in.

Domain access to the Administration Manager is controlled by domain and group membership. Anyone with a user account on the specified domain who is a member of the specified group can log in to the Administration Manager user interface. To restrict access, create a special group in an existing domain.

24. **AUTHENTICATION CRITERIA**

    The default criteria enable workstation-based access. If you want to change the values shown, follow these steps:

    a. **MAXIMUM AUTHENTICATION ATTEMPTS** — Enter the maximum number of times the administrator can attempt authentication. The administrator must then close the window and begin a new session to attempt authentication again. The default is 3.

    b. **ACTIVE DIRECTORY DOMAIN** — If this is an Active Directory domain, mark the check box.

    c. **DOMAIN** — Enter two backslashes and the name of the local workstation on which the administrator has an account (for example: `\\SusanA_workstation`).

d. **GROUP** — Click the browse button and select the name of the group in which the administrator has membership. To restrict access, you can create and use a special group for provisioning platform administrators.

25. **ADDITIONAL UTILITIES TO APPLY AUTHENTICATION CRITERIA TO**

a. **CONFIGURATION MIGRATION UTILITY** — The Configuration Migration Utility enables you to transfer configurations from one server to another. It can speed up the task of configuring sites with many servers, but under some circumstances you may not want the administrator to use it.

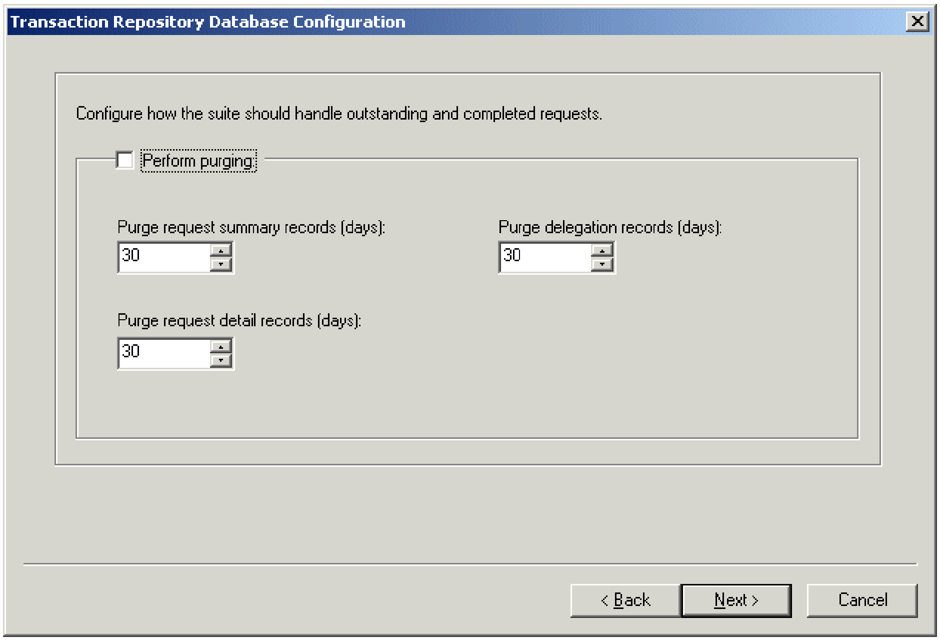
Leave the box checked or uncheck it. It is checked by default.

Click **NEXT.**

Transaction Repository Database Configuration

The Transaction Repository Database Configuration dialog box appears as shown in *Figure 24*.

Figure 24: Transaction Repository Database Configuration



Transaction Repository Database Configuration

Configure how the suite should handle outstanding and completed requests.

□ Perform purging

Purge request summary records (days): 30

Purge delegation records (days): 30

Purge request detail records (days): 30

The transaction repository database stores all request records, delegation records, and verification information. If you want to use the Requester/Approver or Delegation features, the ComplianceCourier Verify feature, the Password History feature, or if you need to store compliance information, then you must configure the Connector for Microsoft ADO with a transaction repository.

Note: Do not use for new installations.

26. **PERFORM PURGING** — Select this option for purging, which prevents the transaction repository from growing without limits. If your organization has data backup and data retention policies, Core Security recommends that you use those polices rather than the purging option. Purging is not the default (the **PERFORM PURGING** option is not checked).

The purging option applies to Request summary, Request detail, and Delegation data in the transaction repository. The system does not purge role definition information that exists as a result of using the Core Role Connector to create or manage roles.

Note: If you choose the **PERFORM PURGING** option, be aware that the system does not try to back up or preserve the data that is deleted during a purge. You must provide a method for backing up data you want to keep before it is purged.

You can select a different number of days between purges for each record type:

a. **PURGE REQUEST SUMMARY RECORDS (DAYS)** — Specifies the interval of days between purges of request summary records.

b. **PURGE DELEGATION RECORDS (DAYS)** — Specifies the interval of days between purges of delegation records.

c. **PURGE DETAIL RECORDS (DAYS)** — Specifies the interval of days between purges of detail records.

The default frequency for **PURGE REQUEST SUMMARY RECORDS** and **PURGE REQUEST DETAIL RECORDS** is 30 days. If you choose to perform purging, setting the **PURGE REQUEST DETAIL RECORDS** interval higher than the **PURGE REQUEST SUMMARY RECORDS** interval ensures that the transaction repository database maintains sufficient space to store all request and delegation records. The request summary records provide enough information for reports and logging.

Additional information on purging:

- A stored procedure is used to do the purging.
- The polling engine of the Core Server checks hourly to see if purging is configured and if there are records to be purged.
- The polling engine runs on low priority thread in the Core Server to not interrupt the main functionality of the product.
- Only closed/completed records are purged after the configured number of days. Any unapproved/unprocessed requests are not purged from the details or summary related tables. Any delegation records where delegation period is still in effect are not purged from delegation related tables.
- If doing reporting against the transaction repository, the request summary records contain the most useful data.
- Purging the transaction repository does not purge data in the role repository (used by RoleCourier).
- There is no built in database archiving of the transaction repository before the purge occurs. You should archive the data, if the records to be purged may be needed in the future.
- If the value of 0 days is specified, then any new items that match the qualification for purging are purged at the next purging opportunity. The "0" value indicates that they do not even have to be around for 1 day.

For more information on the transaction repository, see the manual *Configuring Workflows with the Access Assurance Suite Administration Manager*.

Click **NEXT.**

The dialog that is displayed next depends on the type of installation and the options you selected.

If you have chosen a Custom installation and selected only the Courion Server option, the Connector Framework Manager Web Service Instance dialog appears next.

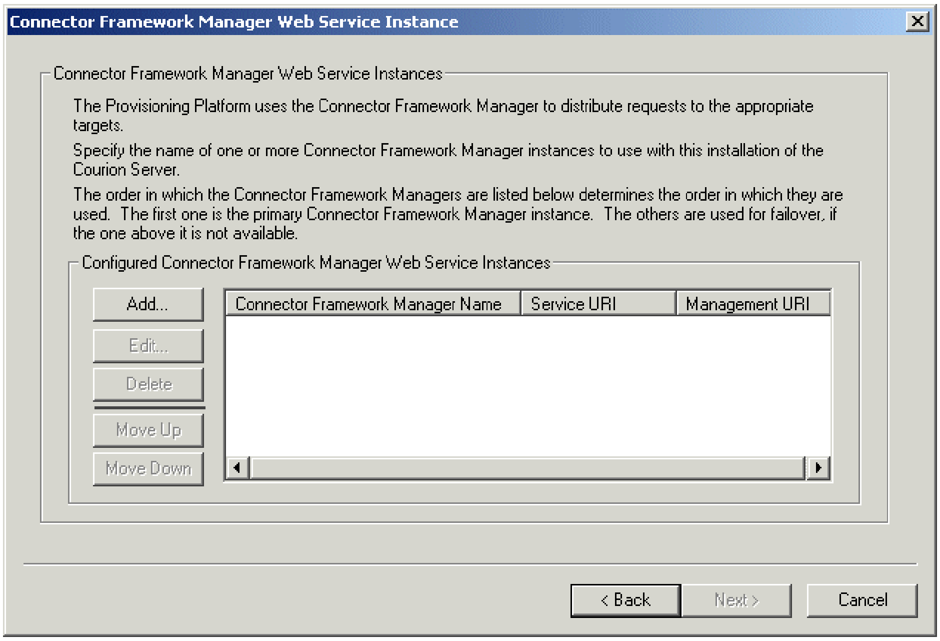
If you selected both platforms or only the classic platform on the Platform Selection dialog, a series of dialogs allow you to select profile and ticketing data sources.

If none of the above applies, the system displays a summary dialog box that lists all data you entered (see *"Summary" on page 64*).

Connector Framework Manager Web Service Instance

If you have installed the Core Server without also installing a Connector Framework Manager (CFM) web service on the same system, this dialog appears, allowing you to connect the Core Server with one or more CFM instances.

Figure 25: Connector Framework Manager Web Service Instance Configuration



Connector Framework Manager Web Service Instance

Connector Framework Manager Web Service Instances

The Provisioning Platform uses the Connector Framework Manager to distribute requests to the appropriate targets.

Specify the name of one or more Connector Framework Manager instances to use with this installation of the Courion Server.

The order in which the Connector Framework Managers are listed below determines the order in which they are used. The first one is the primary Connector Framework Manager instance. The others are used for failover, if the one above it is not available.

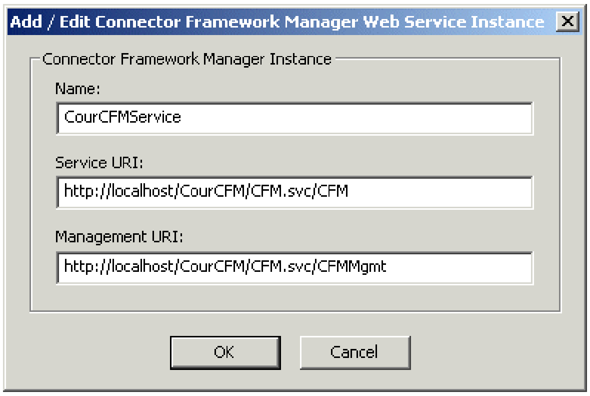
Configured Connector Framework Manager Web Service Instances

Connector Framework Manager Name	Service URI	Management URI

27. Click the **Add** button to open the Add Connector Framework Manager Instance dialog. To change the settings for an existing CFM instance, click the Connector Framework Manager Name, then click **Edit**. To remove a CFM instance from the list, click the Connector Framework Manager Name, then click **Delete**.

    If you have more than one CFM, the order in which they are listed determines the order in which they are used. Click the Connector Framework Manager name and then click the **Move Up** or **Move Down** buttons to set your desired order.

Figure 26: Add/Edit a Connector Framework Manager Web Service Instance



Connector Framework Manager Instance

Name: CourCFMService

Service URI: http://localhost/CourCFM/CFM.svc/CFM

Management URI: http://localhost/CourCFM/CFM.svc/CFMMgmt

28. Select a **Name**, **Service URI**, and **Management URI** for this Connector Framework Manager Instance. These fields are already filled in with default values. The default URIs are based on the server that the Core Server is installed on. This allows you to change the server name to the name for the desired CFM and leave the rest of the path name set to default values.

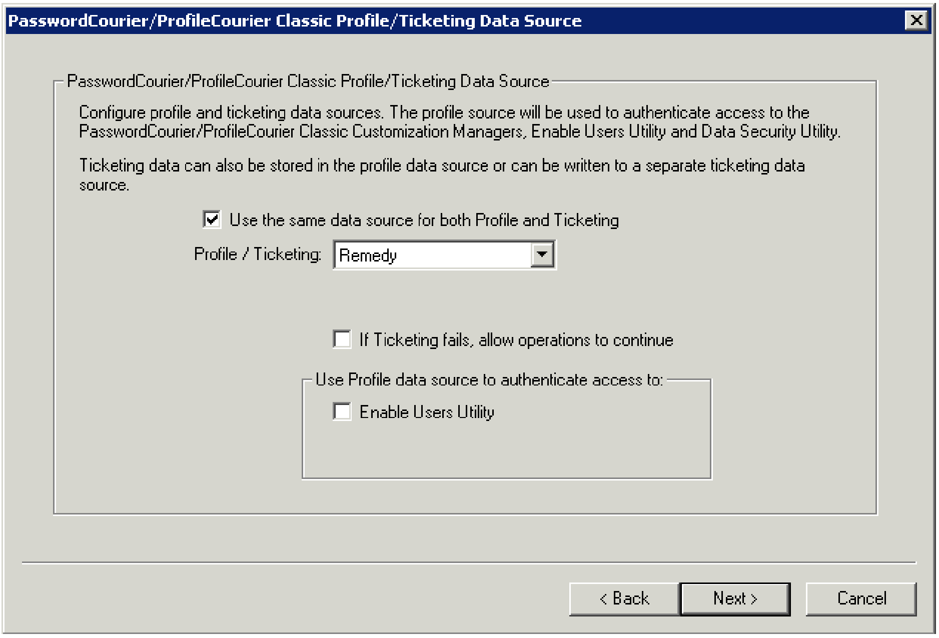
    Click **OK** to return to the Connector Framework Manager Web Service Instance dialog. When finished configuring CFM instances, click **Next**.

Selecting a Data Source

Note: This section applies to the classic platform only.

During PasswordCourier Classic configuration, if you have one or more access keys to data sources, the data source selection dialog box appears, shown in *Figure 27*. If the data source dialog box does not appear, skip to *“Configuring SMTP Email” on page 62*.

Figure 27: Ticketing Data Source Selection



PasswordCourier/ProfileCourier Classic Profile/Ticketing Data Source

Configure profile and ticketing data sources. The profile source will be used to authenticate access to the PasswordCourier/ProfileCourier Classic Customization Managers, Enable Users Utility and Data Security Utility.

Ticketing data can also be stored in the profile data source or can be written to a separate ticketing data source.

☑ Use the same data source for both Profile and Ticketing

Profile / Ticketing: Remedy

☐ If Ticketing fails, allow operations to continue

Use Profile data source to authenticate access to:

☐ Enable Users Utility

29. **Data Source Selection** — A data source is a database used by the Access Assurance Suite. Complete the data source fields as follows.

    a. **Use the same data source for both Profile and Ticketing**

    PasswordCourier Classic and ProfileCourier Classic can use the same data source or separate sources for profile information and ticketing.

    The Profile data source validates PasswordCourier Classic and ProfileCourier Classic end users when they attempt to use a Core Security application.

    The Ticketing data source helps create or update tickets or audit information about any actions taken. AccountCourier and ComplianceCourier use a ticketing connector for integration with a Help Desk application.

    Select the same data source for profile and ticketing by marking the check box; or to use different data sources, leave it unchecked.

    b. **Profile/Ticketing**

    Specify the data source from the pull-down list; for example, ODBC or LDAP. For LDAP, when configuring ticketing, be sure to select field values are selected from the drop down list even if the value to be used is displayed by default.

    c. **If Ticketing fails, allow operations to continue**

    If you select this option, and later the ticketing source is unavailable, PasswordCourier Classic and ProfileCourier Classic allow password reset or profile modification operations. The Core Server logs the failed ticket to a comma separated variable (CSV) file called `failed_tickets.csv`.

    To let password/profile/operations continue without access to a ticketing source, mark this checkbox; or to have these operations terminate, leave it blank.

    d. **Use Profile data source to authenticate access to:**

    This selection governs access to the Enable Users Utility. If you leave the box unchecked, an end user can log into and use that utility without supplying a username or password.

    The Enable Users Utility can re-enable the profile of an end user whose profile has been disabled because he or she has failed the specified number of login attempts. The Enable Users Utility provides the only way for a disabled end user to be re-enabled.

    For more information about the Enable Users Utility, refer to the manual *Using the Access Assurance Suite Administration Manager Utilities*.

30. Click **Next**.

    If you selected ODBC as a data source, the Configuration Manager prompts you for more ODBC information, as shown in the next section.

    If you selected LDAP as a data source, the Configuration Manager prompts you for more information, as described in *"Configuring an LDAP (Lightweight Directory Access Protocol) Data Source" on page 58*.

    If you selected Remedy, which has Help Desk ticketing as a data source, you need to provide the Core Server with additional information. See , then return here.

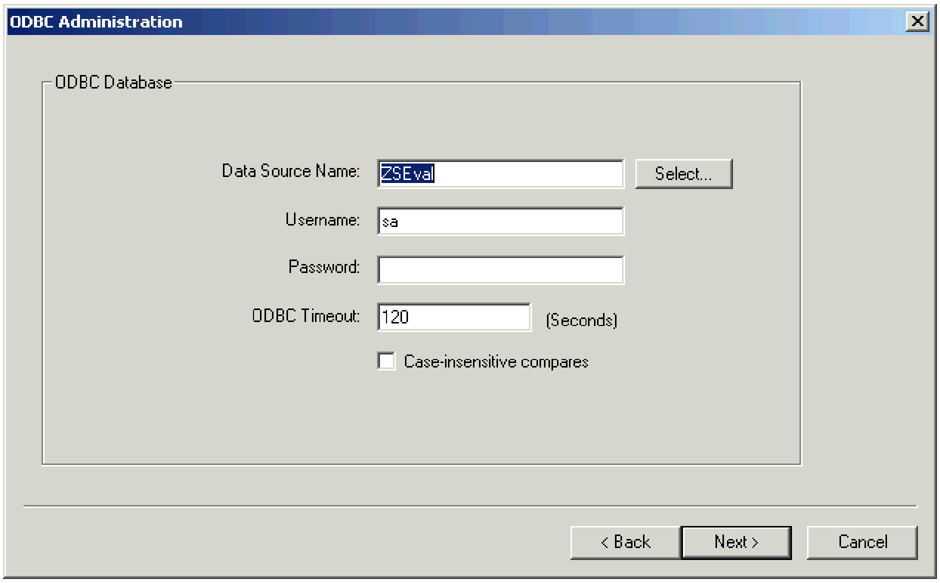
    For other data sources, see the appropriate configuration sections later in this chapter. Then return here. The configuration sections are:

Configuring an ODBC Data Source

Note: This section applies to the classic platform only.

If you specified ODBC as a data source, the ODBC Administration dialog box appears, as in *Figure 28*.

Figure 28: ODBC Database Configuration



ODBC Administration

ODBC Database

Data Source Name: ZSEval

Username: sa

Password:

ODBC Timeout: 120 (Seconds)

□ Case-insensitive compares

31. To configure the Access Assurance Suite to work with the ODBC system:

    a. **Data Source Name** — Specify an ODBC Data Source name. If the name is unknown, click the **Select** button to display a list of data sources configured on the local system.

    b. **Username** — Enter a username for access to the data source.

    c. **Password** — Enter a password for access to the data source.

       When providing a password to allow PasswordCourier Classic to access the ODBC database, make sure the password specified here is consistent with the passwords that may be specified on the database as well as the ODBC

       DSN. If the database is password protected, the Windows ODBC data source must also contain a valid user name and password. You can specify the DSN username and password in the Advanced settings of the OBDC Data Source definition dialog.

    d. **ODBC Timeout** — Enter the time-out value for PasswordCourier when making requests to the ODBC database. The default value is 120 seconds.

    e. **Case-sensitive compares** — Determine whether or not to use case sensitive comparisons for user validation and authentication. Mark the box to use case-insensitive comparisons. This allows the user to enter “smith” or “SMITH” for successful validation of the value “Smith” which is stored in the database. If this box is not checked, then the case-sensitivity of the underlying database is used for the user validation and authentication comparisons. To be validated/ authenticated, the user would be required to enter “Smith” exactly as it appears in the database.

       Microsoft Access does not support this functionality, so the check box is grayed out.

       Microsoft SQL Server is case-insensitive by default.

Sybase® case-sensitivity is determined during installation and configuration of language and character set of the Sybase Server®.

Oracle and IBM DB2® are case-sensitive regardless of platform.

f. Click the **NEXT** button to proceed. You are asked if you want to use Foreign Key Constraints. If you answer yes, see *“Manual Foreign Key Configuration” on page 60*. If you answer no, the next dialog box is SMTP Email Configuration. See *“SMTP Email Configuration” on page 62*.

ODBC Notes and Warnings

Fields in an Oracle database that are defined as required are not properly returned by the ODBC driver as being required. As a result, required fields appear as optional.

Oracle8 ODBC drivers against Oracle7.3.4 and Oracle 8.0.5 databases generally provide more accurate required field information

Fields in a Microsoft Access database that are defined as required are not properly returned by the ODBC driver as being required. As a result, required fields appear as optional.

Microsoft Access ODBC 16- and 32-bit drivers cannot handle tables with more than 40 fields.

If the Core Server is installed on a machine with Microsoft SQL Server, Oracle, or another database that uses Oracle, change the database services used at Windows startup before loading the Core Server or the Core Server may fail to load. In some cases, the Core Server may still fail to load because the database services have not finished loading. In this case, manually restart the Core Server from the Services Control Panel applet after the database services have fully initialized.

Avoid using hyphens or spaces in Microsoft SQL Server table names as they may cause an error message to be returned.

