Contents

About This Book .......................................................................................................................... vii

CHAPTER 1 Overview .................................................................................................................... 1
  What is Unified Agent?................................................................................................................. 1
  Services provided......................................................................................................................... 1
  Agent Management Console....................................................................................................... 2

CHAPTER 2 Installing and Configuring Unified Agent and Agent Management Console ..................................................................................................................... 3
  Pre-installation tasks ................................................................................................................... 3
  Installing Unified Agent ............................................................................................................. 4
    Starting the JINI server............................................................................................................ 4
    Starting and stopping Unified Agent..................................................................................... 4
    Starting Unified Agent as a service on UNIX platforms....................................................... 5
    Starting Unified Agent as a service on Windows platforms.................................................. 5
    Installing the Agent Management Console......................................................................... 6
  Configuring Unified Agent ....................................................................................................... 7
    Post-configuration tasks........................................................................................................ 7

CHAPTER 3 Security ..................................................................................................................... 9
  Unified Agent security .............................................................................................................. 9
  Encryption ................................................................................................................................. 9
  Authentication.......................................................................................................................... 10
  Authorization .......................................................................................................................... 12
  Role mappings ....................................................................................................................... 12
  Permissions ............................................................................................................................. 13

CHAPTER 4 Agent Management Console Features and Functionality ........................................ 15
  Managing look-up services ....................................................................................................... 15
  Creating a network lookup profile........................................................................................... 16
  Discovering Unified Agents ..................................................................................................... 16
  Deleting a network look-up profile ........................................................................................ 17
Contents

Changing a network look-up profile......................................................... 17
Specifying a network plug-in name filter................................................. 17
Managing Unified aAgents ...................................................................... 17
  Manually adding an agent...................................................................... 17
  Connecting to an agent...................................................................... 18
  Adding credentials............................................................................. 18
  Disconnecting from an agent.............................................................. 19
  Restarting an agent .......................................................................... 19
  Retrieving the agent log .................................................................... 19
  Viewing and changing agent properties .......................................... 19
Managing discovery adaptors................................................................. 20
  Creating a discovery adaptor.............................................................. 20
  Deleting a discovery adaptor.............................................................. 21
  Viewing and changing discovery adaptor properties ....................... 21
  Setting discovery adaptor password ................................................. 22
Managing security modules ..................................................................... 22
  Creating a security module................................................................. 22
  Deleting a security module................................................................. 23
  Ordering the security module authentication ..................................... 23
  Viewing and changing security module properties ............................ 23
  Setting a security module password .................................................. 24
Managing role mappings........................................................................ 24
  Creating a role mapping.................................................................... 24
Managing agent services......................................................................... 25
  Starting an agent service .................................................................. 25
  Restarting an agent service................................................................. 26
  Stopping an agent service .................................................................. 26
  Viewing agent service properties ....................................................... 26
Managing agent plug-ins......................................................................... 26
  Loading an agent plug-in .................................................................. 27
  Unloading an agent plug-in ................................................................. 27
  Reloading an agent plug-in ................................................................. 27
  Viewing agent plug-in properties ....................................................... 27
  Server-based plug-ins ....................................................................... 28
Managing attributes.................................................................................. 29
  Viewing attributes ............................................................................. 29
  Updating attribute values ................................................................... 30
  Viewing properties and changing permissions ................................... 30
Managing operations.................................................................................. 31
  Viewing operations ........................................................................... 31

APPENDIX A

Password Encryption and Security Configuration................................. 33
  Password encryption utility.................................................................. 33
  Enabling Rivest-Shamir-Adleman (RSA) encryption ............................ 34
Adaptive Server agent plug-in server password encryption ........... 35
Pluggable Authentication Module (PAM) configuration .............. 35

APPENDIX B

SNMP Agent Plug-in................................................................. 37
Overview ................................................................................. 37
Terminology ............................................................................. 37
Sybase Adaptive Server MIB.................................................... 38
Unified Agent SNMP services and plug-ins................................. 39
  UAF SNMP Service ............................................................... 39
  UAF SNMP Service MIBs ....................................................... 40
Adaptive Server-to-SNMP client architecture ............................ 41
Security ..................................................................................... 41
  Security for connections from Adaptive Server ..................... 41
  Security for connections from SNMP .................................. 42
Configuring Adaptive Server for the SNMP Agent
  Plug-in ................................................................................... 43
TDS services ............................................................................. 43
UAF SNMP Agent Service ....................................................... 44
Adaptive Server runtime .......................................................... 45
Using the Adaptive Server SNMP Agent Plug-in ....................... 47
  Viewing the SNMP information from NET-SNMP ................. 48
Sybase MIB .............................................................................. 48

APPENDIX C

Tabular Data Stream (TDS) Service.......................................... 55
What is TDS? ............................................................................. 55
Using the TDS service .............................................................. 56
UAF services ........................................................................... 57
  uaf_list_service_operations ................................................. 58
  uaf_list_services ................................................................. 59
  uaf_list_service_operation_arguments ............................... 60
  uaf_invoke_service_operation ............................................ 61
  uaf_list_service_attributes ................................................ 61
  uaf_get_service_attribute .................................................. 62
  uaf_set_service_attribute .................................................. 62
UAF plug-ins ........................................................................... 62
  uaf_list_plugins ................................................................. 63
  uaf_list_plugin_operations ............................................... 63
  uaf_list_plugin_operation_arguments ............................... 64
  uaf_invoke_plugin_operation .............................................. 65
  uaf_list_plugin_attributes ................................................ 65
  uaf_get_plugin_attribute .................................................... 66
  uaf_set_plugin_attribute .................................................... 66
TDS listener client examples .................................................. 67
Setting up examples for the Adaptive Server client ................. 67
Examples for configuring the Adaptive Server client ............. 68
Examples for using the Adaptive Server client ...................... 68
Examples for configuring the Adaptive Server client ......... 70
Reporting functionality and error recovery ................. 70

Index................................................................................................................. 71
About This Book

Audience

The *Unified Agent and Agent Management Console User's Guide* is intended as a reference tool for Sybase™ database administrators.

How to use this book

- Chapter 1, “Overview” – provides an description of the Unified Agent.
- Chapter 2, “Installing and Configuring Unified Agent and Agent Management Console” – provides information about installing and configuring Unified Agent and the Agent Management Console.
- Chapter 4, “Agent Management Console Features and Functionality” – provides information about Agent Management Console features and functionality
- Appendix A, “Password Encryption and Security Configuration” – provides information about encrypting passwords and configuring security.
- Appendix B, “SNMP Agent Plug-in” – provides information about using the plug-in to configure and use the Unified Agent.

Related documents

The Adaptive Server® Enterprise documentation set consists of the following:

- The release bulletin for your platform – contains last-minute information that was too late to be included in the books.
  A more recent version of the release bulletin may be available on the World Wide Web. To check for critical product or document information that was added after the release of the product CD, use the Sybase Technical Library.
- The *Installation Guide* for your platform – describes installation, upgrade, and configuration procedures for all Adaptive Server and related Sybase products.
• **What's New in Adaptive Server Enterprise?** – describes the new features in Adaptive Server version 15.0, the system changes added to support those features, and changes that may affect your existing applications.

• **ASE Replicator User’s Guide** – describes how to use the Adaptive Server Replicator feature of Adaptive Server to implement basic replication from a primary server to one or more remote Adaptive Servers.

• **Component Integration Services User’s Guide** – explains how to use the Adaptive Server Component Integration Services feature to connect remote Sybase and non-Sybase databases.

• The **Configuration Guide** for your platform – provides instructions for performing specific configuration tasks for Adaptive Server.

• **Enhanced Full-Text Search Specialty Data Store User’s Guide** – describes how to use the Full-Text Search feature with Verity to search Adaptive Server Enterprise data.

• **Glossary** – defines technical terms used in the Adaptive Server documentation.


• **Java in Adaptive Server Enterprise** – describes how to install and use Java classes as datatypes, functions, and stored procedures in the Adaptive Server database.

• **Job Scheduler User's Guide** – provides instructions on how to install and configure, and create and schedule jobs on a local or remote Adaptive Server using the command line or a graphical user interface (GUI).