Configuring an LDAP (Lightweight Directory Access Protocol) Data Source

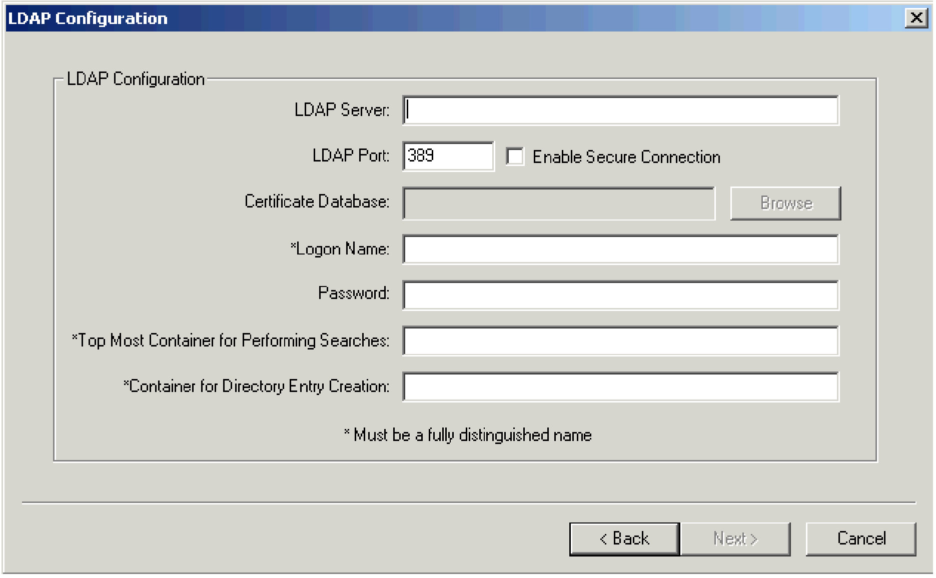
Note: This section applies to the classic platform only.

LDAP configuration allows user authentication and the creation, updating, and closing of tickets in an LDAP directory.

To configure Access Assurance Suite with an LDAP directory, you must direct the Core Server to use LDAP as a profile data source, a ticketing data source, or both.

If you selected LDAP as the data source, the Configuration Manager displays the following dialog box.

Figure 29: LDAP Configuration



LDAP Configuration

LDAP Configuration

LDAP Server:

LDAP Port: 389 □ Enable Secure Connection

Certificate Database:

*Logon Name:

Password:

*Top Most Container for Performing Searches:

*Container for Directory Entry Creation:

* Must be a fully distinguished name

32. To configure an LDAP data source, follow these steps. A sample completed dialog box follows in *Figure 30*.

    a. **LDAP Server** — Enter the LDAP server name.

    b. **LDAP Port:** Enter the TCP/IP port number used by the server to communicate with the Core Server. The default is 389 for a nonsecure connection or 636 for an SSL connection.

       Note: If the SSL option is not enabled on the LDAP server, connection to the server must use a nonsecure port with port number 389.

    c. **Certificate Database** — If the connection to the LDAP server uses SSL, specify the Certificate Database that contains the certificate for the Certificate Authority (CA) that issued the server's certificate. You can use the Browse button to find the database. If a public CA is not being used, see

    d. **Enable Secure Connection** — Select this checkbox to enable an SSL connection to the LDAP Server.

    e. **Logon Name:** Enter a username that the Core Access Assurance Suite can use to access the specified LDAP server directory. This account must have privileges to access the LDAP entries that the Core application uses. The name must be a fully distinguished name. Use the format “cn=name,” for example, “cn=Directory Manager.”

    f. **Password** — Enter the password for the account to access the LDAP directory.

    g. **Top Most Container for Performing Searches** — Enter the base Distinguished Name (search base) of the container from which searches in the tree with begin. You can specify multiple search base trees by using a semicolon to separate the Distinguished Names. Use the format “o=name,” for example, “o=corp.courion.com.”

       Enter the container that is used during searches if no container is specified in the native query (must be a fully distinguished name).

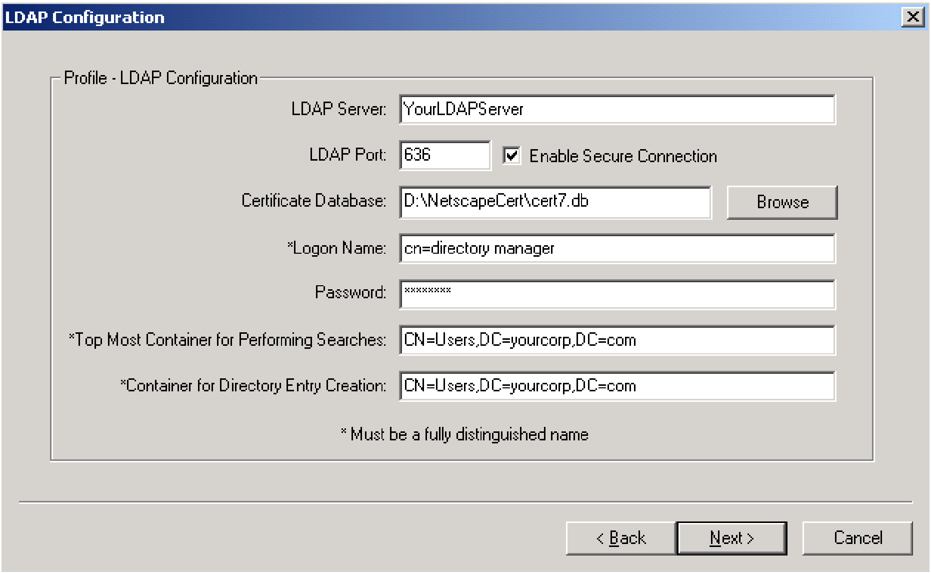
    h. **Container for Directory Entry Creation** — Enter the fully Distinguished Name of a container for the Core Access Assurance Suite to use when creating new LDAP entries (profiles, for example).

33. When you are satisfied with your responses, click the **Next** button to complete LDAP configuration.

    Note: The software tries to verify these entries by binding to the specified LDAP server using the specified username and password. If the entries cannot be verified, configuration cannot continue.

    Certain directory servers do not publish their schema information via LDAP (for example, Microsoft Metadirectory Server®). To support these servers, the Core Server allows specifying a schema by a file. To enable this feature, please contact Core Security Support at support@coresecurity.com.

Figure 30: Sample LDAP Configuration, Completed



LDAP Configuration

Profile - LDAP Configuration

LDAP Server: YourLDAPServer

LDAP Port: 636 ☑ Enable Secure Connection

Certificate Database: D:\NetscapeCert\cert7.db

*Logon Name: cn=directory manager

Password: ********

*Top Most Container for Performing Searches: CN=Users,DC=yourcorp,DC=com

*Container for Directory Entry Creation: CN=Users,DC=yourcorp,DC=com

* Must be a fully distinguished name

LDAP Notes and Warnings

Core Security strongly recommends installing a x.509 server certificate onto the LDAP server and configuring the Core Server to use a secure connection.

All LDAP fields have the display restriction in the Core Access Assurance Suite of a 64-character text field.

If a select statement is built selecting a field that has more than one value in the LDAP entry, the value selected can be of any of the values found for that field.

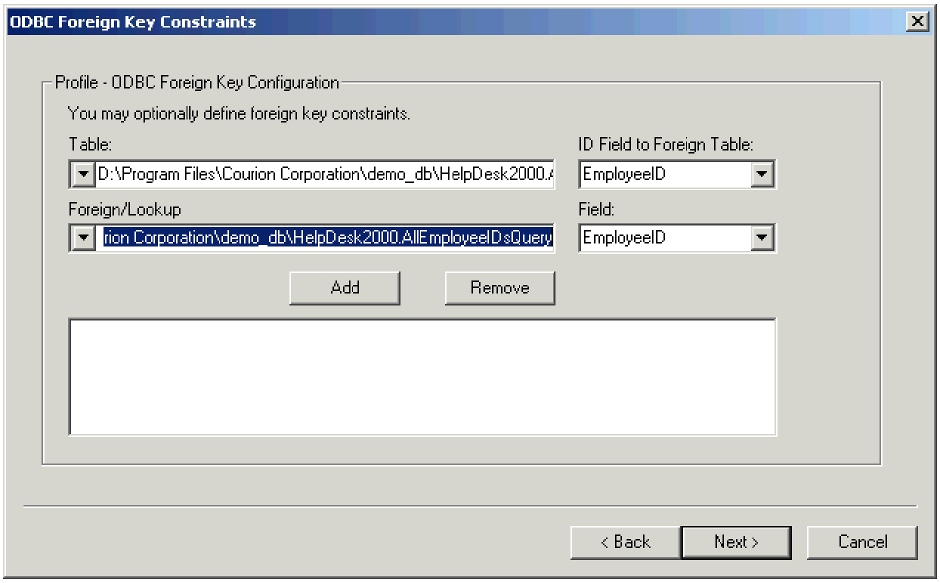
Continue to *“Configuring SMTP Email” on page 62*.

Manual Foreign Key Configuration

Note: This section applies to the classic platform only.

A foreign key is a set of one or more columns in a table which may hold the values found in the primary key columns of another table. The ODBC Foreign Key Constraints dialog box provides an opportunity to configure foreign key constraints that are not implemented as true foreign key columns in the underlying database. The Core Server would otherwise treat the columns as normal database columns. This dialog box appears in *Figure 31*.

Figure 31: ODBC Foreign Key Constraints



ODBC Foreign Key Constraints
Profile - ODBC Foreign Key Configuration
Add
Remove
< Back
Next >
Cancel

You may optionally define foreign key constraints.

Table: D:\Program Files\Courion Corporation\demo_db\HelpDesk2000.A

ID Field to Foreign Table: EmployeeID

Foreign/Lookup: rion Corporation\demo_db\HelpDesk2000.AllEmployeeIDsQuery

Field: EmployeeID

Some database designers may implement foreign key constraints by using database triggers. The Core Server recognizes and handles only true foreign key columns that are displayed in drop-down lists in Core Security products. This configuration dialog box enables administrators to define foreign keys to the Core Server that are implemented using other techniques.

Note: True database foreign key fields cannot be defined in this panel.

For example, in a Help Desk table there is a column named **CALL_STATUS**. It is limited to values such as Critical, High, Medium, and Low. The values set for Critical, High, Medium, and Low are defined in a second table named StatusValues. The **STATUSVALUES** table may contain columns such as **STATUS_ID** and **STATUS_NAME**. This constraint may be configured by defining it as a foreign key. The Help Desk **CALL_STATUS** column is a foreign key to **STATUSVALUES.STATUS_NAME**.

34. To skip foreign key configuration, click **NO** and for a standard configuration skip to the next section. For an Express configuration skip to *“Configuring SMTP Email” on page 62*.

    To configure constraints now, click **YES** and continue in this section.

35. To configure the foreign keys:

    a. **TABLE** — Select the table and ID field column from the top set of choice boxes that correspond to the foreign key field.

    b. **FOREIGN/LOOKUP** — Select the table and ID field column from the bottom set of choice boxes that correspond to the referenced primary key.

    c. Click the **ADD** button.

       You can use the **REMOVE** button to remove foreign key constraints that have been configured.

    d. Click **NEXT** when the foreign keys have been defined.

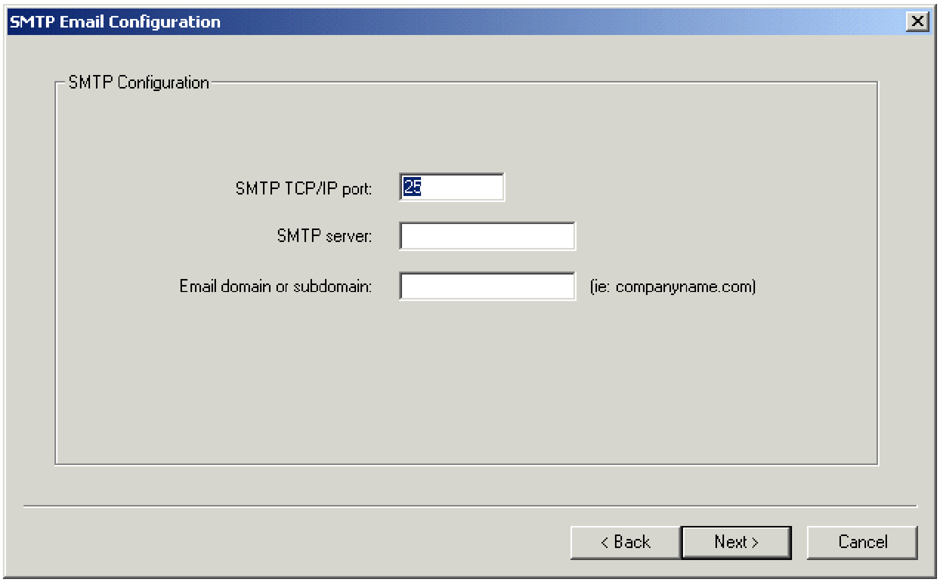
    For an Express configuration skip to *“Configuring SMTP Email” on page 62*.

Configuring SMTP Email

Note: This section applies to the classic platform only.

If your software includes a key for PasswordCourier Classic, the Configuration Manager prompts you to configure SMTP Email, as in *Figure 32*.

Figure 32: SMTP Email Configuration



SMTP Email Configuration

SMTP Configuration

SMTP TCP/IP port: 25

SMTP server:

Email domain or subdomain: (ie: companyname.com)

The Access Assurance Suite uses an SMTP connector to enable e-mail notification. See *Configuring Password Management Modules (PMMs), Connectors, and Agents* for more information on how to configure this connector.

36. To enable the SMTP e-mail messaging in the Core Server, follow these steps. If you don't know information required to configure the SMTP Server at this time, leave the field blank and continue to the next dialog box.

    a. **SMTP TCP/IP PORT** — Specify the TCP/IP port for the SMTP server. The default setting is 25.

    b. **SMTP SERVER** — Specify the network-resolvable name of the system running the SMTP server.

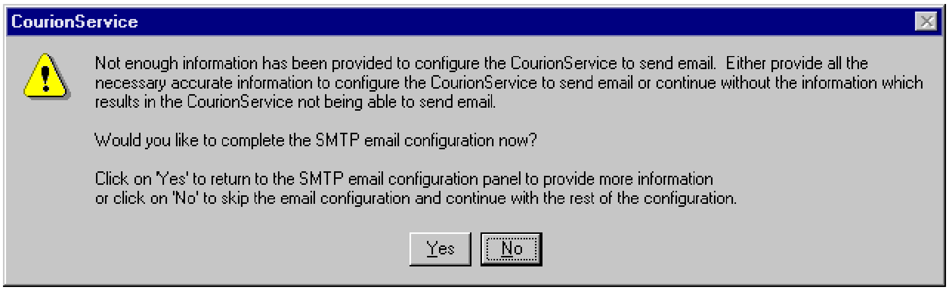
    c. **EMAIL DOMAIN OF SUBDOMAIN** — Specify the name of the domain or subdomain managed by the SMTP server.

       The domain or subdomain name of the SMTP service depends on the configuration of the SMTP server. While the Access Assurance Suite accepts any entry for the e-mail domain or subdomain, this domain name must be the corporate domain if the SMTP server is configured to prohibit relaying. SMTP/ mail errors are recorded in the smtpemail.log found in the Core Server folder. Higher level logging is available. For additional information, contact support@coresecurity.com.

37. Click the **NEXT** button.

If any fields are left blank, the following message box appears:

Figure 33: SMTP Configuration Incomplete Message



CourionService

Not enough information has been provided to configure the CourionService to send email. Either provide all the necessary accurate information to configure the CourionService to send email or continue without the information which results in the CourionService not being able to send email.

Would you like to complete the SMTP email configuration now?

Click on 'Yes' to return to the SMTP email configuration panel to provide more information or click on 'No' to skip the email configuration and continue with the rest of the configuration.

- Click the **YES** button to bypass SMTP configuration. Click the **NO** button to return to the SMTP Configuration dialog box.

To configure the SMTP server later, you can select Start>Programs>Core Access Assurance Suite>Core Server>Configuration Manager

For more information on configuring the SMTP connector, see the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents*.

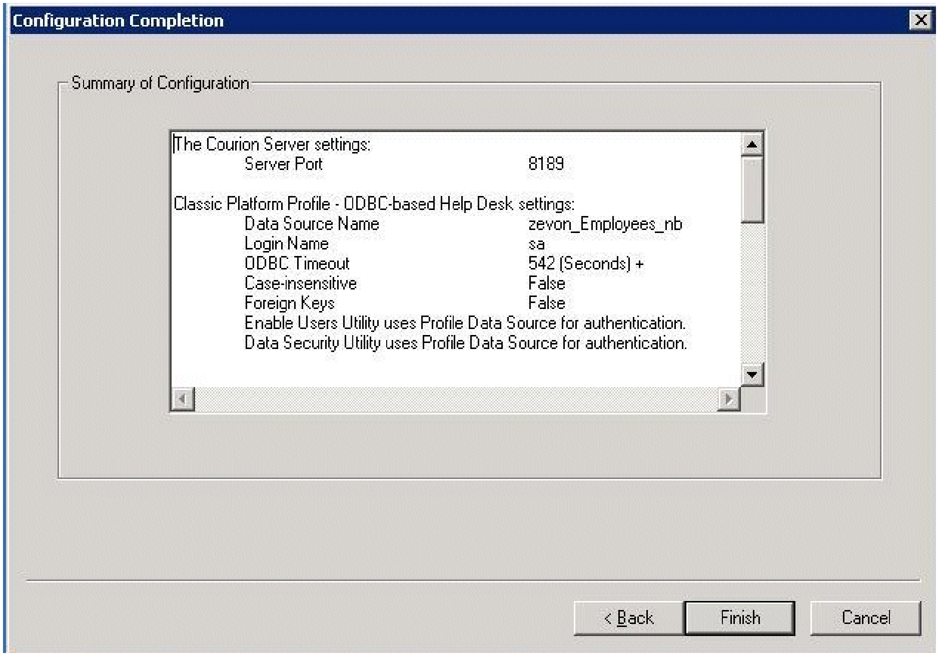
For instructions on how to configure e-mail as a compensating control (security tickets) or user notification, please refer to the appropriate guide: *Using PasswordCourier and PasswordCourier Support Staff Classic*, *Using ProfileCourier Classic*, or *Configuring Workflows with the Access Assurance Suite Administration Manager* for applications on the provisioning platform.

For information on customizing the standard SMTP email (such as specifying a nonstandard line length, or a special end-of-line indicator character like comma or parenthesis), consult Core Security Customer Support.

The system now displays a summary dialog box that lists all information you have entered (*Figure 34*).

Summary

Figure 34: Configuration Summary



```
The Courion Server settings:
          Server Port                          8189

Classic Platform Profile - ODBC-based Help Desk settings:
          Data Source Name                     zevon_Employees_nb
          Login Name                           sa
          ODBC Timeout                         542 (Seconds) +
          Case-insensitive                     False
          Foreign Keys                         False
          Enable Users Utility uses Profile Data Source for authentication.
          Data Security Utility uses Profile Data Source for authentication.
```

38. Click the **FINISH** button to accept the current configuration of the Core Server. If you are changing an existing configuration, it asks if you want to stop the Courion Service now. Click **YES**.

    The software displays a message that new parameters do not take effect until the service is started.

39. To start the service, click **YES**. If you want to specify other parameters, click **NO**.

    The Configuration Manager first asks about configuring SSL, then Password Management Modules PMMs, and then connectors. Configuration of PMMs and connectors is dependent upon the access keys you have.

SSL Configuration Utility

The SSL Configuration utility program enables the setup of server certificates for use with the Core Server.

Note: The SSL Configuration Utility does not appear if you entered an evaluator access key. You can access this utility using program shortcuts.

40. Answer **YES** to configure the SSL Utility and see the manual *Using the Access Assurance Suite Administration Manager Utilities*. To skip SSL configuration, click **NO**.

    The system prompts you to configure all Password Management Modules for which you entered access keys.

Password Management Module Configuration

If you have an access key for PasswordCourier, the Configuration Manager prompts you to configure Password Management Modules (PMMs). Each PMM is an interface that PasswordCourier requires to work with a specific target. PMMs are described in the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents*.

If you are configuring the Core Server for the first time, the connector manager asks about various types of PMM in sequence.

41. For each PMM you want to configure click **YES.** Go to the appropriate manual named above, find the PMM, and configure it. To configure PMMs later, click **NO** and skip the next step.

    If you are changing an existing configuration, the Configuration Manager asks if you want to update any Password Management Module configurations.

42. If you want to change a PMM configuration, click the **YES** button; you are prompted about various PMMs. For each PMM to reconfigure click **YES.** Go to the appropriate manual named above, find the PMM and reconfigure it. If you don't want to reconfigure PMMs, click **NO**.

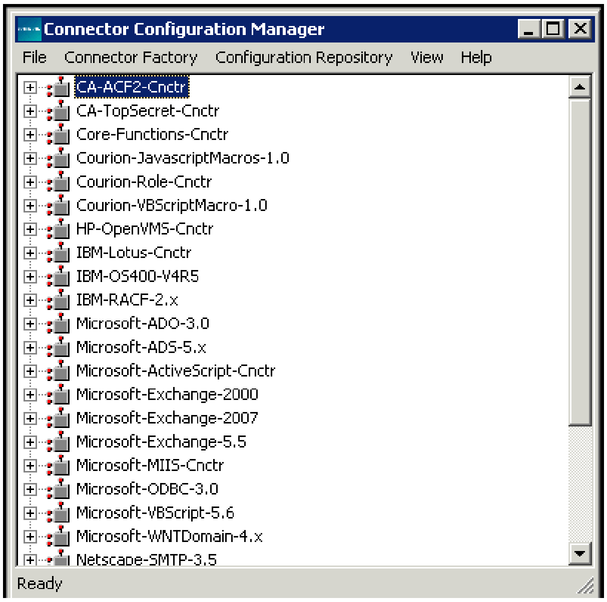
Connector Configuration

During any configuration, if your software includes pertinent access keys, the system prompts you about configuring connectors. Connectors are interfaces that let applications work with other vendors' platforms.

43. If you want to configure connectors now, click the **YES** button. If you want to configure them later, click **NO**.

    If you answer Yes, the software displays a list of connectors you can add and configure, as in *Figure 35*.

Figure 35: List of Connectors to Add and Configure



CA-ACF2-Cnctr
CA-TopSecret-Cnctr
Core-Functions-Cnctr
Courion-JavascriptMacros-1.0
Courion-Role-Cnctr
Courion-VBScriptMacro-1.0
HP-OpenVMS-Cnctr
IBM-Lotus-Cnctr
IBM-OS400-V4R5
IBM-RACF-2.x
Microsoft-ADO-3.0
Microsoft-ADS-5.x
Microsoft-ActiveScript-Cnctr
Microsoft-Exchange-2000
Microsoft-Exchange-2007
Microsoft-Exchange-5.5
Microsoft-MIIS-Cnctr
Microsoft-ODBC-3.0
Microsoft-VBScript-5.6
Microsoft-WNTDomain-4.x
Netscape-SMTP-3.5

You can add and configure connectors as described in the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents*.

For any configuration without Express Connector Config, the connector query completes server configuration and the following prompt appears:

```
Core Server configuration is complete
```

44. Click **OK**.

The Core Server application is now configured and started. It is configured to start automatically when the system boots.

If you need to change the configuration at a later time, go to:

Start>Programs>Core Access Assurance Suite>Configuration Manager

Now you can configure any additional connectors, PMMs, and workflows as described in the following manuals:

- For connectors and PMMs, *Configuring Password Management Modules (PMMs), Connectors, and Agents*.
- For workflow configuration, *Configuring Workflows with the Access Assurance Suite Administration Manager*.
- For Core Password, *Using PasswordCourier and PasswordCourier Support Staff Classic*.

Additional Configuration for the Access Assurance Portal Components

The following procedures are required as part of configuring the Access Assurance Portal. The Access Request Manager Solution, Identity Mapping Solution, and Access Certification Solution require these changes.

This section does not apply if you are implementing the provisioning platform or the classic platform as described in *“Core Access Assurance Suite Platforms and Applications” on page 12*.

Editing Connection Strings

This section describes how to point the Access Assurance Suite to the database.

1. In a text editor, open the [courion-installation-folder]\CourionARMS\CustomerConnStrings.config file for editing.
2. Edit the MetricRepositoryDefault, Default, dbConnectionString, and ARMEntities connection strings which appear below. The connection strings should be uncommented. Point the connection strings to the correct database by replacing $$YOURSERVERHERE$$ with the name of the database server, and $$YOURDBHERE$$ with the name of the Courion database.

```
<add name="MetricRepositoryDefault"
connectionString="Data Source=$$YOURSERVERHERE$$;Initial
Catalog=$$YOURDBHERE$$;Trusted_Connection=True"
providerName="System.Data.SqlClient" />

<add name="Default" connectionString="Data
Source=$$YOURSERVERHERE$$;Initial
Catalog=$$YOURDBHERE$$;Trusted_Connection=True"
providerName="System.Data.SqlClient" />

<add name="dbConnectionString" connectionString="Data
Source=$$YOURSERVERHERE$$;Initial
Catalog=$$YOURDBHERE$$;Trusted_Connection=True"
providerName="System.Data.SqlClient" />

<add name="ARMEntities" connectionString="metadata=res://
*/;provider=System.Data.SqlClient;provider connection
string=&quot;Data Source=$$YOURSERVERHERE$$;Initial
Catalog=$$YOURDBHERE$$;Trusted_Connection=True;MultipleAc
tiveResultSets=True&quot;"
providerName="System.Data.EntityClient" />

<add name="AccessEntities"
connectionString="metadata=res://*/AccessModel.csdl|res:/
/*/AccessModel.ssdl|res://*/
AccessModel.msl;provider=System.Data.SqlClient;provider
connection string=&quot;Data
Source=$$YOURSERVERHERE$$;Initial
Catalog=$$YOURDBHERE$$;Integrated Security=false;User
```

```
ID=$$YOURUSERIDHERE$$;Password=$$YOURDBPASSWORDHERE$$;mul
tipleactiveresultsets=True;App=EntityFramework&quot;"
providerName="System.Data.EntityClient" />
```

3. Save your changes and close the file.
4. To prevent passwords from being passed in clear text, you can encrypt the CustomerConnStrings.config file. In a command window, encrypt the connection strings using the following commands:

```
cd C:\Windows\Microsoft.Net\Framework\v4.0.30319

aspnet_regiis.exe -pef connectionStrings "C:\ProgramFiles (x86)\Courion
Corporation\CourionARMS\CustomerConnStrings.config"
```

5. In a text editor, open the CustomerConnStrings.config file to confirm that it has been encrypted.

Note: If database credentials change, an administrator needs to decrypt the customerconnstrings.config file, update the login credentials, and encrypt the file again. The commands to decrypt the file are as follows:

```
cd C:\Windows\Microsoft.Net\Framework\v4.0.30319

aspnet_regiis.exe -pdf connectionStrings "C:\ProgramFiles (x86)\Courion
Corporation\CourionARMS\CustomerConnStrings.config"
```

If you are also using the Identity Mapping solution for data collection, refer to the *Using the Identity Mapping Solution* manual for details about creating additional connection strings.

Note: The commands to decrypt and encrypt the CustomerConnStrings.config file can be used every time the file needs to be updated.

Configuring the Courion Request Service

Add the configuration for the Request Service to the *Courion.Framework.RequestService.exe.config* file found in the [courion-installation-folder]\CourionService folder. The Request Service processes the automated and manual requests on which approvers have taken action. The configuration file contains the following information:

```
<configuration>

  <appSettings>

    <add key="CourionServer" value="http://localhost/
courion/WebSamples/AccessOptions/XMLAO/xmlao.asp"/>

    <add key="NumberToProcess" value="5"/>

    <add key="SleepTime" value="1"/>

    <add key="AppName" value="Courion Request Service"/>

    <add key="ManualProcedure" value=""/>
```

```
<add key="LogLevel" value="0" />

  </appSettings>

  <connectionStrings>

    <!--<add name="dbConnectionString"
connectionString="Data Source=$$YourServerHere$$;Initial
Catalog=$$DatabaseName$$;User
ID=$$UserID$$;Password=$$Password$$"
providerName="System.Data.SqlClient"/>-->

  </connectionStrings>

</configuration>
```

Note: To encrypt the Courion.Framework.RequestService.exe.config file, refer to the Microsoft® documentation.

CoreServer: The link to the XMLAO processing engine where the Core Server is hosted.

NumberToProcess: The number of requests to process.

SleepTime: The number of minutes the service goes to sleep after processing the number of requests specified in NumberToProcess.

AppName: The name of the service.

ManualProcedure: By default, the value is empty.

LogLevel: The log level is classified into the following categories:

Warnings and Errors – 0

Information, Warnings, and Errors – 1

Trace, Information, Warnings and Errors – 2

dbConnectionString: The connection string points to the database which holds the ARM schema.

The *courrequestservice.log* log file for the Request Service is found in the [courion-installation-folder]\CourionService folder. It logs errors, warnings or any information related to the Request Service.

Configuring the Courion Notification Service

Add the configuration for the Notification Service to the *Courion.Framework.NotificationService.exe.config* file found in the [courion-installation-folder]\CourionService folder. The Notification Service sends notification when an event occurs, such as a request is submitted, approved or denied. The configuration file contains the following information:

```
<configuration>

  <appSettings>
```

```
    <add key="SMTPServer" value="$$YourSMTPServerName$$"
/>

    <add key="SenderAddress"
value="$$SenderEmailAddress$$" />

    <add key="SleepTime" value="1" />

    <add key="EnableLogging" value="true" />

    <add key="AppName" value="Courion Notification
Service" />

    <add key="UserName" value="$$UserName$$" />

    <add key="Password" value="$$Password$$" />

    <!--<add key="LogLevel" value=""/>-->

  </appSettings>

  <connectionStrings>

    <!--<add name="dbConnectionString"
connectionString="Data Source=$$YourServerHere$$;Initial
Catalog=$$DatabaseName$$;User
ID=$$UserID$$;Password=$$Password$$"
providerName="System.Data.SqlClient"/>-->

  </connectionStrings>

</configuration>
```

Note: To encrypt the Courion.Framework.NotificationService.exe.config file, refer to the Microsoft® documentation.

SMTPServer: The server that sends emails.

SenderAddress: The email address from which emails are sent to requesters, approvers and administrators.

SleepTime: The number of minutes the service goes to sleep after processing pending notifications.

EnableLogging: Not applicable; however, do not remove.

AppName: The name of the service.

UserName: The account used for authentication to the SMTP server.

Password: The password for the account used for authentication to the SMTP Server.

- The user account must have, at a minimum, “Send As” permission level from the SMTP server administrator.

LogLevel: Log level can be classified into different levels with 0 for error, 1 for warning and error, 2 for information, warning and error.

The *cournotificationservice.log* log file for the Notification Service is found in the [courion-installation-folder]\CourionService folder. It logs errors, warnings or any information related to the Notification Service.

Configuring Policy Violation Checking

By default, the Policy Violations Checking feature of the Access Request Manager is not enabled. To enable the feature, do the following:

1. Open the {AASInstallDir}\CourionARMS\web.config file in a text editor.
2. Modify the IntelligenceEngine.PolicyValidation appSetting entry by changing the value to AIE.

   ```
   <!-- This defines the mode with which policies can be
   validated. Valid values are "None" and "AIE". -->

   <add key="IntelligenceEngine.PolicyValidation"
   value="AIE" />
   ```

3. If you installed AIE on a different server than AAS or if AIE is using a non-default App name, uncomment the IntelligenceEngine.PolicyValidation.ServiceUri appSetting and modify the value accordingly.

   ```
   <!-- This is the URI at which the AIE policy validation
   service can be found.-->

   <!-- <add
   key="IntelligenceEngine.PolicyValidation.ServiceUri"
   value="http://localhost/AIAnalytics/DataServices/
   PolicyValidation.svc"/> -->
   ```

4. Save your changes and exit the text editor.

Once the feature is enabled, two global configurations apply to the feature:

- AccessCatalogPolicyOverrideAllowOverride
- AccessCatalogPolicyOverrideRequireJustification

When policy checking is enabled, requesters are allowed to request a policy override by default. If you do not want to allow requesters to request a policy override, change the value of AccessCatalogPolicyOverrideAllowOverride to ‘false’. When a policy violation is found, the user must modify the request and resubmit. The resubmission fails if the request has any associated policy violations.

If policy checking is enabled and if policy overrides are allowed, justification for each override is required. If you do not want to require the justification comment, set the value of AccessCatalogPolicyOverrideRequireJustification to ‘false’.

Chapter 4: Configuring the Web Service Options

This chapter describes how to configure the Core Web Services, using the Web Service Configuration Manager, and includes the following sections:

Launching the Web Service Configuration Manager

The Web Service Configuration Manager is a wizard that you use to configure the various Core Compliance web services. It also includes an option for archiving the web services configuration data. If you installed any of the web services without installing the Courion Server, the Configuration Manager starts automatically. If not, you can configure the server by selecting:

Start>All Programs>Core Core Compliance>Web Service Configuration Manager

Note: If you install all web services and the Courion Server onto a single server (a "Complete" installation), all web services are configured with default values. Although you can edit these values in the Web Service Configuration Manager, Core recommends that you leave them at their default values.

On a server in which the Connector Framework Manager is installed, it can take up to a minute or two to open the Web Service Configuration Manager the first time you run it.

The Web Service Configuration Manager has five configuration options. You can configure only one option at a time. The options are described in the following sections:

- *"Configuring Access Keys and a Pass Phrase" on page 76*
- *"Configuring the Connector Framework" on page 77*
- *"Configuring the Connector Framework Manager" on page 81*
- *"Archiving the Web Services Configuration" on page 90*
- *"Archiving the Web Services Configuration" on page 90*

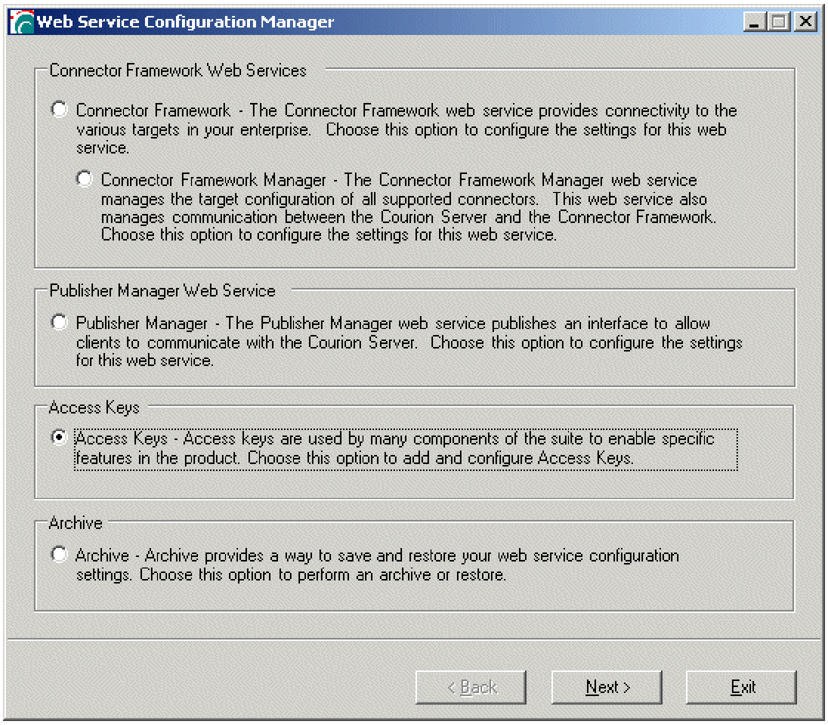
If a feature is not installed on the server, the corresponding configuration option is not available for selection.

If the Core Server feature is installed on the same server as any of the web services, the Access Keys option is not available in the Web Services Configuration Manager (but it can be configured from the Access Keys shortcut in the Start Menu).

If the Courion Server feature is not installed on the same server as any of the web services, the Access Keys option is the only available option the first time you access the Web Services Configuration Manager after installation (as shown in *Figure 36*). This is because you need to enter a pass phrase before any other configuration is allowed.

All fields in the configuration of the Connector Framework, Connector Framework Manager, and Publisher Manager have default values. The only exception to this occurs if the Publisher is installed on a separate server from the Core Server. In this case, you need to enter the Core Server name and Core Server port in the Publisher Manager Configuration.

Figure 36: Web Service Configuration Manager



Web Service Configuration Manager

Connector Framework Web Services

○ Connector Framework - The Connector Framework web service provides connectivity to the various targets in your enterprise. Choose this option to configure the settings for this web service.

○ Connector Framework Manager - The Connector Framework Manager web service manages the target configuration of all supported connectors. This web service also manages communication between the Courion Server and the Connector Framework. Choose this option to configure the settings for this web service.

Publisher Manager Web Service

○ Publisher Manager - The Publisher Manager web service publishes an interface to allow clients to communicate with the Courion Server. Choose this option to configure the settings for this web service.

Access Keys

● Access Keys - Access keys are used by many components of the suite to enable specific features in the product. Choose this option to add and configure Access Keys.

Archive

○ Archive - Archive provides a way to save and restore your web service configuration settings. Choose this option to perform an archive or restore.

Click **Next** to continue with access key and pass phrase configuration.

IIS Objects Created During Web Service Configuration

During web service configuration, the following objects are created in Internet Information Services (IIS), depending on which particular web services you install and configure:

- **Virtual Directories** — Virtual Directories are created within the Web Sites/Default Web Site folder: CourCF, CourCFM, and CourPublisher.
- **Application Pools** — The following application pools are created: CourCFAppPool, CourCFMAppPool, and CourPublisherAppPool.

  These application pools are created with the **Recycle work process (in minutes)** and **Shutdown worker processes after being idle for (time in minutes)** fields (located in the Properties dialog box) disabled. This prevents IIS from restarting or shutting down the web services. (The default application pool has these fields enabled.)
- **Message Queues** — Message queues are created within the Private Queues folder: cfmdispatchresponse and cfmessagedispatch. These are used for communication between the Connector Framework and Connector Framework Manager.

Configuring Access Keys and a Pass Phrase

The Access Key Configuration dialog lets you add access keys and specify a pass phrase.

Figure 37: Access Key Configuration

1. Since you added access keys in step 6 of the installation (see *Figure 9*), the keys appear in the left window. If you highlight a key, a description appears in the right window. To add more keys, click **ADD KEY FILE**.

   Note: If you add or delete access keys, you need to restart IIS for the changes to take effect.

2. Enter a **PASS PHRASE** and verify it by typing it again. The pass phrase you enter generates an encryption key. This key is used to encrypt configuration data to prevent access by unauthorized users. The key itself is encrypted using Microsoft CryptoAPI.

   Note: You must use the same pass phrase on all servers hosting the different components. If you decide to change the pass phrase at any time, be sure to go back and change it on all servers as described in *“Site-specific Information” on page 43*.

3. Click **FINISH** and then **YES** in the confirmation dialog.

   A warning message appears to notify you that the pass phrase must be identical on all servers hosting a component of the Core Compliance, and that if you have changed it here, you must change it on those servers.

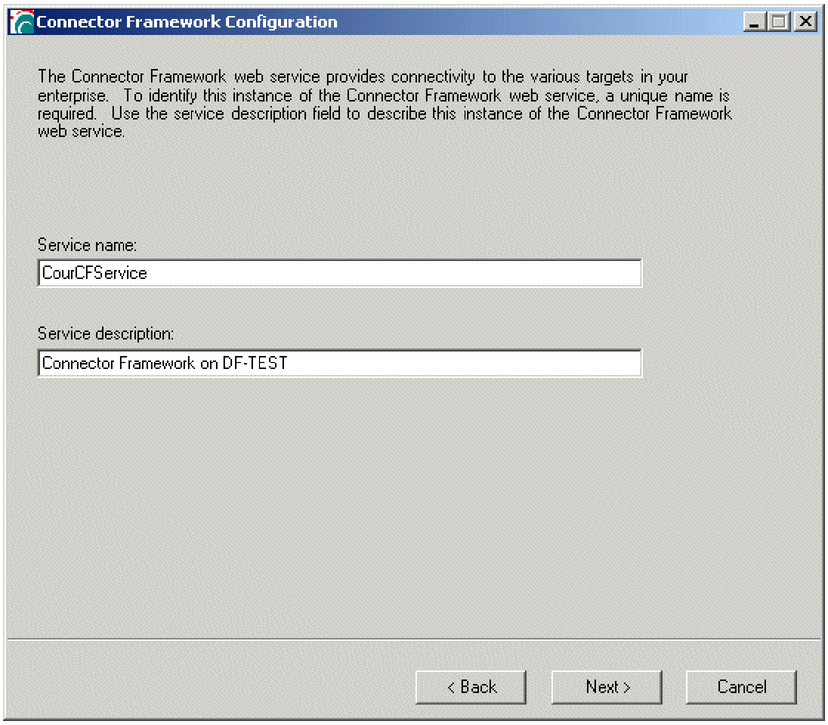
Configuring the Connector Framework

The Connector Framework (CF) web service provides connectivity to the various targets in your enterprise.

Service Name

The first dialog box that appears when you select the CF option is the Service Name dialog box.

Figure 38: Connector Framework Service Name



The Connector Framework web service provides connectivity to the various targets in your enterprise. To identify this instance of the Connector Framework web service, a unique name is required. Use the service description field to describe this instance of the Connector Framework web service.

Service name: CourCFService

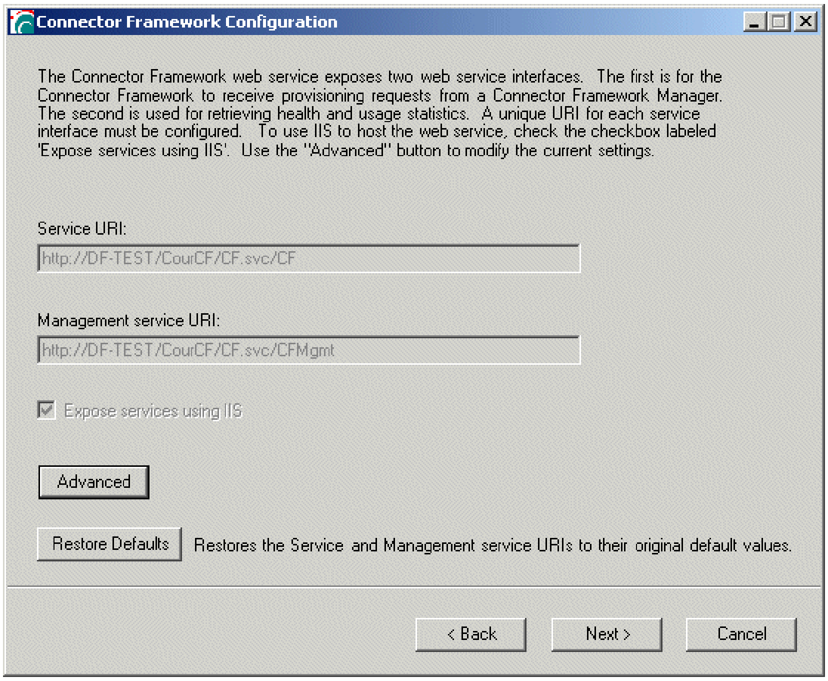
Service description: Connector Framework on DF-TEST

1. Enter a **SERVICE NAME** and **SERVICE DESCRIPTION** to uniquely identify this service. Click **NEXT**.

Uniform Resource Identifier

The Uniform Resource Identifier (URI) dialog box defines the names of two web service interfaces used by the CF.