• **Messaging Service User's Guide** – describes how to use Real Time Messaging Services to integrate TIBCO Java Message Service and IBM WebSphere MQ messaging services with all Adaptive Server database applications.

• **Monitor Client Library Programmer's Guide** – describes how to write Monitor Client Library applications that access Adaptive Server performance data.


• **Performance and Tuning Series** – a series of books that explain how to tune Adaptive Server for maximum performance:
About This Book

- **Basics** – the basics for understanding and investigating performance questions in Adaptive Server.

- **Locking and Concurrency Control** – describes how the various locking schemas can be used for improving performance in Adaptive Server, and how to select indexes to minimize concurrency.

- **Query Processing and Abstract Plans** – describes how the optimizer processes queries and how abstract plans can be used to change some of the optimizer plans.

- **Physical Database Tuning** – describes how to manage physical data placement, space allocated for data, and the temporary databases.

- **Monitoring Adaptive Server with sp_sysmon** – describes how to monitor Adaptive Server’s performance with `sp_sysmon`.

- **Improving Performance with Statistical Analysis** – describes how Adaptive Server stores and displays statistics, and how to use the `set statistics` command to analyze server statistics.

- **Using the Monitoring Tables** – describes how to query Adaptive Server’s monitoring tables for statistical and diagnostic information.

- **Quick Reference Guide** – provides a comprehensive listing of the names and syntax for commands, functions, system procedures, extended system procedures, data types, and utilities in a pocket-sized book (regular size when viewed in PDF format).

- **Reference Manual** – is a series of four books that contains the following detailed Transact-SQL information:
  - **Building Blocks** – Transact-SQL datatypes, functions, global variables, expressions, identifiers and wildcards, and reserved words.
  - **Commands** – Transact-SQL commands.
  - **Procedures** – Transact-SQL system procedures, catalog stored procedures, system extended stored procedures, and `dbcc` stored procedures.
  - **Tables** – Transact-SQL system tables and `dbcc` tables.
  - **System Administration Guide** –
• **Volume 1** – provides an introduction to the basics of system administration, including a description of configuration parameters, resource issues, character sets, sort orders, and diagnosing system problems. The second part of this book is an in-depth description of security administration.

• **Volume 2** – includes instructions and guidelines for managing physical resources, mirroring devices, configuring memory and data caches, managing multiprocessor servers and user databases, mounting and unmounting databases, creating and using segments, using the `reorg` command, and checking database consistency. The second half of this book describes how to back up and restore system and user databases.

• **System Tables Diagram** – illustrates system tables and their entity relationships in a poster format. Full-size available only in print version; a compact version is available in PDF format.

• **Transact-SQL User’s Guide** – documents Transact-SQL, the Sybase enhanced version of the relational database language. This manual serves as a textbook for beginning users of the database management system. This manual also contains descriptions of the `pubs2` and `pubs3` sample databases.

• **Troubleshooting Series (for release 15.0)** –
  • **Troubleshooting: Error Messages Advanced Resolutions** – contains troubleshooting procedures for problems that you may encounter when using Sybase® Adaptive Server® Enterprise. The problems addressed here are those which the Sybase Technical Support staff hear about most often
  • **Troubleshooting and Error Messages Guide** – contains detailed instructions on how to resolve the most frequently occurring Adaptive Server error messages. Most of the messages presented here contain error numbers (from the `master..sysmessages` table), but some error messages do not have error numbers, and occur only in Adaptive Server’s error log.

• **User Guide for Encrypted Columns** – describes how configure and use encrypted columns with Adaptive Server

• **Using Adaptive Server Distributed Transaction Management Features** – explains how to configure, use, and troubleshoot Adaptive Server DTM features in distributed transaction processing environments.
• Using Sybase Failover in a High Availability System – provides instructions for using Sybase Failover to configure an Adaptive Server as a companion server in a high availability system.

• Unified Agent and Agent Management Console – describes the Unified Agent, which provides runtime services to manage, monitor and control distributed Sybase resources.

• Utility Guide – documents the Adaptive Server utility programs, such as *isql* and *bcp*, which are executed at the operating system level.


• XA Interface Integration Guide for CICS, Encina, and TUXEDO – provides instructions for using the Sybase DTM XA interface with X/Open XA transaction managers.