Figure 39: Connector Framework URI



Connector Framework Configuration

The Connector Framework web service exposes two web service interfaces. The first is for the Connector Framework to receive provisioning requests from a Connector Framework Manager. The second is used for retrieving health and usage statistics. A unique URI for each service interface must be configured. To use IIS to host the web service, check the checkbox labeled 'Expose services using IIS'. Use the "Advanced" button to modify the current settings.

Service URI: http://DF-TEST/CourCF/CF.svc/CF

Management service URI: http://DF-TEST/CourCF/CF.svc/CFMgmt

☑ Expose services using IIS

Restores the Service and Management service URIs to their original default values.

2. By default, the fields on the CF URI dialog box are disabled for editing. Core Security recommends that you leave the default values. Click the **ADVANCED** button to edit the fields.

   SERVICE URI — The CF uses this service to receive provisioning requests from the Connector Framework Manager (CFM).

   MANAGEMENT SERVICE URI — This service retrieves health and usage statistics.

   EXPOSE SERVICES USING IIS — Select this checkbox to use IIS to host the web service. If you do not select this checkbox, the service is self hosted.

   If you edit the values, you can click the **RESTORE DEFAULTS** button to return the URIs to their original default values.

   Click **NEXT**.

URI Syntax when the Connector Framework is Self Hosted

If the CF is self hosted, the URIs must conform to the following syntax:

[protocol]://[hostname:port number]/[service endpoint]

URI Syntax when IIS Hosts the Connector Framework

If IIS hosts the CF, the URIs must conform to the following syntax:

[protocol]://[hostname]/[virtual directory name]/[service].svc/[service endpoint]

The following is an example of a valid CF service endpoint:

http://localhost/CourCF/CF.svc/CF

Additionally, endpoints for a particular service must share the same URI up to (but not including) the service endpoint. Using the previous example of a CF service endpoint, a valid management endpoint URI is:

http://localhost/CourCF/CF.svc/CFMgmt

The following is an invalid management endpoint URI because the file path up to the service endpoint is not identical:

http://localhost/CourCFMgmt/CF.svc/CFMgmt

Furthermore, Any 2 services that you configure for IIS must have different virtual directories. Using the previous example, the CFM's service could be

http://localhost/CourCFM/CFM.svc/CFM

but it could not be

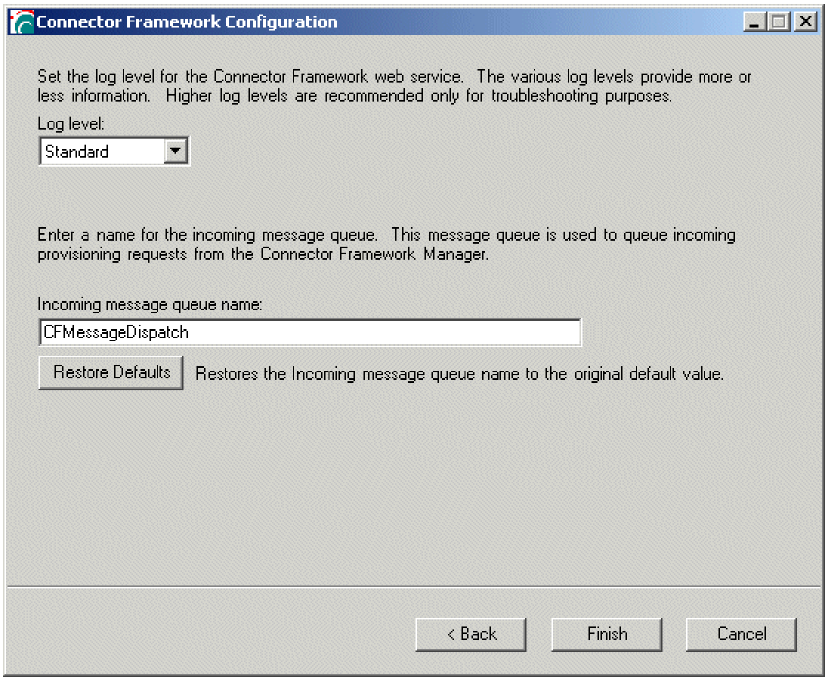
http://localhost/CourCF/CFM.svc/CFM

Note: If you enable SSL for the virtual directory on the server where the CF is installed, you need to change the URI to use HTTPS instead of HTTP. See*"Configuring SSL for Web Services" on page 91* for details.

Logging and Message Queuing

The Logging and Message Queue dialog box allows you to set the logging level and message queue.

Figure 40: Connector Framework Logging and Message Queue



Set the log level for the Connector Framework web service. The various log levels provide more or less information. Higher log levels are recommended only for troubleshooting purposes.

Log level: Standard

Enter a name for the incoming message queue. This message queue is used to queue incoming provisioning requests from the Connector Framework Manager.

Incoming message queue name: CFMessageDispatch

Restore Defaults — Restores the Incoming message queue name to the original default value.

3. **Log Level** — Set the log level for the CF service. The default is Standard. A value of Full provides more information in the log. To reduce the size of the log file, Core Security recommends that you select a value of Full only for troubleshooting purposes.
4. **Incoming message queue nam** — Enter a name for the incoming message queue. This message queue is used to queue incoming provisioning requests from the CFM.

   If you edit the value, you can click the **Restore Defaults** button to return the Incoming message queue name to its original default value.
5. Click **Next**.

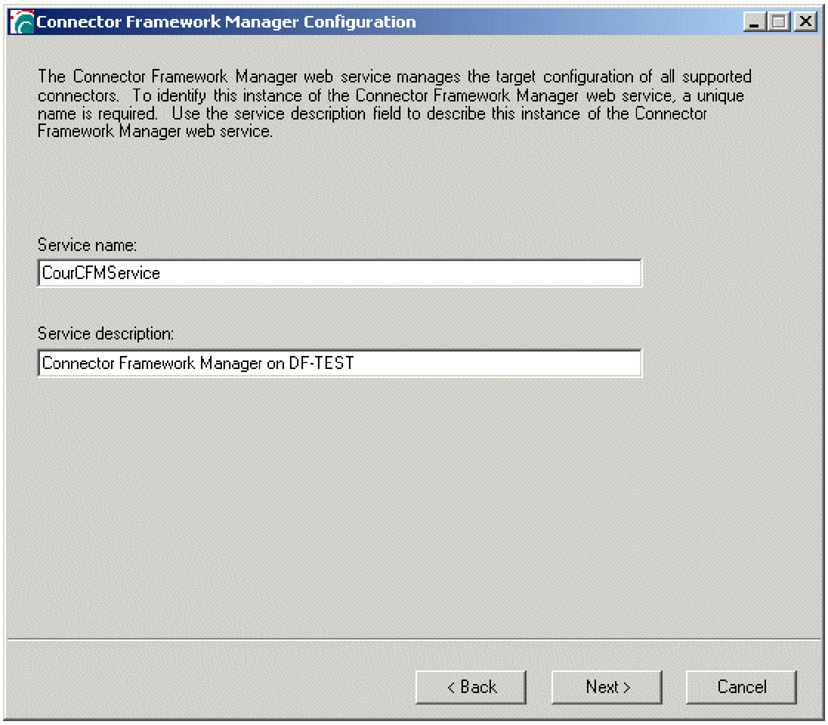
Configuring the Connector Framework Manager

The Connector Framework Manager (CFM) web service manages the target configuration of all supported connectors, as well as communication between the Core Server and the CF.

Service Name

The first dialog box displayed when you select the CFM option is the Service Name dialog box.

Figure 41: Connector Framework Manager Service Name



The Connector Framework Manager web service manages the target configuration of all supported connectors. To identify this instance of the Connector Framework Manager web service, a unique name is required. Use the service description field to describe this instance of the Connector Framework Manager web service.

Service name: CourCFMService

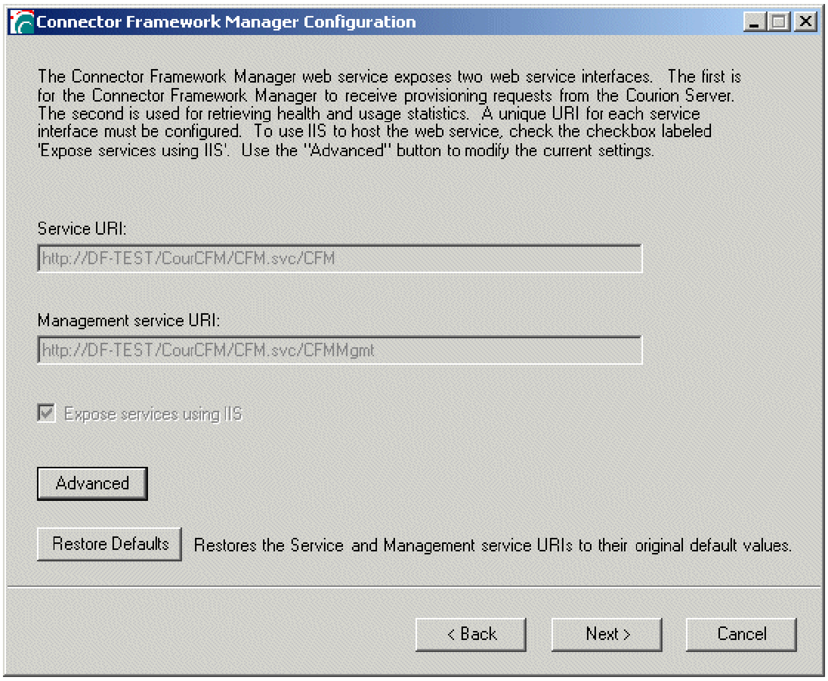
Service description: Connector Framework Manager on DF-TEST

1. Enter a **SERVICE NAME** and **SERVICE DESCRIPTION** to uniquely identify this service. Click **NEXT**.

Uniform Resource Identifier

The Uniform Resource Identifier (URI) dialog box defines the names of two web service interfaces used by the CFM.

Figure 42: Connector Framework Manager URI



Connector Framework Manager Configuration

The Connector Framework Manager web service exposes two web service interfaces. The first is for the Connector Framework Manager to receive provisioning requests from the Courion Server. The second is used for retrieving health and usage statistics. A unique URI for each service interface must be configured. To use IIS to host the web service, check the checkbox labeled 'Expose services using IIS'. Use the "Advanced" button to modify the current settings.

Service URI: http://DF-TEST/CourCFM/CFM.svc/CFM

Management service URI: http://DF-TEST/CourCFM/CFM.svc/CFMMgmt

☑ Expose services using IIS

Restore Defaults Restores the Service and Management service URIs to their original default values.

2. The fields on the CFM URI dialog box are disabled for editing by default. Core Security recommends that you leave the default values. Click the **ADVANCED** button to edit the fields.

   SERVICE URI — The CFM uses this service to receive provisioning requests from the Core Server and send the requests to the CF.

   MANAGEMENT SERVICE URI — This service retrieves health and usage statistics.

   EXPOSE SERVICES USING IIS — Select this checkbox to use IIS to host the web service. If you do not select this checkbox, the service is self hosted.

   If you edit the values, you can click the **RESTORE DEFAULTS** button to return the URIs to their original default values.

   Click **NEXT**.

URI Syntax when the Connector Framework Manager is Self Hosted

If the CFM is self hosted, the URIs must conform to the following syntax:

[protocol]://[hostname:port number]/[service endpoint]

URI Syntax when IIS Hosts the Connector Framework

If IIS hosts the CFM, the URIs must conform to the following syntax:

[protocol]://[hostname]/[virtual directory name]/[service].svc/[service endpoint]

The following is an example of a valid CFM service endpoint:

http://localhost/CourCFM/CFM.svc/CFM

Additionally, endpoints for a particular service must share the same URI up to (but not including) the service endpoint. Using the previous example of a CFM service endpoint, a valid management endpoint URI is:

http://localhost/CourCFM/CFM.svc/CFMMgmt

The following is an invalid management endpoint URI because the file path up to the service endpoint is not identical:

http://localhost/CourCFM/CFMgmt.svc/CFMMgmt

Furthermore, Any 2 services that are configured for IIS must have different virtual directories. Using the previous example, the CF's service could be:

http://localhost/CourCF/CF.svc/CF

but it could not be

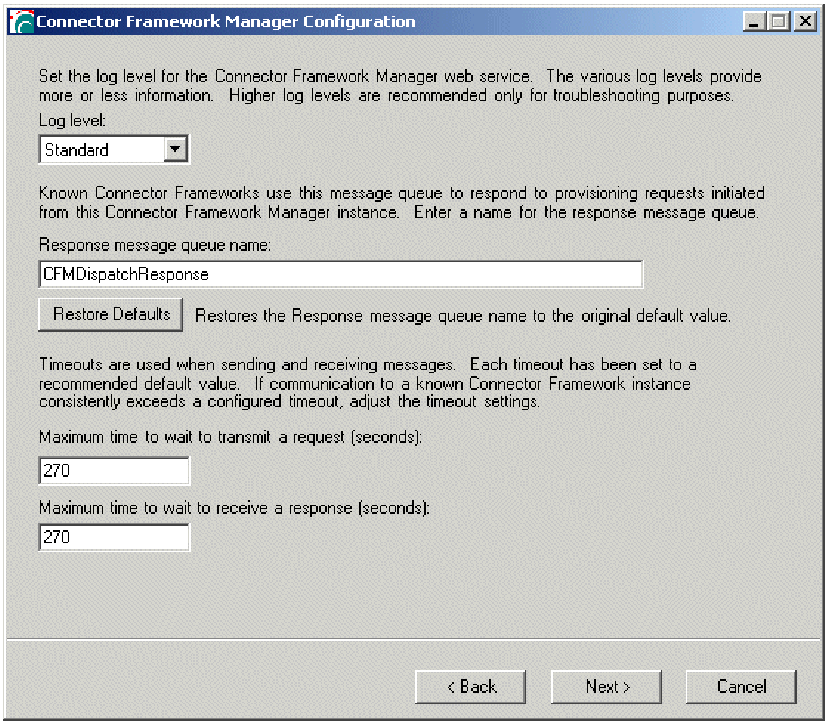
http://localhost/CourCFM/CF.svc/CF

Note: If you enable SSL for the virtual directory on the server where the CFM is installed, you need to change the URI to use HTTPS instead of HTTP. See*"Configuring SSL for Web Services" on page 91* for details.

Logging and Message Queuing

This dialog box allows you to set the logging level and message queue.

Figure 43: Connector Framework Manager Logging and Message Queue



Set the log level for the Connector Framework Manager web service. The various log levels provide more or less information. Higher log levels are recommended only for troubleshooting purposes.

Log level: Standard

Known Connector Frameworks use this message queue to respond to provisioning requests initiated from this Connector Framework Manager instance. Enter a name for the response message queue.

Response message queue name: CFMDispatchResponse

Restores the Response message queue name to the original default value.

Timeouts are used when sending and receiving messages. Each timeout has been set to a recommended default value. If communication to a known Connector Framework instance consistently exceeds a configured timeout, adjust the timeout settings.

Maximum time to wait to transmit a request (seconds): 270

Maximum time to wait to receive a response (seconds): 270

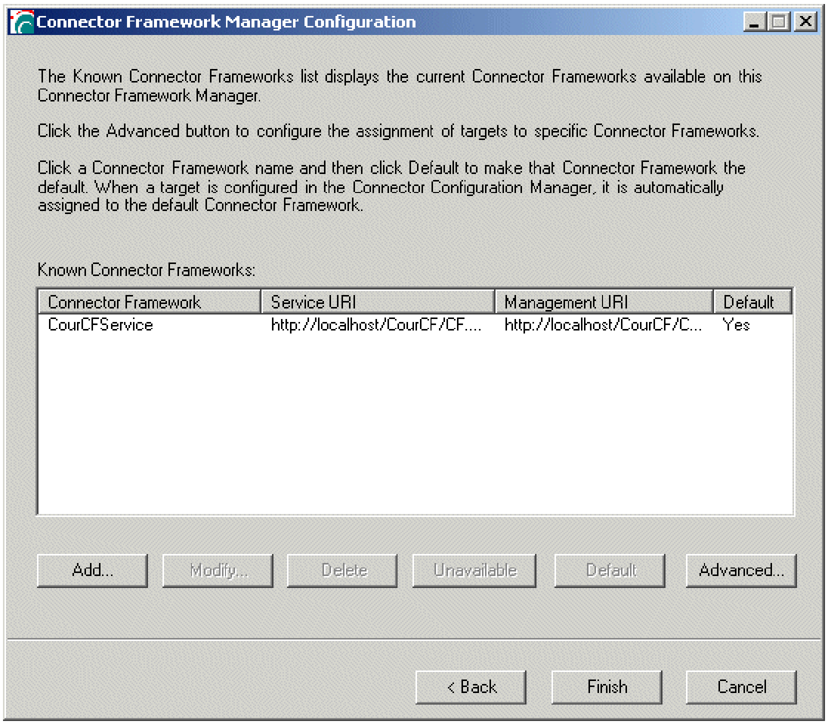
3. **Log Level** — Set the log level for the CFM service. The default is Standard. A value of Full provides more information in the log. To reduce the size of the log file, Core Security recommends that you select a value of Full only for troubleshooting purposes.
4. **Response message queue name** — Enter a name for the incoming message queue. This message queue is used to queue incoming responses from the CF.

   If you edit the value, you can click the **Restore Defaults** button to return the Response message queue name to its original default value.
5. **Maximum time to wait to transmit a request (seconds)** and **Maximum time to wait to receive a response (seconds)** — Enter the number of seconds before a timeout is issued. Timeouts determine when a known CF instance has become unresponsive.
6. Click **Next**.

Known Connector Frameworks

The Known Connector Frameworks list displays the current Connector Frameworks available on this Connector Framework Manager.

Figure 44: Known Connector Frameworks



Connector Framework Manager Configuration

The Known Connector Frameworks list displays the current Connector Frameworks available on this Connector Framework Manager.

Click the Advanced button to configure the assignment of targets to specific Connector Frameworks.

Click a Connector Framework name and then click Default to make that Connector Framework the default. When a target is configured in the Connector Configuration Manager, it is automatically assigned to the default Connector Framework.

Known Connector Frameworks:

Connector Framework	Service URI	Management URI	Default
CourCFService	http://localhost/CourCF/CF....	http://localhost/CourCF/C...	Yes

7. A CF that is installed on the same server as the CFM automatically appears in the list of known CFs, as shown in the example in *Figure 44*. If a CF is installed and configured on a different server, click **Add** to open the Add Connector Framework dialog box, as shown in *Figure 45*. To modify the settings of a CF, select the Connector Framework name, then click **Modify**. To delete a CF from the list, click the Connector Framework name, then click **Delete.** You can not delete the default CF

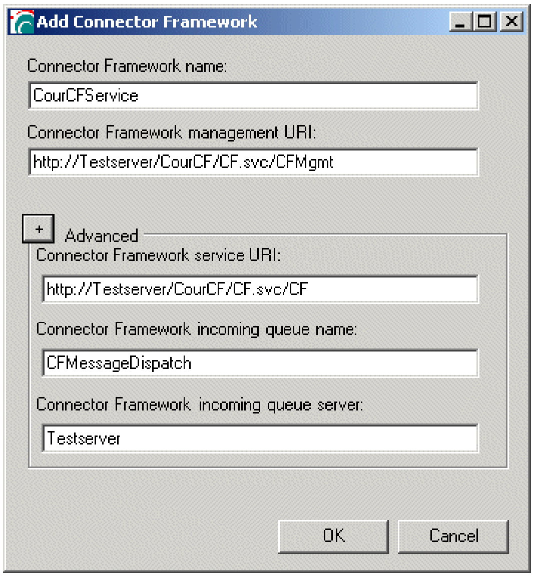
If a server hosting a CF is down for maintenance or troubleshooting, you can click the Connector Framework name, then click **Unavailable**. This eliminates the time spent waiting before a time-out is issued and therefore reduces the amount of time an end-user waits before receiving an error, when running a workflow that uses a target on that CF. You can not make the default CF unavailable.

8. One CF in the list is designated as the default CF. When you configure a target in the Connector Configuration Manager, it is automatically assigned to the default CF. To change the default, click the Connector Framework name, then click **Default**.

9. To move a target to a different CF or assign the target to more than one CF, edit the target assignment. Click the **Advanced...** button to open the Target Assignment list (see *"Target Assignment" on page 86*).

   Note: If you install and configure or remove a CF after you configure the CFM, you need to reconfigure the CFM to reflect the change.

Figure 45: Add Connector Framework



Add Connector Framework

Connector Framework name: CourCFService

Connector Framework management URI: http://Testserver/CourCF/CF.svc/CFMgmt

Advanced

Connector Framework service URI: http://Testserver/CourCF/CF.svc/CF

Connector Framework incoming queue name: CFMessageDispatch

Connector Framework incoming queue server: Testserver

10. On the Add Connector Framework dialog box, enter the **Connector Framework name**, and the **Connector Framework management URI**. These fields are pre-populated with default values. The default URI is based on the server that the CFM is installed on. This allows you to change the server name to the name for the desired CF and leave the rest of the path name set to default values.

    Click the **+** button to expand the dialog and show the **Connector Framework service URI**, **Connector Framework incoming queue name**, and **Connector Framework incoming queue server** fields. The service URI and incoming queue server fields are automatically updated based on the server name you specify in the management URI field, so you should not need to edit these fields.

    Click **OK** to return to the Known Connector Frameworks dialog box.

11. Click **Finish** to complete the CFM configuration.

Target Assignment

When you click the **ADVANCED...** button on the Known Connector Frameworks dialog, the Target Assignment list dialog opens. This list defines the assignment of each available target to one or more Connector Frameworks. Target assignment is based on the category of connector used to communicate with the target.

There are two categories of connectors:

- **Standard Connector** — You can assign a target associated with a standard connector to one Connector Framework.
- **Distributable Connector** — You can assign a target associated with a distributable connector to multiple Connector Frameworks.

  Note: It is possible to re-categorize a standard connector as a distributable connector. See *“Moving a Standard Connector to the List of Distributable Connectors” on page 20* for information about how to do this.

In the Target Assignment list dialog, you can move the target from one CF to another, or (if the target uses a distributable connector) you can copy a target and add it to additional CFs.

The Target Assignment list has two views. Both views display the targets in a tree structure, but each view organizes the information differently.

- The *“Connector Framework View”* displays each CF and the CourionService at the top level, with the connectors and associated targets listed under the CF, as shown in *Figure 46* and *Figure 47*.
- The *“Target View”* displays each target and associated connector at the top level, as shown in *Figure 48*.

Connector Framework View

The Target Assignment list dialog opens in Connector Framework view, with the tree collapsed so that only the individual Connector Frameworks are visible. A Connector Framework has a **+** icon next to it if any targets are assigned to that CF. Click the **+** to see a list of connectors assigned to that CF, then click the **+** next to the connector to see the specific targets assigned to that CF.

Click **EXPAND ALL**, to show all targets. Click **COLLAPSE ALL** to show only the CFs.

There are two different icons used for connectors:

- The □ icon indicates a standard connector.
- The ⊡ icon indicates a distributable connector.

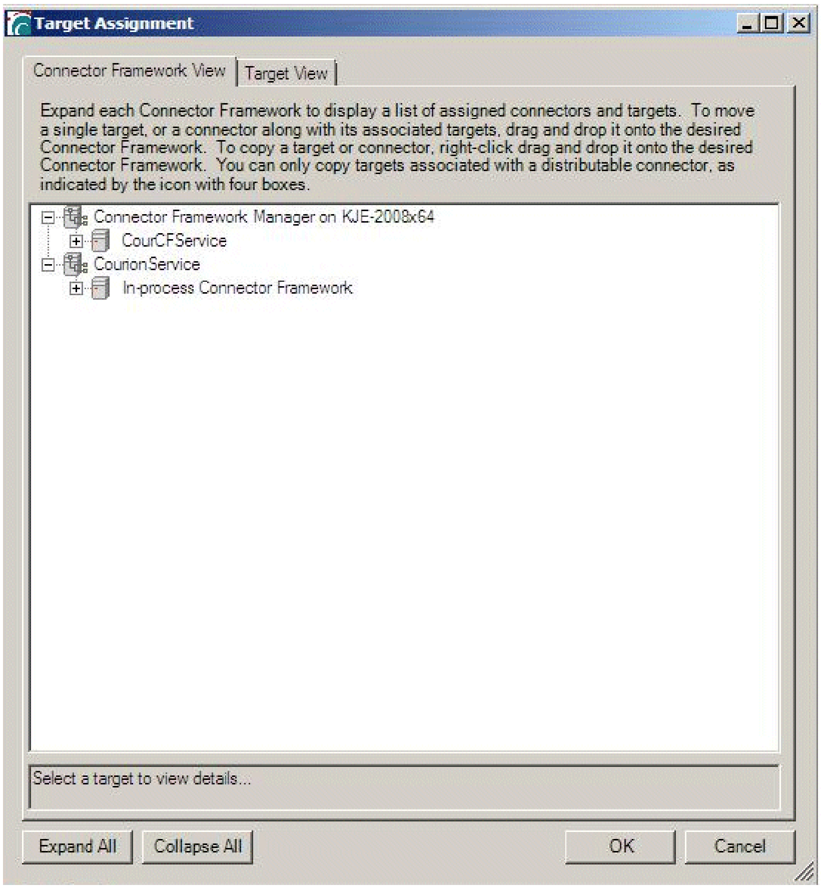
Target icons also indicate whether the associated connector is standard or distributable.

- The ■ icon indicates a target associated with standard connector.
- The ⊞ icon indicates a target associated with a distributable connector.

If a connector is distributable, the associated targets are as well. If a connector is a standard connector, the associated targets are not distributable. The only exception to this is the PMM Gateway Connector. Since that connector is used for targets that access different PMMs, the PMM Gateway Connector uses a distributable icon while the associated PMM targets may use a standard or distributable icon.

Figure 46 shows an example of Connector Framework View collapsed. Expand each CF to display a list of assigned connectors and targets as shown in *Figure 47*.

Figure 46: Connector Framework View (Collapsed)



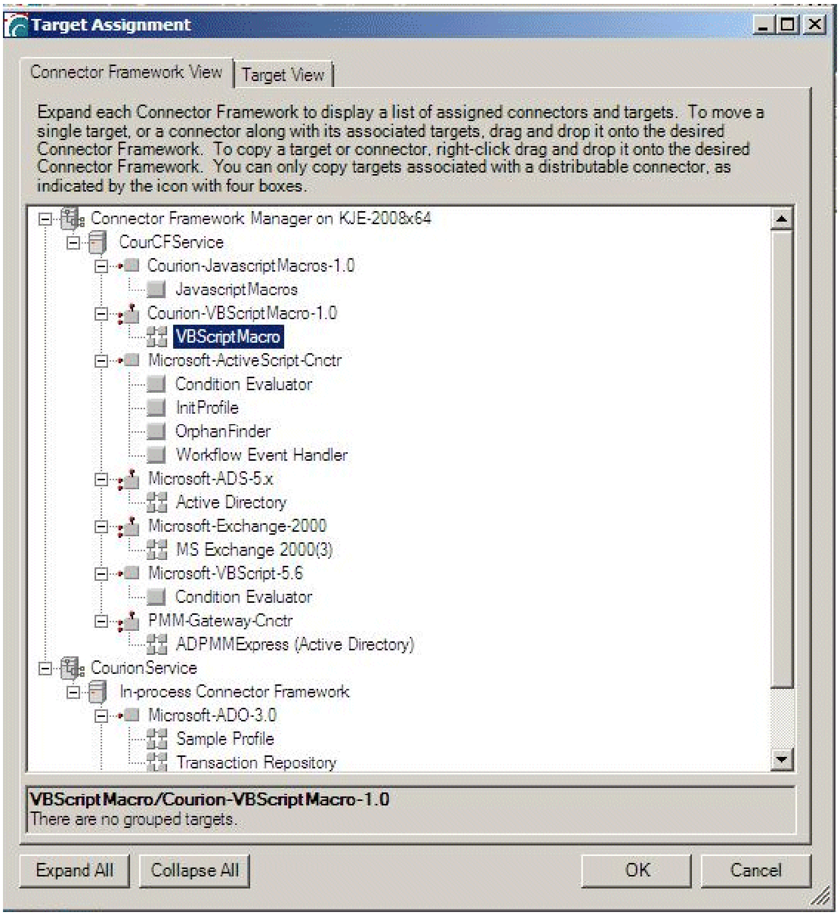
Expand each Connector Framework to display a list of assigned connectors and targets. To move a single target, or a connector along with its associated targets, drag and drop it onto the desired Connector Framework. To copy a target or connector, right-click drag and drop it onto the desired Connector Framework. You can only copy targets associated with a distributable connector, as indicated by the icon with four boxes.

Connector Framework Manager on KJE-2008x64
CourCFService
CourionService
In-process Connector Framework

Select a target to view details...

Figure 47 shows an example of Connector Framework View expanded.

Figure 47: Connector Framework View (Expanded)



Expand each Connector Framework to display a list of assigned connectors and targets. To move a single target, or a connector along with its associated targets, drag and drop it onto the desired Connector Framework. To copy a target or connector, right-click drag and drop it onto the desired Connector Framework. You can only copy targets associated with a distributable connector, as indicated by the icon with four boxes.

VBScriptMacro/Courion-VBScriptMacro-1.0
There are no grouped targets.

1. To move a target from one CF to another, click and hold the target icon, then drag and drop it on top of a CF icon. You can also drag a connector icon onto a CF icon. All targets associated with that connector are moved at the same time.

   Click on a target to see details in the text box at the bottom of the window (*Figure 47*).

   You can only move connectors between a CF and the CourionService. To move a connector onto or off of the CourionService, drag and drop the connector icon.

   Note: Some PMM targets are grouped with other PMM targets. You need to assign grouped PMMs to the same CF. If you attempt to move a grouped PMM target, a warning message appears that informs you that the target is grouped and lists the other targets that will be moved. You can then choose whether to move all the targets or cancel the operation.

2. To copy a target to an additional CF, right-click and hold the target icon, then drag and drop it on top of a CF icon. You can only copy targets that use distributable connectors, as indicated by the ⊞ icon.

   Note: You can not remove a target from a CF in the Connector Framework view. If you want to remove a target that you previously copied to a CF, switch to Target view, where you can unassign that CF for the target.

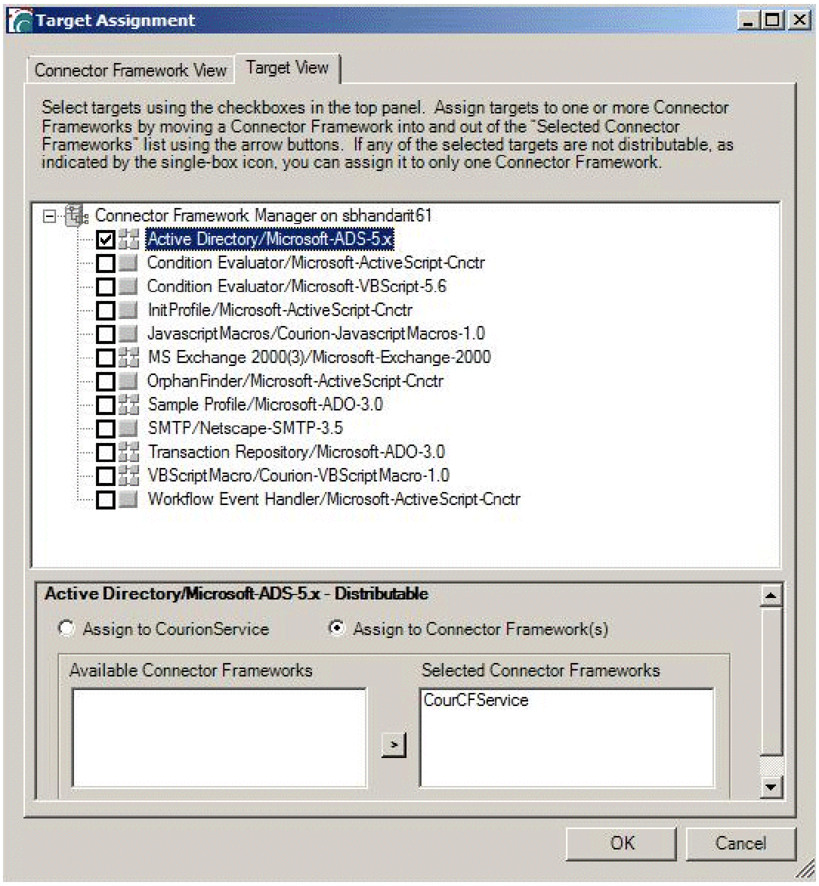
3. Click **OK** to return to the Known Connector Frameworks dialog.

Target View

The Target view initially displays with the tree expanded, with the individual targets visible. Each target entry shows both the target name and the connector that is used by that target.

When you click on a target, you see a list of all CFs configured for the CFM at the bottom of the window. You assign the target to a CF or the CourionService using the radio buttons and arrow buttons. *Figure 48* shows an example of Target View with a target associated with a CF.

Figure 48: Target View



Target Assignment

Connector Framework View | Target View

Select targets using the checkboxes in the top panel. Assign targets to one or more Connector Frameworks by moving a Connector Framework into and out of the "Selected Connector Frameworks" list using the arrow buttons. If any of the selected targets are not distributable, as indicated by the single-box icon, you can assign it to only one Connector Framework.

- Connector Framework Manager on sbhandarit61
  - ☑ Active Directory/Microsoft-ADS-5.x
  - □ Condition Evaluator/Microsoft-ActiveScript-Cnctr
  - □ Condition Evaluator/Microsoft-VBScript-5.6
  - □ InitProfile/Microsoft-ActiveScript-Cnctr
  - □ JavascriptMacros/Courion-JavascriptMacros-1.0
  - □ MS Exchange 2000(3)/Microsoft-Exchange-2000
  - □ OrphanFinder/Microsoft-ActiveScript-Cnctr
  - □ Sample Profile/Microsoft-ADO-3.0
  - □ SMTP/Netscape-SMTP-3.5
  - □ Transaction Repository/Microsoft-ADO-3.0
  - □ VBScriptMacro/Courion-VBScriptMacro-1.0
  - □ Workflow Event Handler/Microsoft-ActiveScript-Cnctr

Active Directory/Microsoft-ADS-5.x - Distributable

○ Assign to CourionService ● Assign to Connector Framework(s)

Available Connector Frameworks	Selected Connector Frameworks
	CourCFService

1. To move a target from one CF to another CF:
   - Check the checkbox next to the target you want to move.
   - Use the arrow buttons to move Connector Frameworks in and out of the **SELECTED CONNECTOR FRAMEWORKS** list.

   Note: Only targets with the ⊞ icon are distributable. You cannot move targets from one CF to another if they have the ▢ icon.

   You cannot move individual targets onto or off of the CourionService. If you attempt to move some, but not all, targets associated with a connector to or from the CourionService, the Web Service Configuration prompts you to approve moving all other targets associated with that connector to the same location.

2. Click **OK** to return to the Known Connector Frameworks dialog.

Archiving the Web Services Configuration

You can archive the configuration settings of the individual web service components from the Archive option of the Web Services Configuration Manager. In addition, all web services configuration settings are archived if you use the standard Archive function, accessed from the Administration Manager.

For details on archiving, see the chapter “Using the Archive Option” in the manual *Configuring Workflows with the Core Compliance Administration Manager*.

Configuring SSL for Web Services

Note: The information in this section applies only if your web services are hosted by IIS (the default setting).

If you install a server certificate on any web server hosting a Core web service, you should install the same certificate on any other servers hosting a Core web service. If you enable SSL on a server with a certificate installed, you need to change the default settings for all of the Core web services on that server. Since data communication between the different components of the Core Compliance is encrypted with AES, you do not need to enable SSL for secure communication. However, if you decide that you do want to use SSL, all of the Core web services on that server must use SSL.

By default, the SSL configuration of all virtual directories installed within a web server match the configuration of the web server. Also by default, the URIs of the web services use HTTP instead of HTTPS. If the web services are configured for SSL and the URIs use HTTP, communication fails.

You have two options to restore communication in this situation:

- Option 1. Disable SSL for each Core web service virtual directory. Since the Core Compliance uses AES encryption, Core recommends using this option unless your company policy requires the use of HTTPS.
- Option 2. Change each URI to use HTTPS.

To disable SSL for each web service virtual directory:

1. Go to the Control Panel and double-click on the Administrative Tools icon.
2. Double-click the **INTERNET INFORMATION SERVICE MANAGER** icon in the window that comes up.
3. In the tree in the left pane, expand the local computer, then expand the **WEB SITES** folder.
4. Click on **DEFAULT WEB SITE.**
5. Right-click on the virtual directory for one of the web services (**COURCF**, **COURCFM**, or **COURPUBLISHER**) and select **PROPERTIES**.
6. Select the **DIRECTORY SECURITY** tab. In the Secure Communications section, click **EDIT**.
7. Uncheck the **REQUIRE SECURE CHANNEL (SSL)** checkbox and click **OK** twice to return to the main IIS Manager window.
8. Repeat steps 5 through 7 for each of the Courion web services installed on that server.

To change each URI to use HTTP:

1. In the Web Service Configuration Manager, select one of the web service configuration options and click **NEXT** to display the Uniform Resource Identifier dialog box.
2. Click **ADVANCED** and edit both the **SERVICE URI** and **MANAGEMENT SERVICE URI** fields to change the HTTP in the URI to HTTPS.
3. Repeat steps 1 and 2 for each web service installed on that server.

Adjusting Session Timeout for Access Assurance Portal Applications

By default, when a user is logged into an Access Assurance Portal application such as the Access Request Manager solution, Core Compliance Access Certification, and the Identity Mapping solution, the session will timeout after twenty minutes of no activity. To change the length of time before a session timeout occurs, do the following:

1. Open the [courion-installation-folder]\CourionARMS\AspxCommon\Web.config file for editing.
2. In the SessionTimeout entry, replace 20 with the number of minutes for which a session can be inactive before timing out:

   ```
   <add key="SessionTimeout" value="20"/>
   ```

   Note: The minimum value is 1. You cannot set a session timeout to less than one minute.
3. Save your changes and exit the file.

Chapter 5: Using the ConfigPortalAuthentication Utility

The ConfigPortalAuthentication command line utility enables administrators to select the type of authentication to use for the portal: either by using the Active Directory connector or bypassing it. By using this utility, you can also enable integrated authentication.

If connector authentication is selected, the user specifies the connector and the target for authentication.

If connector bypass authentication is selected, the user has the option for manual login or integrated authentication with Windows credentials. The user needs to specify the domain controller name and domain against which authentication should occur. If domain information is not specified then the user's current domain is used.

You use the ConfigPortalAuthentication utility by issuing commands from the Windows Command Prompt. This chapter describes how to access the ConfigPortalAuthentication utility, use the utility to modify the portal's authentication mode, and provides examples of how to do this in the following sections:

- *"Using the ConfigPortalAuthentication Utility from the Windows Command Prompt" on page 95*
- *"Examples" on page 97*
- *"Logging" on page 98*

Multi-Domain Authentication

The multi-domain feature enables a user to authenticate in to the Access Assurance Portal by selecting a Microsoft® Active Directory® domain from a drop-down list. The user name and password provided are authenticated against the selected domain, and the user is authorized to start using the Access Request Manager.

The ConfigPortalAuthentication command line utility includes two arguments that enable multi-domain authentication, depending on the type of portal authentication you select (see Table 3). To use multi-domain authentication, you also need to specify certain options in the Core Configuration Repository. See *"Configuring Multi-Domain Authentication" on page 99* for information about how to do this.

Requirements

You need to be a member of the local Administrator group on the local system where you run the ConfigPortalAuthentication utility.

You need to launch the ConfigPortalAuthentication utility with the Run as Administrator option.

Stopping and Starting Courion Services

When you have completed running the utility, restart the Courion Community Claim Provider.

Using the ConfigPortalAuthentication Utility from the Windows Command Prompt

To access the ConfigPortalAuthentication utility from the Windows command prompt, navigate to the following directory:

<installdir>:\<installpath>\CourionService

The default location is:

C:\Program Files\Courion Corporation\CourionService

You can then enter the ConfigPortalAuthentication.exe command with arguments to modify portal’s authentication mode.

ConfigPortalAuthentication.exe Arguments

Table 3 describes the `ConfigPortalAuthentication.exe` command arguments. Entering ConfigPortalAuthentication.exe without arguments displays a list of all available arguments.

Table 3: ConfigPortalAuthentication.exe Arguments

Argument	Description

`-help`
`-h`
`-?`
	Displays help for this command.

`-BypassConnector`	Use Active Directory Authentication without the connector.
`-UseConnector`	Use authentication with the connector you specify.
`-IntegratedAuth`	Use Windows Integrated Authentication. The user is not prompted for credentials, and no login screen appears.
`-Server="Server Name"`	Use Active Directory Authentication without the connector in the domain residing in the domain controller you specify.
`-Domain="Domain Name"`	Use Active Directory Authentication without the connector in the domain you specify.

Table 3: ConfigPortalAuthentication.exe Arguments (Continued)

Argument	Description
`-Connector="Connector Name"`	Use authentication with the connector you specify.
`-Target="Target Name"`	Use authentication against the target you specify.
`-MultiDomainAuth`	Use multi-domain authentication for authentication and authorization to the portal. When using this argument, the end user is prompted for the distinguished names of different target groups. To get the distinguished names of an Active Directory group, execute the following command from a command prompt: dsquery group –name "<group name>" Example: dsquery group –name "administrators"
`-AuthenticationModule="ADForestAuthModule.dll"`	The name of the Authentication Module located in CourionService\WS\bin

Examples

Example of using Active Directory authentication without a connector in the specified domain:

```
ConfigPortalAuthentication -BypassConnector -Server="server1.corp.com"
-Domain="corp.com"
```

Example of using Active Directory authentication with the connector against a specific Active Directory target:

```
ConfigPortalAuthentication -UseConnector -Connector="Microsoft-ADS-5.x"
-Target="Active Directory"
```

Example of using ActiveX Data Object (ADO) authentication with the connector against a specific ADO target:

```
ConfigPortalAuthentication -UseConnector -Connector="Microsoft-ADO-3.0"
-Target="Transaction Repository"
```

Example of using multi-domain authentication when the login screen appears:

```
ConfigPortalAuthentication -MultiDomainAuth
-AuthenticationModule="ADForestAuthModule.dll"
```

After the user presses enter, the user is prompted to enter the Target Group Domain Name for each Dynamic Community in the Configuration Repository, except the Everyone community.