• XML Services in Adaptive Server Enterprise – describes the Sybase native XML processor and the Sybase Java-based XML support, introduces XML in the database, and documents the query and mapping functions that comprise XML Services.

Use the Sybase Getting Started CD, the SyBooks CD, and the Sybase Product Manuals Web site to learn more about your product:

• The Getting Started CD contains release bulletins and installation guides in PDF format, and may also contain other documents or updated information not included on the SyBooks CD. It is included with your software. To read or print documents on the Getting Started CD, you need Adobe Acrobat Reader, which you can download at no charge from the Adobe Web site using a link provided on the CD.

• The SyBooks CD contains product manuals and is included with your software. The Eclipse-based SyBooks browser allows you to access the manuals in an easy-to-use, HTML-based format. Some documentation may be provided in PDF format, which you can access through the PDF directory on the SyBooks CD. To read or print the PDF files, you need Adobe Acrobat Reader.

Refer to the *SyBooks Installation Guide* on the Getting Started CD, or the *README.txt* file on the SyBooks CD for instructions on installing and starting SyBooks.
The Sybase Product Manuals Web site is an online version of the SyBooks CD that you can access using a standard Web browser. In addition to product manuals, you will find links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and the Sybase Developer Network.

To access the Sybase Product Manuals Web site, go to Product Manuals at http://www.sybase.com/support/manuals/.

Technical documentation at the Sybase Web site is updated frequently.

**Finding the latest information on product certifications**

2. Click Certification Report.
3. In the Certification Report filter select a product, platform, and timeframe and then click Go.
4. Click a Certification Report title to display the report.

**Finding the latest information on component certifications**

2. Either select the product family and product under Search by Base Product; or select the platform and product under Search by Platform.
3. Select Search to display the availability and certification report for the selection.

**Creating a personalized view of the Sybase Web site (including support pages)**

Set up a MySybase profile. MySybase is a free service that allows you to create a personalized view of Sybase Web pages.

2. Click MySybase and create a MySybase profile.
Finding the latest information on EBFs and software maintenance


2. Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.

3. Select a product.

4. Specify a time frame and click Go. A list of EBF/Maintenance releases is displayed.

   Padlock icons indicate that you do not have download authorization for certain EBF/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the “Technical Support Contact” role to your MySybase profile.

5. Click the Info icon to display the EBF/Maintenance report, or click the product description to download the software.

Conventions

The following sections describe conventions used in this manual.

SQL is a free-form language. There are no rules about the number of words you can put on a line or where you must break a line. However, for readability, all examples and most syntax statements in this manual are formatted so that each clause of a statement begins on a new line.Clauses that have more than one part extend to additional lines, which are indented. Complex commands are formatted using modified Backus Naur Form (BNF) notation.

Table 1 shows the conventions for syntax statements that appear in this manual:

<table>
<thead>
<tr>
<th>Element</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command names, procedure names, utility names, and other keywords display in sans serif font.</td>
<td><code>select</code></td>
</tr>
<tr>
<td>Database names and datatypes are in sans serif font.</td>
<td><code>sp_configure</code></td>
</tr>
<tr>
<td>Book names, file names, variables, and path names are in italics.</td>
<td><code>System Administration Guide</code></td>
</tr>
<tr>
<td></td>
<td><code>sql.ini file</code></td>
</tr>
<tr>
<td></td>
<td><code>column_name</code></td>
</tr>
<tr>
<td></td>
<td><code>$SYBASE/ASE</code> directory</td>
</tr>
</tbody>
</table>

Table 1: Font and syntax conventions for this manual
**Element** | **Example**
---|---
Variables—or words that stand for values that you fill in—when they are part of a query or statement, are in italics in Courier font. | `select column_name`<br>`from table_name`<br>`where search_conditions`

Type parentheses as part of the command. | `compute row_aggregate (column_name)`

Double colon, equals sign indicates that the syntax is written in BNF notation. Do not type this symbol. Indicates “is defined as”. | `::=`

Curly braces mean that you must choose at least one of the enclosed options. Do not type the braces. | `{cash, check, credit}`

Brackets mean that to choose one or more of the enclosed options is optional. Do not type the brackets. | `[cash | check | credit]`

The comma means you may choose as many of the options shown as you want. Separate your choices with commas as part of the command. | `cash, check, credit`

The pipe or vertical bar (|) means you may select only one of the options shown. | `cash | check | credit`

An ellipsis (…) means that you can repeat the last unit as many times as you like. | `buy thing = price [cash | check | credit]`
`[, thing = price [cash | check | credit]]…`

You must buy at least one thing and give its price. You may choose a method of payment: one of the items enclosed in square brackets. You may also choose to buy additional things: as many of them as you like. For each thing you buy, give its name, its price, and (optionally) a method of payment.