Example of using multi-domain authentication with integrated authentication when the login screen does not appear:

```
ConfigPortalAuthentication -MultiDomainAuth
-AuthenticationModule="ADForestAuthModule.dll" -IntegratedAuth
```

After the user presses the Enter key, the user is prompted to enter the Target Group Domain Name for each Dynamic Community in the Configuration Repository, except the Everyone community.

Logging

For each action related to authentication, all messages are logged to the courion.log file.

Configuring Multi-Domain Authentication

To configure multi-domain configuration, follow the steps in this section:

- Configure the Web.config Files for Multi-Domain Authentication
- Configuring a Database
- Adding Values Using the Configuration Editor
- Configuring the Configuration Editor and Web.config Files for Integrated Authentication
- Additional Configuration for Access Request Manager to Support Multi-Domain Authentication

Modifying the Web.config Files for Multi-Domain Authentication

To support multi-domain authentication, add the following text to the Analytics\AccessCertification\Web.config and Analytics\AccessRequest\Web.config files:

```
<InputFields>

     <add id="Account Name" label="User name:" controlType="TEXTBOX"/>

     <add id="Password" label="Password:" controlType="PASSWORD"/>

<add id="Domain Name" label="Domain:" controlType="DROPDOWN"
connectionStringName="MetricRepositoryDefault"
storedProcedureName="spGetDomainNames"/>

</InputFields>
```

The order that the fields are displayed to the user is determined by the order in which they are listed in the InputFields section. The attributes specified in the InputFields are:

- **Id** - A required Attribute that is case-sensitive such as Account Name, Password, or Domain Name.
- **Label** - Text that is displayed to the user at the time of authentication to the Access Assurance Suite portal page. If you do not specify a label, the value specified for the Id attribute appears.
- **ControlType** - A required attribute that determines whether to display a textbox or a drop-down list:

  TEXTBOX - Accept text input.

  PASSWORD - Accept a password as input.

  DROPDOWN - Display a list of values in a drop-down lit.

  If you are specify DROPDOWN, two additional attributes are required to specify the data source: connectionStringName and storedProcedureName. The connectionStringName specifies the database connection string name defined in the respective Web.config file. The storedProcedureName specifies the name of the stored procedure that returns the list of domain names to display in the id=Domain Name drop-down list.

Configuring a Database

The domains displayed in the drop-down list on the Access Assurance Suite portal page are supplied from a database. This database needs to have a stored procedure, such as spGetDomainNames. This stored procedure fetches the content from the first column of the returned table to populate the drop-down list.

The following sample SQL script creates a stored procedure that returns one domain:

```
CREATE PROCEDURE spGetDomainNames

AS

 BEGIN

     -- SET NOCOUNT ON added to prevent extra result sets from

     -- interfering with SELECT statements.

     SET NOCOUNT ON;

     SELECT 'mydomain.com'

 END

GO
```

The stored procedure returns a list of domain names by selecting from a table, as follows:

```
SELECT domainNames from MyDomainsTable
```

Chapter 6: Configuring a Proxy Server for Remote Password Management

This chapter explains how to configure a proxy server for remote password management and includes the following sections:

Overview

This chapter describes how to configure and use the Access Assurance Suite Web Service for Remote Password Management (WSRPM) to perform remote password reset requests. This feature is available for use in PasswordCourier Classic only.

The WSRPM feature allows users to execute remote password reset requests. This ability to perform remote resets removes requirements such as needing a domain trust relationship between the target domain and the domain in which the Core Server is operating. *Figure 49* illustrates the relationship between the user making the reset request and the remote PMM agent.

Figure 49: Web Service for Remote Password Management

Reset requests are sent to the remote machine by using the Web services technology. Since Web services use HTTP(S) as the data transport mechanism, requests can be routed easily to the remote machine.

You may configure multiple servers for remote password reset requests. Many PMMs process password reset requests serially. By configuring more than one server for remote password reset requests, simultaneous password reset requests on a target are sent out evenly among the set of remote servers, thereby reducing the load on any single server.

Configuring a Remote Proxy Server

To configure a remote proxy server, you must select the Remote Password Management on the Select Features dialog box during installation (see *Figure 11 on page 34*).

The Configuration Manager for Remote Password Management is a wizard that walks you through configuration of a remote proxy server.

If you selected the Configure Password Management Modules option at the end of installation, the Configuration Manager starts automatically. If not, you can configure the server by selecting:

Start>All Programs>Core Access Assurance Suite>Configuration Manager

The Configuration Manager lets you configure PMMs. The PMM contains information needed to perform password management on systems connected to the Core Server

When you install the Access Assurance Suite, any PMM to which you have an access key is automatically included on the list of added PMMs.

If you installed the Remote Password Management feature on more than one server, you must run the Configuration Manager on each server in which the option is installed, and configure the individual PMMs on each server.

Access Keys

Figure 50: Access Key Selection

1. Since access keys were added in step 6 of the installation (see *Figure 9*), the keys appear in the left window. If you highlight a key, a description appears in the right window. To add more keys, click **ADD KEY FILE**.

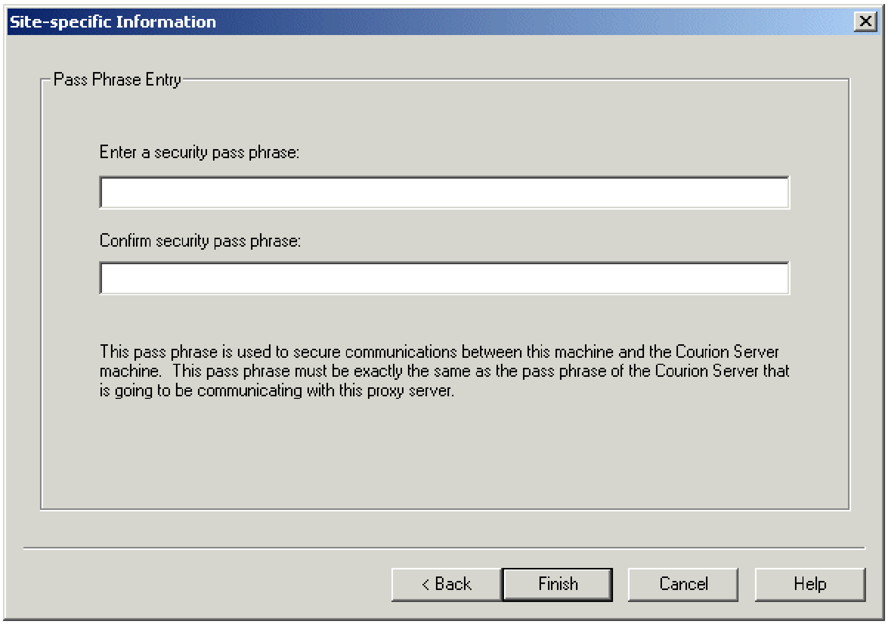
You can also add more keys from the Start menu after installation and configuration is complete, by selecting

Start>All Programs>Core Access Assurance Suite>Access Keys

Note: If you add or delete access keys, you need to restart IIS for the changes to take effect.

Click the **NEXT** button.

Figure 51: Pass Phrase Entry

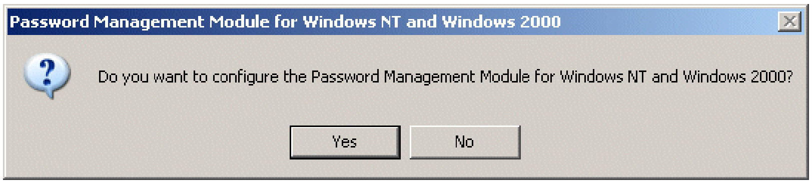


Site-specific Information
Pass Phrase Entry
Enter a security pass phrase:
Confirm security pass phrase:
< Back
Finish
Cancel
Help

This pass phrase is used to secure communications between this machine and the Courion Server machine. This pass phrase must be exactly the same as the pass phrase of the Courion Server that is going to be communicating with this proxy server.

2. Enter the **PASS PHRASE** for the remote machine. This pass phrase must match the one used on the Core Server. Click **FINISH**.
3. You are prompted to configure individual PMMs. An example is shown in *Figure 52*. The individual PMMs you are prompted to configure are dependent on the installed access keys.

Figure 52: Update PMM Configurations



Do you want to configure the Password Management Module for Windows NT and Windows 2000?

4. Click **YES** to configure each password reset target. See the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents* for details on configuring individual PMMs.
5. The installation creates a web share folder in the following location:

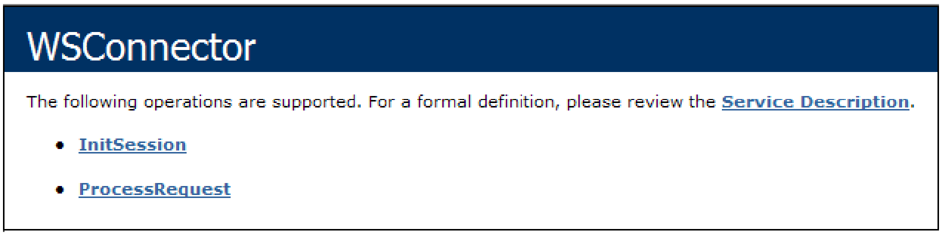
```
C:\Program Files\Core Security\www\CourRPM
```

Test the new web share by using a browser and navigating to the following URL:

```
http://localhost/CourRPM/WSConnector.asmx
```

The content should look similar to *Figure 53*.

Figure 53: WS Connector



WSConnector

The following operations are supported. For a formal definition, please review the Service Description.

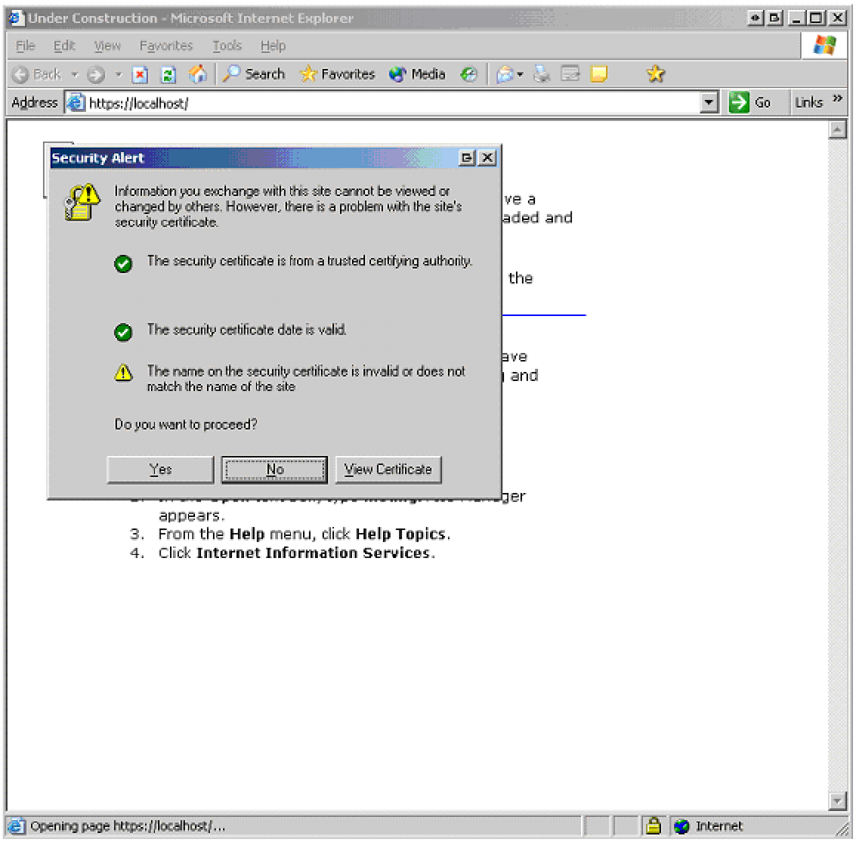
- InitSession
- ProcessRequest

HTTPS is not enabled in IIS by default. HTTPS requires that a valid server certificate be installed in IIS. Refer to the Microsoft IIS documentation, available at the following location, for more information about how to do this:

```
http://www.microsoft.com/resources/documentation/
WindowsServ/2003/standard/proddocs/en-us/
sec_ssl_certsetssl.asp
```

To test that the HTTPS URL does not produce warnings, launch Internet Explorer and browse to the secure URL. If a warning dialog box appears, such as the one in *Figure 54*, you must resolve the warning before you can use the URL as a remote target.

Figure 54: Security Alert Dialog Box Example



Under Construction - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go
Links
Security Alert
Yes
No
View Certificate
Internet

Address: https://localhost/

Information you exchange with this site cannot be viewed or changed by others. However, there is a problem with the site's security certificate.

✓ The security certificate is from a trusted certifying authority.

✓ The security certificate date is valid.

⚠ The name on the security certificate is invalid or does not match the name of the site

Do you want to proceed?

appears.
3. From the **Help** menu, click **Help Topics**.
4. Click **Internet Information Services**.

Opening page https://localhost/...

There are several different types of security alerts. Two of the most common are:

- **COMMON NAME IN THE SERVER CERTIFICATE DOES NOT MATCH THE HOST NAME SUPPLIED IN THE URL**. To resolve this warning, use the matching hostname in the URL or create a new server certificate that has the correct common name.

- **Local machine on which the Core Service is running does not trust the Certificate Authority that the server certificate is registered in.** To resolve this warning, export the Certificate Authority's certificate and then add it to the local host's Trusted Root Certification Authorities.

6. Lastly, verify that the Core Server can communicate to the remote machine by browsing to the previously mentioned WSConnector URL from the Core Server. Replace localhost with the remote machine's DNS name or IP address.

Configuring the Core Server and Performing Password Resets

For the Core Server to remotely reset a password for a target, you need to specify that the resets for that target should be performed remotely and where the resets for that target should be executed.

The PMM Target name that you use on the remote machine must match the name used on the Core Server.

Multiple remote servers may be specified for each target.

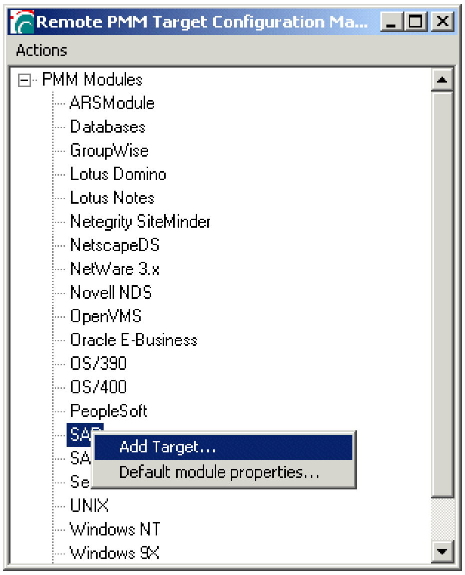
Remote PMM Target Configuration Manager

From the Start menu on the Core Server, select:

Programs>Core Access Assurance Suite>PasswordCourier Classic> Remote Target PMM Configuration Manager

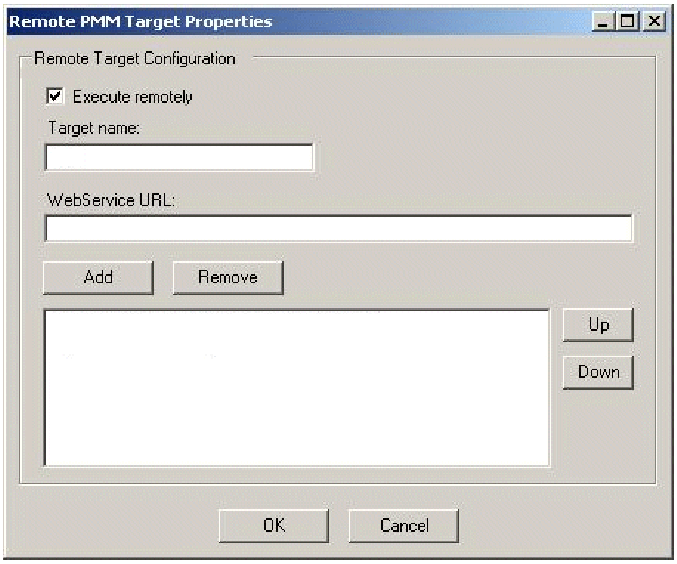
To add a target select a PMM and right mouse click to bring up the context menu, and click **ADD TARGET…**

Figure 55: Remote PMM Target Configuration



DEFAULT MODULE PROPERTIES… allows you to turn on remote execution of reset requests with undefined target names (the PMM target was not configured).

Figure 56: Remote PMM Target Properties



Enter a **TARGET NAME**. The target name must match the name you used when configuring the PMM on the remote server. Specify the **WEBSERVICE URL** for the remote server, then click **ADD** to add that URL to the list of URLs used to access the target. If you use multiple servers for remote reset requests, add an entry for each server.

The format for the WebService URL is:

`http://`*`DNS Name or IP Address`*`/CourRPM/wsconnector.asmx`

The Core Server accesses the remote servers in the order they are placed in the WebService URL list. A reset request is always routed to the first server in the list unless it is busy, in which case the request is routed to the second server. A request is only routed to the third server in the list if the first two servers are busy, and so on. If all servers in the list are busy, the Core Server waits until a server is free and routes the request to that server. To change the order of the servers in the list, highlight a server and click **UP** or **DOWN**.

To remove a server from the list, highlight the server and click **REMOVE**.

Clicking **OK** validates the URLs and saves the target. You can de-select the **EXECUTE REMOTELY** check box to revert a remote target to local, while saving the configuration.

Now, add the newly created target to the using the customization manager.

Remote System Configuration

If you did not previously create the actual PMM target on the remote system during the remote proxy server configuration, from the Start menu select

Programs> Core Access Assurance Suite>Password Management Modules

Select the module where you will create the target.

Performing Remote Resets

To the end user, remotely executed resets behave no differently than normal (locally processed) resets.

The log files for the remotely executed password reset modules (log files such as ntmodule.log) reside on the remote machine:

- The `wsconnector.log` file exists on both the remote system and the Core Server. It contains Web service related messages.

  Note: The following error is seen in the wsconnector.log, when a browser proxy is configured for the account used to run the Core Server (which by default is the local System):

  ```
  ”Exception occurred: The underlying connection was closed:
  Unable to connect to the remote server. Source: System Stack
  trace: at System.NET.HttpWebRequest.CheckFinalStatus()”
  ```

  If the account used to run the Core Server is configured to use a browser proxy, you need to add the following parameter to the Configuration Repository and give it a value of “1”.

  Key: \RemoteResetModule\DisableBrowserProxy

- The `cnctrinvoker.log` file is new and it exists on the remote machine. It contains messages related to invoking the reset module to perform the reset on the remote machine.

Chapter 7: Upgrading the Core Access Assurance Suite

This chapter describes how to upgrade to Access Assurance Suite 9.0. Some sections may not apply to your environment so read the introductions carefully. This chapter contains the following sections:

Before Upgrading

Core Security recommends exiting all running applications before upgrading the Core Access Assurance Suite.

Preserving Customized Macros

If you have customized dynamic communities, macros, STS application and want to use them after upgrading, do the following:

Export the customized configuration:

1. From the Start menu, go to Start > All Programs > Core Access Assurance Suite > Configuration Editor. This launches the Configuration Editor.
2. Navigate to CfgFile.db > Web Services > Dynamic Community STS > Dynamic Communities in the left pane.
3. Right-click Dynamic Communities node and export. Provide the name and location to store the file on export. For example: [CoreInstallPath]\ [DynamicCommunities].xml
4. Click EXPORT. If the export is successful, the message "Export is complete" is displayed in the Export Configuration Settings window.
5. Click CLOSE.
6. Navigate to CfgFile.db > Web Services > Macro Engine > Custom Macros in the left pane.
7. Right-click Custom Macros node and export. Provide the name and location to store the file on export. For example: [CoreInstallPath]\ [CustomMacro].xml
8. Click EXPORT. If the export is successful, the message "Export is complete" is displayed in the Export Configuration Settings window.
9. Click CLOSE.
10. Navigate to CfgFile.db > Web Services > Session STS > Applications in the left pane.
11. Right-click Applications node and export. Provide the name and location to store the file on export. For example: [CoreInstallPath]\ [STSApplications].xml
12. Click EXPORT. If the export is successful, the message "Export is complete" is displayed in the Export Configuration Settings window.
13. Click CLOSE.
14. Close the Configuration Editor.
15. Run the installer to upgrade to the latest version of the Access Assurance Suite as described in *"Upgrading the Access Assurance Suite" on page 115*. Then return here.
16. To preserve your customizations, follow these steps to import the customized configuration:
    a. Go to [CoreInstallPath]\ CourionServer\Baseline\CFG Import folder.

b. Compare the Dynamic Communities.xml file with the export taken before the upgrade.

c. Merge the required changes and save it to the new xml.

d. Repeat steps b and c for Custom Macros and SessionSTS-Applications XML’s.

e. From the Start menu, go to Start > All Programs > Core Security Access Assurance Suite > Configuration Editor. This launches the Configuration Editor.

f. Navigate to CfgFile.db > Web Services > Dynamic Community STS > Dynamic Communities in the left pane.

g. Right-click Dynamic Communities node and import. Provide the name and location of the Dynamic Communities xml file which is created on step b and c.

h. Navigate to CfgFile.db > Web Services > Macro Engine > Custom Macros in the left pane.

i. Right-click Custom Macros node and export. Provide the name and location of the Custom Macro xml file which is created on step d.

j. Navigate to CfgFile.db > Web Services > Session STS > Applications in the left pane.

k. Right-click Applications node and export. Provide the name and location of the Custom Macro xml file which is created on step d.

l. Close the Configuration Editor.