- Syntax statements (displaying the syntax and all options for a command) appear as follows:
  ```
  sp_dropdevice [device_name]
  ```
  For a command with more options:
  ```
  select column_name
  from table_name
  where search_conditions
  ```
  In syntax statements, keywords (commands) are in normal font and identifiers are in lowercase. Italic font shows user-supplied words.

- Examples showing the use of Transact-SQL commands are printed like this:
  ```
  select * from publishers
  ```

- Examples of output from the computer appear as follows:
In this manual, most of the examples are in lowercase. However, you can disregard case when typing Transact-SQL keywords. For example, SELECT, Select, and select are the same.

Adaptive Server’s sensitivity to the case of database objects, such as table names, depends on the sort order installed on Adaptive Server. You can change case sensitivity for single-byte character sets by reconfiguring the Adaptive Server sort order. For more information, see the System Administration Guide.

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

Adaptive Server HTML documentation has been tested for compliance with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

Note You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at http://www.sybase.com/accessibility. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

If you need help Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support. If you cannot resolve a problem using the manuals or online help, please have the designated person contact Sybase Technical Support or the Sybase subsidiary in your area.
CHAPTER 1  Overview

This chapter discusses the following topics.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is Unified Agent?</td>
<td>1</td>
</tr>
<tr>
<td>Services provided</td>
<td>1</td>
</tr>
<tr>
<td>Agent Management Console</td>
<td>2</td>
</tr>
</tbody>
</table>

What is Unified Agent?

Unified Agent provides runtime services to manage, monitor, and control distributed Sybase resources. The Unified Agent provides a common set of services, as well as the ability to host agent plug-ins to manage server resources or to perform various operations. These agent plug-ins can perform product specific command, control, and discovery capabilities including: status, start, stop, restart, ping and server log retrieval.

Unified Agent extends Java Management Extensions (JMX) to provide these management interfaces. Unified Agent enables other Sybase products to “plug in” to an agent to provide remote management capabilities.

Services provided

The most important services provided are port configuration and port conflict resolution.
Agent Management Console

The Agent Management Console is a Sybase Central™ plug-in that manages Unified Agent and its components. The Agent Management Console is a centralized tool for a system administrator to monitor, configure, administer, and manage the agent environment.

For more information on Agent management, see “Agent Management Console Features and Functionality” on page 15.
CHAPTER 2

Installing and Configuring
Unified Agent and Agent
Management Console

This chapter contains information about installing and configuring
Unified Agent and the Agent Management Console.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-installation tasks</td>
<td>3</td>
</tr>
<tr>
<td>Installing Unified Agent</td>
<td>4</td>
</tr>
<tr>
<td>Configuring Unified Agent</td>
<td>7</td>
</tr>
</tbody>
</table>

Pre-installation tasks

Before installing Unified Agent, determine what type of discovery
services to use:

- User Datagram Protocol (UDP) Listener – opens a listener on the
current subnet and “listens” for a broadcast request for information.
  This discovery method contacts those agents that respond to the
  broadcast request. If the resources you want to discover are all on the
  same subnet, choose UDP.

- JINI registry – requires a running JINI server to be accessible in the
  enterprise, that is across subnet boundaries. If a JINI discovery
  service is to be used, you must enter the JINI server’s host name and
  port number during the installation process. If the resources you want
to discover are not all on the same subnet, allocate a resource to run
the JINI discovery server.

These values are provided during installation at the Custom Configure
Unified Agent step of the Adaptive Server™ Installer. After installation,
users can use the Agent Management Console to add, configure, and
remove discovery adaptors.
Installing Unified Agent

Unified Agent is installed as a component of the Adaptive Server™ installation. For information about installing Adaptive Server, see the *Adaptive Server Installation Guide* for your platform.