17. Restart the Courion Macro Engine and the Courion services.

Back Up of Default and Customized ASPs

During installation, all Core Access Assurance Suite web pages are saved to

[CoreInstallPath]\www_DoNotModify-*X.XX.XXX* (where *X.XX.XXX* is the version number)

When installation is complete, the files are copied to

[CoreInstallPath]\www

The default ASP files contained within the www_DoNotModify directory should not be modified because they serve as original backup files.

Always maintain a copy of the www folder containing customized ASP files and save that copy to another location on your server for safe keeping.

When you upgrade, the installer creates a backup copy of the current www directory before installing the www_DoNotModify-*X.XX.XXX* directory and overwriting the current www directory. The name of the backup folder is based upon the version of the suite installed when the upgrade install is started. For example, if you upgrade from 8.5 update 02, the backup directory is "www-8.05.000-Backup".

Note: The ASPs shipped with the Core Access Assurance Suite are version-specific and you cannot use them with previous or future versions of the Core Access Assurance Suite. Use the backup customized files as references when customizing the newer ASPs. Do not copy or paste code from an older ASP to a newer ASP (or vice versa), because the functionality within the ASP will fail.

File Paths in ASP Files

Once the ASP files are copied to the www directory, any ASP pages containing “#INCLUDE FILE=” statements with relative paths are modified to point to the virtual directory “Courion” that is created during installation. This change is required because the default configuration of IIS virtual directories does not allow parent paths. The installer logs all of these files changes.

For example, the statement

```
<!-- #INCLUDE FILE="../../WebSamples/AccessOptions/HTML/AccountCourier/
Common/CourACUtils.asp" -->
```

is changed to

```
<!-- #INCLUDE VIRTUAL="Courion/WebSamples/AccessOptions/HTML/
AccountCourier/Common/CourACUtils.asp" -->
```

These changes are made only to the files in the www directory, not the www_DoNotModify directory. Therefore, if you restore an ASP file by copying it from the www_DoNotModify directory to the www directory, you will need to change any #INCLUDE FILE= statements to #INCLUDE VIRTUAL= statements. Or, you can change the #INCLUDE FILE= statements from relative to absolute file paths.

Upgrading the Access Assurance Suite

Follow these steps to upgrade to the latest version of Core Access Assurance Suite 9.0 release from Access Assurance Suite 8.5:

Note: Upgrading to Core Access Assurance Suite 9.0 is only supported from 8.5 Update 04.

1. Log on as an Administrator to the system running the Core Server.
2. Back up the Content Repository folder from the location [Core-installation-folder]\CourionARMS before upgrading to the latest version.
3. Execute the CoreInstall.exe on an existing 8.5 Access Assurance Suite released installation. Follow on-screen instructions of the Installation wizard.
4. Required – Run the Configuration Manager located at Core Access Assurance Suite > Configuration Manager.
5. Optional - Copy/merge any updated files from the www_DoNotModify-8.5.000 folders (if necessary) to the corresponding directory in the Core virtual directory (for example www).
6. Optional – If you have made changes to the DCOM security configuration of the CourATLService, CourATLAdmin service, CourProtocolService, or CourProfiler service, these settings must be re-applied after the installation is complete.
7. After the installation is complete, update the database schema by performing the following steps:
    a. Copy the update SQL (Core 8.5.4 to 9.0 Upgrade.sql) files to the SQL server from this location [Core-installation-folder]\CourionService.
    b. Open the copied SQL file in the SQL Server Management Studio. Select SQLCMD mode. If the Courion database you use has the default database (DB) name “Courion,” run the script. If the database has a custom DB name, edit the following line in an editor before running the script:

    setvar DatabaseName "Courion" to :setvar DatabaseName "Custom DB"

    where “Custom DB” is the name of the custom database

    c. Run the SQL file.
    d. After the upgrade is complete, the new collectors are replaced with the new version under the CourionARMS\Content Repository. If you are using data collection, follow these steps:
        - Stop the Apache Tomcat service.
        - Go to the Manage Content page in the Core Access Assurance Portal.
        - Install all the collectors that are new. Refer to the Chapter 3: Setting Up Data Collection Rules in the manual, Configuring_the_Identity_Mapping_Solution.pdf in www/Docs folder, for information about how to do it.
        - Restart the Apache Tomcat service.

To upgrade your Provisioning Platform Applications and Classic Platform, do the following:

1. Log on as an Administrator to the system running the Core Server.

2. Launch the CoreInstall.exe file and follow instructions of the Installation wizard.
3. Update your database by running the SQL scripts (one at a time) against the current database.

   You must run the SQL scripts in the expected order. The expected order is from lower version to higher version. For example to upgrade from AAS 8.3 to 8.5, run “Courion 8.3 to 8.4 upgrade.sql”, then run “Courion 8.4 to 8.5 upgrade.sql”.

   Note: You must upgrade your database before running the Configuration Manager.
4. Run the Configuration Manager located at Core Access Assurance Suite > Configuration Manager. On the “Specify Random Salt Configuration” screen, specify the database connection parameters and salt configuration. For details, refer to *“Site-specific Information” on page 43*.

5. **Optional** - Copy/merge any updated files from the www_DoNotModify-A.B.xxx folders (A is the major release version, B is the update version, if necessary) to the corresponding directory in the Courion virtual directory (for example www).
6. **Optional** – If you have made changes to the DCOM security configuration of the CourATLService, CourATLAdmin service, CourProtocolService, or CourProfiler service, these settings must be re-applied after the installation is complete.
7. **Required** – To upgrade the Transaction Repository Schema (if you have a Transaction Repository target configured):
   a. Launch the Connector Configuration Manager.
   b. Locate the Connector for Microsoft ADO Transaction Repository target. Right-click on the target and select Modify Target.
   c. Select Next on the target configuration wizard.
   d. Re-enter the password for the configured Privileged user and click Finish.
   e. Close the Connector Configuration Manager, and select Yes when prompted to restart the Courion Services.

Chapter 8: Additional Configuration Options

This chapter includes the following sections which apply to certain types of environments:

- *“Configuring a Web Server” on page 118*
- *“Configuring Security of Web Access Option” on page 120*
- *“Multilanguage Support” on page 121*
- *“Disabling Client DNS Hostname Resolution when AAS is Hosted in a DMZ” on page 122*
- *“Configuring PasswordCourier Classic with BMC Remedy Action Request System” on page 123*
- *“Configuring PasswordCourier Classic with Clarify eFrontOffice” on page 126*
- *“Configuring PasswordCourier Classic with HP OpenView SCAuto” on page 128*
- *“Configuring PasswordCourier Classic with Peregrine Archway” on page 133*

Configuring a Web Server

In a single server installation, the Core Access Assurance Suite requires that a web server be installed on the same machine as the Core Server. In a distributed installation, a web server must be installed on the system in which the Publisher is installed.

There are two options for making the Core Access Assurance Suite web files, including the documentation, accessible to the end users and support staff:

- Copy the files to a location on the network that is reachable by the Intranet web server.

  or

- Use a virtual directory pointing to the files in the default install/Courion Corporation/www folder. The installation software attempts to do this for you, using the virtual directory name `Core`.

Choose the first option if you customize the files in any way. Always customize the copy, not the original, so that the files are protected if an update is applied or if the product is upgraded in the future.

If you choose the first option, copy the “www” folder of the Core Security folder to the correct location on the Intranet (typically, in the InetPub folder). Remember, each time a new version of Core Access Assurance Suite is installed, you need to copy this folder to the correct location.

For the second option, you can use the default directory Core, or create a different virtual directory pointing to the “www” folder of the Courion Corporation folder.

The Core Access Assurance Suite includes an administration console through which you can access most elements of the suite, including the customization managers. Use the web server to view the administration console file, `default.htm.`

Microsoft IIS (Internet Information Service)

Note: Installing IIS may cause ODBC driver mismatches with the supporting ODBC DLLs, requiring reinstallation of the ODBC components.

Note: The IIS option “Enable buffering” must be enabled to display Active Server Page files properly. By default, this option is enabled for Server Windows 2003 operating system. See the IIS documentation for more information.

IIS Timeout Value

Some large script-driven procedures such configuration migration may require more time than the IIS timeout value allows. If this happens, a message such as the following appears:

```
Active Server Pages error 'ASP 0113'
Script timed out
/core/utils/configmigration/ExportCfgData.asp
```

This error means that the IIS maximum amount of time for a script to execute was exceeded. You can change this limit by specifying a new value for the property Server.ScriptTimeout or by changing the value in the IIS administration tools.

Configuring Security of Web Access Option

By default, the Core Access Assurance Suite installation procedure configures the gLevelOfSecurity setting to 2 – the most secure level. This setting protects the product from any cross-frame, cross-domain scripting attacks. If you have placed the Core Access Assurance Suite web access option within a frame of your portal page, you need to lower the gLevelOfSecurity setting to 1 or 0 to continue using the product in a frame environment.

The gLevelOfSecurity settings have the following implications:

- 0 - Allows the application to run from within a frame environment. This is the lowest security setting and should not be used if the application is running on a publicly available network such as the Internet.
- 1 - Forces the application to break out of the frame environment and reload itself as the only web page in the browser window. This is a medium security setting that allows the application to continue functioning after it detects that it was embedded in a frame environment.
- 2 - Forces the application to stop processing and abort within a frame environment. This is the highest security setting available. Use this setting if the application is exposed in a hostile environment such as on the Internet. This setting protects the application from scripts that are capable of leveraging browser vulnerabilities to circumvent frame-breaking attempts.

This setting is in the following locations:

Table 4: Web Access Option Security Setting File Locations

Product	gLevelOfSecurity Setting
PasswordCourier Classic and PasswordCourier Support Staff Classic	CourUtils.js
Profile Courier Classic	PCUtils.js
PasswordCourier, ProfileCourier, AccountCourier and ComplianceCourier, RoleCourier	CourACClientUtils.asp

Multilanguage Support

The multilanguage support feature of the Core Access Assurance Suite provides support for languages other than American English, including French, German, UK English, and other languages. For information about this feature, refer to the *Core Access Assurance Suite Implementation Guide*.

Date Formats for the Access Request Manager, Access Certification, and Identity Mapping

To change the date formats in the Access Request Manager solution, the Access Certification solution, or the IdentityMapping solution, change the Culture option in IIS 7. Follow these steps:

1. Open the IIS Manager.
2. Navigate to the web site where you want to change the culture. For example: Sites>Default Web Sites>CourionARMS.
3. Open .Net Globalization to configure the appropriate Culture tag. Examples: en-US or en-GB.

Add the globalization tag to the web.config file in Portal Frameworks manually.

Note: The web.config file needs to be write-enabled to add the tag.

Disabling Client DNS Hostname Resolution when AAS is Hosted in a DMZ

If client systems that are connecting to a CourATLService instance do not have DNS entries on the system (such as when the CourATLService is hosted in a DMZ), connections attempt to resolve but fail after a timeout of several seconds.

You can prevent this issue by using a command line parameter that always disables client DNS hostname resolution: -BypassClientDNSResolution. After this parameter is set, the %CLIENT_HOST_NAME% macro always resolves to an empty string.

To enable this parameter and disable client DNS hostname resolution, follow these steps:

1. Go to **START** > **RUN** > **REGEDIT** to open the Registry Editor.
2. Navigate to KEY > HKEY_LOCAL_MACHINE\Software\CourionCorporation\Courion Enterprise Provisioning Suite.
3. Select **EDIT** > **NEW** > **STRING VALUE** to create a new string value.
4. Set the string value name to CourATLCommandLine.
5. Set the value of the registry setting to -BypassClientDNSResolution.
6. Save the change.
7. Stop the CourATLService.
8. Go to **START** > **RUN** > **CMD** to open a command window.
9. Navigate to the [CoreInstallPath]\CourionService folder.
10. Run the following command: CourATLService.exe -Service
11. Start the CourATLService.

Configuring PasswordCourier Classic with BMC Remedy Action Request System

The Core Access Assurance Suite requires no changes to the Remedy® databases. However, Core Security recommends one fixed license for the Core Server. A floating license might time out. If a timeout occurs, the Core Server cannot update a trouble ticket with password reset status.

Configuring the Core Server for BMC Remedy Action Request System

If your access keys include a key for Remedy Action Request System, the installation software displays the Remedy ARS Help Desk Administration dialog box. Use this to configure the Core Server for the Remedy Action Request System.

Note: For provisioning applications to work properly with Remedy Action Request System, you must configure the corresponding connector. See the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents* for more information.

To configure the Core Access Assurance Suite to work with the Remedy Action Request System:

1. Start>All Programs>Core Access Assurance Suite>Configuration Manager

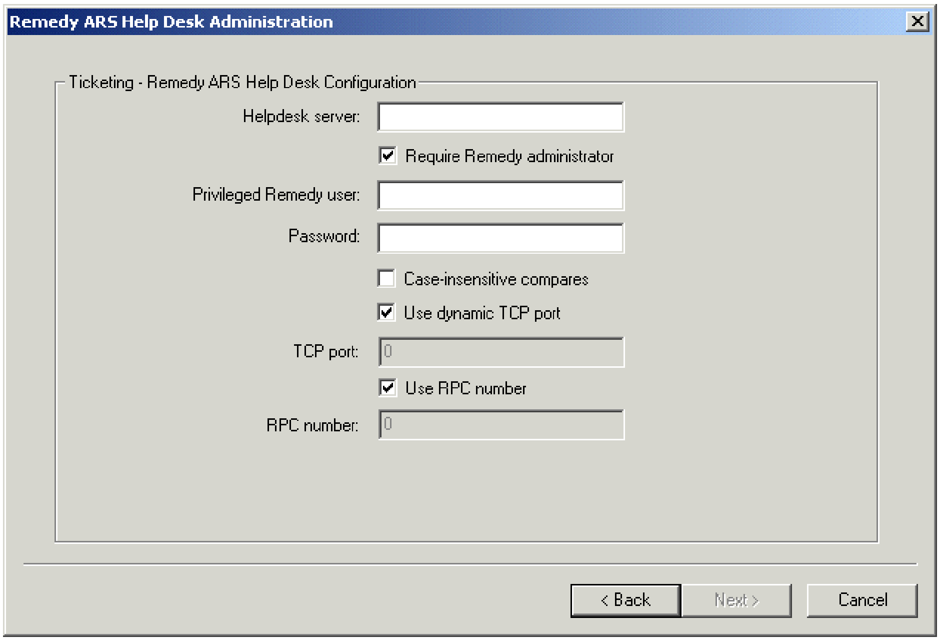
   The Core Server Configuration Manager launches.

2. Proceed as described in *“Running the Core Server Configuration Manager” on page 42*, and complete or default the dialog boxes as appropriate until you reach the Data Source Selection

3. Select or default the Profile data source, and then select **REMEDY** as the Ticketing data source.

4. Click the **NEXT** button, to configure Remedy as a ticketing data source.

Figure 57: Remedy ARS Help Desk Administration


Remedy ARS Help Desk Administration
Ticketing - Remedy ARS Help Desk Configuration
< Back
Next >
Cancel

Helpdesk server:

☑ Require Remedy administrator

Privileged Remedy user:

Password:

☐ Case-insensitive compares

☑ Use dynamic TCP port

TCP port: 0

☑ Use RPC number

RPC number: 0

1. **HELPDESK SERVER** — Enter the name of the server running the Action Request System.
2. **REQUIRE REMEDY ADMINISTRATOR** — Determine if the privileged username (Remedy Administrator) should be used. If so, make sure this box is checked. If not, clear the box and make sure the username provided has the necessary access to the schemas or forms needed for user information, if configuring a profile data source, and Help Desk trouble tickets, if configuring a ticketing data source.
3. **PRIVILEGED REMEDY USER** — Enter a the username of a privileged Remedy user.
4. **PASSWORD** — Enter the password for the privileged username.
5. **CASE-INSENSITIVE COMPARES** — Determine whether or not to use case insensitive comparisons for user validation and authentication. Check the box if you want case-insensitive comparisons. This allows the user to enter “smith” or “SMITH” for successful validation of the value “Smith” which is stored in the database. If this box is not checked, then the case-sensitivity of the underlying database is used for the user validation and authentication comparisons. To be validated/ authenticated, the user would need to enter “Smith” exactly as it appears in the database.
    - Microsoft SQL Server is case-insensitive by default. This is determined during installation of the SQL Server.
    - Sybase case-sensitivity is determined during installation and configuration of language and character set of the Sybase Server.
    - Oracle and IBM DB2 are case-sensitive regardless of platform.
6. **USE DYNAMIC TCP PORT** — Select this option if the Remedy server is configured to use the Port Mapper feature. This option specifies that the Core Server uses an available TCP port on the Remedy server.
7. **TCP PORT** — If you do not check **USE DYNAMIC TCP PORT**. Enter the port number on the Remedy AR System server.
8. **USE RPC NUMBER** — Select option to specify that the Core Server uses an available Remote Procedure Call (RPC) port on the Remedy server.
9. **RPC NUMBER** — Enter a specific RPC port number in the text box.

   Note: The RPC number specifies which thread to use within the Remedy AR System server. The administrator on the Remedy server can assign a thread to a specific RPC number. The threads can be configured to perform certain functions and to be able to handle a certain number of connections. This feature helps load balance the Remedy server.

   For example, you can send all tickets from PasswordCourier to one thread and all tickets from AccountCourier to another thread. This helps balance the load on the Remedy server. The thread for PasswordCourier may be able to handle ten connections, while the thread for AccountCourier may only need to handle two.
10. After completing the steps above, click the **NEXT** button to conclude Core Server configuration.

Notes and Warnings

The Core Server does not handle keywords such as default values for fields ($USER$ for example). If you intend to use this field in any customization manager for trouble ticket creation, you must supply a value for the field.

Diary fields cannot be used for user validation.

Remedy text fields allow a maximum of 255 characters. This should be kept in mind when configuring the Password Management Module for Synchronization so that text messages are not cut off.

Configuring PasswordCourier Classic with Clarify eFrontOffice

The Clarify® help desk configuration allows the creation, update, and closing of tickets in Clarify eFrontOffice. The Microsoft SQL client must be installed on all machines running the Clarify client.

To configure the Core Access Assurance Suite with Clarify eFrontOffice, you must direct the Core Server to use Clarify as a ticketing data source.

Note: The Clarify client and the Microsoft SQL client must be installed on the Core Server machine.

To configure the Core Access Assurance Suite to work with Clarify:

1. Select Start > Programs > Courion > Core Server>Configuration Manager.

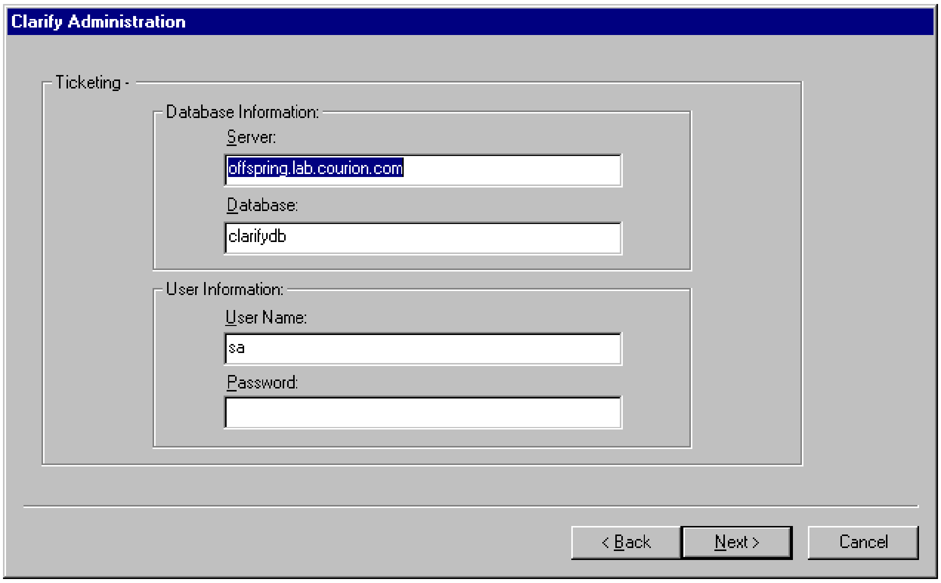
   The Core Server Configuration Manager launches.

2. Proceed as described in *"Running the Core Server Configuration Manager" on page 42*, and complete or default the dialog boxes as appropriate until you reach the Data Source Selection

3. Select or default the Profile data source, and then select **CLARIFY** as the Ticketing data source.

4. Click the **NEXT** button, configure the profile data source if needed, and then click the **NEXT** button again to configure Clarify as a ticketing data source. The Clarify Administration panel is displayed, *Figure 58*.

Figure 58: Clarify Administration



Clarify Administration

Ticketing -

Database Information:

Server: offspring.lab.courion.com

Database: clarifydb

User Information:

User Name: sa

Password:

5. Type the name of the Clarify database server into the **SERVER** field.

6. Type the name of the Clarify database into the **DATABASE** field.

7. Type the username of the Clarify database administrator into the **USER NAME** field.

   If you change the password for the Clarify eFrontOffice administrator specified during Core Access Assurance Suite installation, then you must update the server configuration.

8. Type the password for the administrator user name entered above into the **PASSWORD** field.
9. Click the **NEXT** button and continue to configure the Core Server as appropriate.

Note: Ticket numbering is handled through the automatic numbering facility used within Clarify.

Note: When using Clarify for Ticketing and entering data for the contact FirstName, LastName and Phone Number fields, the data used must be the same as what is in the database for ticket creation. If any other data is used, the ticket is not created and the password reset fails, generating an error.

Configuring PasswordCourier Classic with HP OpenView SCAuto

The DLL for SCAutomate™ is included in the installation of this software. The agent used is part of the SCAutomate™ product.

Note: The Connector for HP OpenView ServiceCenter (previously known as Peregrine ServiceCenter) requires a license key from HP called `SCAuto\SDK`. Contact HP for information about how to obtain this license key.

Note: If upgrading to Core Access Assurance Suite from an earlier version, the SCAutomate configuration GUI displays the configuration files currently in use by default. Once a ServiceCenter version is selected, the configuration files displayed change to that of the version selected, or the fields are blank if **CUSTOM** is selected. Prior to making any selection on the SCAutomate configuration GUI, write down the configuration files currently in use. Then, if these files are used again, select **CUSTOM** and type the names of these files back into their respective fields in the interface.

Configuration Integration

Provisioning applications use a connector to integrate with HP OpenView ServiceCenter. See the manual *Configuring Password Management Modules (PMMs), Connectors, and Agents* for more information.

1. First, edit the file `OS-installation-drive\winnt\system32\drivers\etc\services` and add:

   ```
   scauto [portnum]/tcp
   scauto [portnum]/udp
   ```

   where [`portnum`] is the TCP/IP port number on which ServiceCenter is listening.

   Configure the port number entries on the ServiceCenter server according to HP OpenView ServiceCenter documentation.

2. From the Start menu, select

   All Programs>Core Access Assurance Suite>Configuration Manager

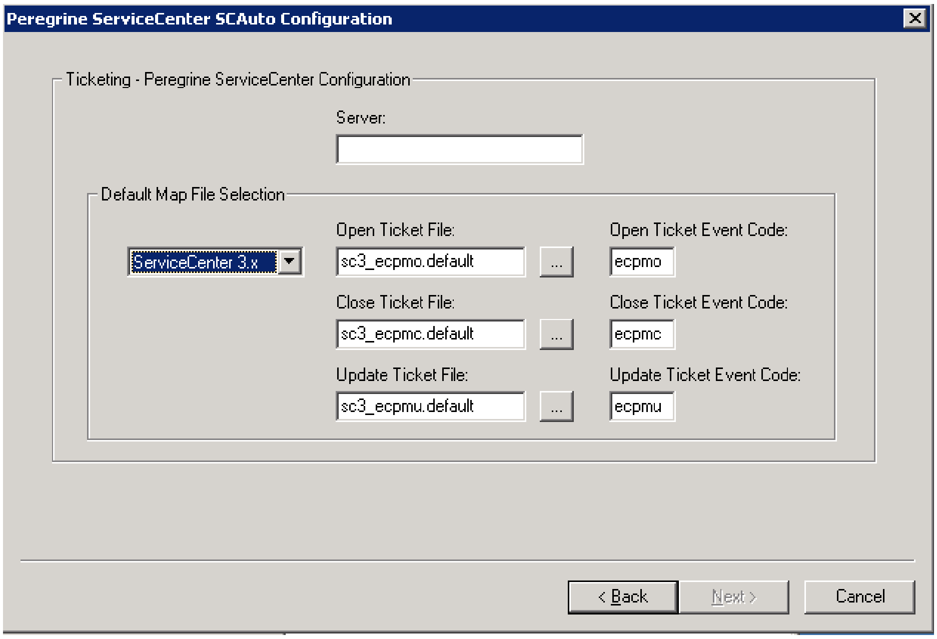
   The Core Server Configuration Manager launches.

3. Proceed as described in *"Running the Core Server Configuration Manager" on page 42*, and complete or default the dialog boxes as appropriate until you reach the Data Source Selection dialog box.

4. Select the appropriate Profile data source, and then select **PEREGRINE SCAUTO** as the ticketing data source

   The HP OpenView ServiceCenter Configuration dialog box appears (*Figure 60* next).

Figure 59: HP Open View (Peregrine) ServiceCenter Configuration



Peregrine ServiceCenter SCAuto Configuration

Ticketing - Peregrine ServiceCenter Configuration

Server:

Default Map File Selection

ServiceCenter 3.x

Open Ticket File: sc3_ecpmo.default

Open Ticket Event Code: ecpmo

Close Ticket File: sc3_ecpmc.default

Close Ticket Event Code: ecpmc

Update Ticket File: sc3_ecpmu.default

Update Ticket Event Code: ecpmu

5. In the **SERVER** field, enter the name of the server running HP OpenView ServiceCenter.
6. For **DEFAULT MAP SELECTION**, select the correct default map.

   The Configuration Manager chooses Default Map File based on the **DEFAULT MAP SELECTION** chosen.

To map PMO, PMU and PMC files:

1. Log into your HP OpenView ServiceCenter as a HP OpenView administrator.
2. Click the Utilities tab and then click the **EVENT SERVICES** icon.
3. Click the Administration tab and click **MAPS**.
4. On the Event Map dialog box: find the field labeled **MAP NAME** and type “problem open” (without the quotes). This field is case-sensitive; “problem open” must be lowercase.
5. Next, find the field labeled **TYPE** and select **INPUT** from the drop-down list.
6. Click the **SEARCH** button.
7. From the menu bar, click **LIST OPTIONS** and select **EXPORT TO TEXT FILE**.
8. Choose **COMMA** as the delimiter, specify the folder and filename, and then click the button with the green check mark.

   The Database menu is displayed.
9. Select **OK**.
10. Repeat steps 4-9 using “problem close” and “problem update” in place of “problem open” (all entries must be lowercase).

Note: Use the character set recommended by ServiceCenter. Some characters, notably the question mark (?), ampersand (&), caret (^), and vertical bar (|), may not produce the expected result. For example, using a vertical bar to separate data in a non-array field causes all data after the first bar to be lost.

HP OpenView ServiceCenter Ticketing

Integrating the Core Server with HP OpenView ServiceCenter allows Core Access Assurance Suite applications to create and update tickets in ServiceCenter. While integrating the Core Server with ServiceCenter supports ticketing, it does not support authentication. When configuring the Core Server for use with ServiceCenter, you must configure the authentication data source to point to the authentication database. This may or may not be the underlying database in the HP OpenView implementation.

When Core Access Assurance Suite applications create or update a ticket in ServiceCenter, they submit the request in a standard configuration through the ServiceCenter auto-ticket submission interface. The ServiceCenter auto-ticket interface processes the request by interpreting the event codes and data contained within that request.

The event codes define how data is mapped into the HP OpenView database fields. Core Security supports the default event codes:

- ecpmo (open ticket)
- ecpmu (update ticket)
- ecpmc (close ticket)

These codes are mapped to their respective maps:

- **PROBLEM OPEN**
- **PROBLEM UPDATE**
- **PROBLEM CLOSE**

which indicate the action to be performed.

Core Security provides support for users who wish to modify the existing event codes and their respective maps and/or add additional fields. However, Core Security does not support user-created events mapped to user-created maps.

If the ticket is opened on a start action, it may be updated on a success action. When tickets are updated, the user has the option of using one of two maps: **PROBLEM UPDATE** and **PROBLEM CLOSE**. The former leaves the ticket in the open state while the latter closes it. Both update the ticket.

If ServiceCenter has been previously customized to use specific fields, this customization can be leveraged by the Core Access Assurance Suite.

For information on parameters that let you specify a custom ticket prefix or retain (instead of deleting) the output queue, consult Core Security Customer Support.

SCAutoTicketing Update

The PeregrinApi.dll file checks to see if the ticket ID it receives from HP OpenView is actually a valid id before deleting that event from the output queue. Three configuration parameters exist to configure this feature. You need to add them to the configuration file. All three parameters share the base key of:

SYSTEM\CURRENTCONTROLSET\SERVICES\COURIONPASSWORDCOURIER\Server\Ticketing

These are the names of the keys:

- **PreUpdate17** — This is a "Y" or "N" field. If this key is "N" or does not exist, then it uses the new code to double check the ticket ID. If you want HP OpenView ticketing to function as it did in previous versions, use "Y" as the value.
- **CustomTicketPrefix** — Enter in a text value for this key. The defaults in the PeregineAPI are: IM and CALL. The HP OpenViewAPI checks the defaults and this Prefix when determining a valid ticket.

  Note: If you enable the PreUpdate17 key, this key is not checked even if you have configured values for it.
- **DoNotPurgeOutputQueue** — Enter "Y" if you do not want to purge the output queue. Enter "N" or do not enter a value to purge it. No value is the default, meaning the output queue is purged.

Data Types with HP OpenView ServiceCenter

Core Security supports number, character, logical (true/false), and array data types in maps for ServiceCenter. In the list of fields for a table, all fields that are arrays appear multiple times. After the first appearance of the array, the name of the array is followed by an underscore and a number.

Because arrays are supported by working with the first occurrence of the array field name, arrays should be populated by separating each element with a pipe, “|” (for example, `This is data for element1 | This is data for element2 | This is data for element3...`). The other occurrences of the array field name should be ignored and should read “settable.” When using the default maps for pmo, pmu and pmc provided by HP OpenView, the array fields are:

- pmc: resolution, resolution_1, resolution_2
- pmu: update.action, update.action_1, update.action_2
- pmo: $ax.field.name, action_2, action_3

The following are required to configure ticketing for HP OpenView ServiceCenter:

- Tickets need at least one field populated.
- The **NETWORK.NAME** field must not be altered from **SETTABLE** to open, close, or update tickets.
- The **NUMBER** field must not be altered from **SETTABLE** to close and update tickets.
- Configure the HP OpenView ServiceCenter data source.
- Specify the full path name for the map files (labeled **TICKET FILES** in the Core Server configuration window). The ellipse (**...**) buttons open a window to browse the files and enter the full path name.

In Request Tracking, the ability to define the value for a field exists by clicking on the button labeled **DEFINE 'FIELDNAME'**.

Exporting Map Files in HP OpenView ServiceCenter

If an error message appears during export of map files in HP OpenView ServiceCenter, you must configure an additional parameter. The steps required depend on whether or not the Record list is active. To determine if the Record list is active, view the drop-down menu. If there is a check next to "Record list," then it is turned on.

If Record list is turned off:

1. From the main menu, under the Toolkit tab, select **DATABASE MANAGER**.
2. Under **FORM**, type "displayoption" and press **ENTER**. A "displayoption.g" format is displayed.
3. Under Screen ID, type "QBE.display.qbe.gui" and press **ENTER**.
4. Scroll down the list until the option for **EXPORT TO TEXT FILE** is visible. Double-click that option.
5. Under **NAMES/VALUES** there is **RECORD/$L.FILED** in the first element array. Enter "name/$L.dd.format" to the second array.
6. Save the record.

If Record list is turned on:

1. From the main menu, under the Toolkit tab, select **DATABASE MANAGER**.
2. Under **FORM**, type in "displayoption" and press **ENTER**.
3. A "displayoption.g" format is displayed.
4. Under Screen ID, type in "QBE.display.gui" and press **ENTER**.
5. Remove **NAME IN** from "name in $L.dd.qbe.format" and press **ENTER**.

Notes and Warnings

- Output handling of HP OpenView ServiceCenter tickets is now more robust. All events are processed and then deleted in a batch.
- HP OpenView ServiceCenter administrators should index the username field of output events. Indexing the usernames lets the Core Server search and function more efficiently.

Configuring PasswordCourier Classic with Peregrine Archway

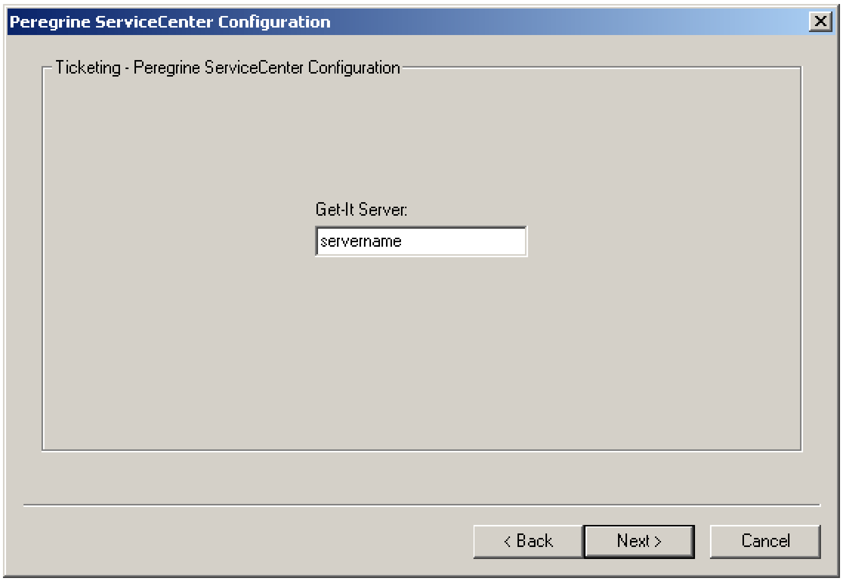
To configure PasswordCourier Classic with ServiceCenter using Peregrine Archway, you must tell the Core Server to use Archway as a ticketing data source. To configure the Core Server:

1. From the Start menu, select

   All Programs>Core Access Assurance Suite>Configuration Manager

   The Core Server Configuration Manager launches.
2. Proceed as described in *“Running the Core Server Configuration Manager” on page 42*, and complete or default the dialog boxes as appropriate until you reach the Data Source Selection dialog box.
3. Select or default the appropriate Profile data source, and then select **PEREGRINE ARCHWAY** as the Ticketing data source.
4. Click the **NEXT** button to configure Archway as a ticketing data source. The Peregrine ServiceCenter Configuration dialog box is displayed (*Figure 60*).

Figure 60: Peregrine ServiceCenter Configuration



Peregrine ServiceCenter Configuration

Ticketing - Peregrine ServiceCenter Configuration

Get-It Server: servername

5. Type the name of the Get-It server into the Get-It Server field.
6. Click the “Next” button and continue to configure the Core Server as appropriate.

Note: Use the character set recommended by ServiceCenter. Some characters, notably the question mark (?), ampersand (&), caret (^), and vertical bar (|), may not produce the expected result. For example, using a vertical bar to separate data in a non-array field causes all data after the first bar to be lost.

Chapter 9: Problem Reports

This chapter explains how to contact Core Security Customer Support:

Core Security Support Contacts

Core Security is committed to facilitating the work and overall productivity of its customers. To that end, Core Security's self-service products are designed to be easy to use. Please contact Core Security immediately if you encounter any problems while using an Access Assurance Suite product. Support phone lines are staffed from 8am-6pm Eastern Standard Time. Problem reports may also be submitted by e-mail or online at https://support.coresecurity.com.

Table 5 lists the addresses and telephone numbers you can use.

Table 5: Addresses for Submitting Problem Reports

Description	Address
e-Mail	support@coresecurity.com
World Wide Web	https://support.coresecurity.com
Phone (voice)	(678) 304-4500 1-(866)-268-7466 (domestic toll-free)
Mail	1000 Holcomb Woods Parkway, Suite 401 Roswell, GA 30076

Submitting a Problem Report

Please include and be prepared to discuss the following files when submitting a problem report:

- all logs, including server logs

Note: In the installation, these files (log, configuration, and other output files) are found in the default folder \Program Files x86\Courion Corporation\CourionService.

If you have a distributed installation, you may need to retrieve the logs from multiple servers.

In addition to those files, the following files might also help in diagnosing the problem. The files required are dependent on the type of problem that occurs. For example:

- Type of problem: PMM for OS/390® won't work

  Files required: PMM for OS/390 system console log, PMM SAMPLIB GLOBAL MEMBER
- Type of problem: PMM for UNIX® won't work

  File required: pxy_def file from UNIX machine
- Type of problem: Java technology-enabled applets refuse to load or behave strangely

  File required: Java console log from web browser
- Type of problem: Application crashes

  File required: Dr. Watson log file (or log file from equivalent debugger)
- Type of problem: Core Server refuses to start

  File required: Application event log from the Windows® Event Viewer

Start Menu Options for Problem Diagnosis

When you contact Customer Support, you may be asked to use the following options, available from the Start menu, to diagnose a problem:

- The **Configuration Editor** (CFGEdit2) — Run as administrator to access the Configuration Editor from the Start menu, enter:

  Start>All Programs>Core Access Assurance Suite>Configuration Editor

  When you do, the following message appears:

  WARNING! MAKING CHANGES TO YOUR CONFIGURATION CAN CAUSE THE ACCESS ASSURANCE SUITE TO STOP FUNCTIONING.

  PLEASE BACK UP YOUR CONFIGURATION REPOSITORY BEFORE MAKING ANY CHANGES.

  Core Security recommends that you do not use the Configuration Editor unless Customer Support has instructed you to do so.

- The **Courion Version Information** option — To view information about the version of the Core Server that you are running and the access keys you are using, from the Start menu, enter:

  Start>All Programs>Core>Access Assurance Suite>Version Information

Appendix A: Running the Connector Framework in Command-Line Mode

This appendix describes how to run a self-hosted Connector Framework (CF) in command-line mode. In command-line mode, all connector actions run by the CF interact with your personal desktop. You can use command-line mode to observe and debug screen scraping connectors.

Note: The CF must be self-hosted to run in command-line mode.

To run the CF in command-line mode, you need to stop it, then run the CF from the command line. This appendix describes how to do this in the following sections:

- *“Stopping the Connector Framework” on page 140*
- *“Running the Connector Framework from a Command Line” on page 141*

Note: If you change the configuration of any connectors or targets using the Connector Configuration Manager while the CF is in command-line mode, you need to manually restart the CF for the changes to take effect.

Stopping the Connector Framework

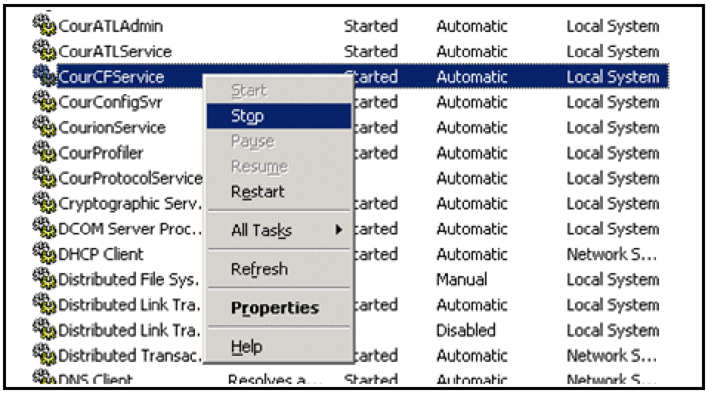
This section describes how to stop a self-hosted connector framework.

To stop a self-hosted CF, follow these steps:

1. From the Start menu, select:

   Start>Administrative Tools>Services
2. Right click on CourCFService and click Stop, as shown in the following figure:

Figure 1: Stopping CourCFService



CourATLAdmin		Started	Automatic	Local System
CourATLService		Started	Automatic	Local System
CourCFService		tarted	Automatic	Local System
CourConfigSvr		tarted	Automatic	Local System
CourionService		tarted	Automatic	Local System
CourProfiler		tarted	Automatic	Local System
CourProtocolService			Automatic	Local System
Cryptographic Serv.		tarted	Automatic	Local System
DCOM Server Proc..		tarted	Automatic	Local System
DHCP Client		tarted	Automatic	Network S...
Distributed File Sys.			Manual	Local System
Distributed Link Tra.		tarted	Automatic	Local System
Distributed Link Tra.			Disabled	Local System
Distributed Transac.		tarted	Automatic	Network S...
DNS Client	Resolves a...	Started	Automatic	Network S...

Running the Connector Framework from a Command Line

After you stop the Connector Framework, open a command prompt. To do this, navigate to the directory where you installed the CourionService, then the WS, and Bin directories. The full path looks similar to the following:

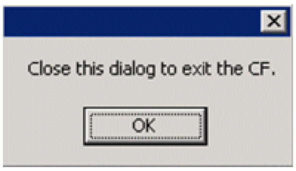
```
C:\Program Files (x86)\Courion Corporation\CourionService\WS\Bin>
```

You can now run the CourCFWindowsService.exe with the command line option of -debug:

```
C:\Program Files (x86)\Courion
Corporation\CourionService\WS\Bin>CourionService.
exe -debug
```

When it has fully started, the following dialog box appears:

Figure 2: Closing Command-Line Mode



Close this dialog to exit the CF.

At this point, all actions that run through this CF fully interact with your desktop. To exit command-line mode, click **OK** on this dialog box and the CF in command-line mode stops.

If you have finished and want to run the CF normally, you need to restart the CF: restart the CourCFService in the Administrative Services tool.

Index