Installation requires that you log in with administrative privileges.

Upon installation, the $SYBASE/ua directory includes the following folders:

- **bin** – start and stop scripts for the agent and JINI server.
- **conf** – configuration information for agent, security service, and security policy.
- **ldap** – LDAP schema definition.
- **log** – log file for agent, RMI server, and JINI server.
- **plugins** – contains agent plug-ins for each managed resource (for example, the ASEAgentPlugin.)

Starting the JINI server

If JINI has been selected as the discovery method, you must start a JINI server. On the host where the JINI server will run:

1. Change to the Sybase installation directory $SYBASE on UNIX platforms or %SYBASE% on Windows platforms.
2. Set the Sybase environment variables by sourcing SYBASE.csh or executing SYBASE.bat.
3. Change to the $SYBASE_UA/bin or %SYBASE_UA%\bin directory.
4. Execute startJini or startJini.bat.

Starting and stopping Unified Agent

Before you open a support call, or when you identify candidates for new boundaries:

1. Change to the Sybase installation directory—$SYBASE or %SYBASE%.
2. Set the Sybase environment variables by sourcing SYBASE.csh or executing SYBASE.bat.
3 Change to the $SYBASE_UA/bin or %SYBASE_UA%\bin directory.
4 Execute uafstartup.sh or uafstartup.bat, or add the parameter /background to run the Unified Agent in the background.

To stop Unified Agent:
1 Change to the Sybase installation directory—$SYBASE or %SYBASE%.
2 Set the Sybase environment variables by sourcing SYBASE.csh or executing SYBASE.bat.
3 Execute shutdown or uafshutdown.bat.

Starting Unified Agent as a service on UNIX platforms

A UNIX service is a process that automatically runs in the background after the machine is started. On UNIX and Linux platforms, a shell script, agentd, is available in the directory $SYBASE_UA/bin. To install the agent as a service:

1 Copy the agentd into the /etc/init.d directory.
2 Make two soft links in the directory /etc/rcX.d (where X is the run level, for example, 3) that link to /etc/init.d/agentd: SXXagentd and KXXagentd (where XX is a two digit number, for example, 88). One is for starting the service and the other is for stopping the service.

Note Different UNIX systems might have slightly different directory structures.

Starting Unified Agent as a service on Windows platforms

Unified Agent can be started as a Windows service using the Microsoft Windows Administrative Tools | Services dialog. If you cannot find the “Service” dialog, contact your system administrator.

1 Locate the “Sybase Unified Agent” in the list of services.
2 Double-click the service. A dialog containing several tabs displays.
3 From the General tab, choose Start. Unified Agent starts.
4 You can set the service to automatically start when the system starts by changing “Startup type” to “Automatic.”
Additionally, you can automatically restart the service in case of failover by choosing the “Recovery” tab on this same dialog and changing the First, Second, and Subsequent failures to “Restart Service.”

Click Apply to save the modifications before closing the dialog.

Installing the Agent Management Console

The Agent Management Console is a Sybase Central™ plug-in that manages Unified Agent and its components. The Agent Management Console is a centralized tool for a system administrator to monitor, configure, administer, and manage the agent environment.

After successfully installing Adaptive Server and the Unified Agent, register the Agent Management Console Plug-in to Sybase Central and start Sybase Central. To do this, run the following scripts:

For Windows:
- \%SYBASE\%AMCP\bin\registerAMCP.bat
- \%SYBASE\%Shared\Sybase Central 4.3\win32\scjview.exe

For UNIX:
- $SYBASE/AMCP/bin/registerAMCP
- $SYBASE/source SYBASE.csh
- $SYBASE/shared/sybcentral43/scjview.sh

An alternate way to register the Agent Management Plug-in from within Sybase Central is to select Tools | Plug-ins. Click the Register button and specify the plug-in registration file found in:

- On Windows – \%SYBASE\%AMCP\amcplugin.jpr
- On UNIX – $SYBASE/AMCP/amcplugin.jpr

Click Finish to complete the plug-in registration.
Configuring Unified Agent

For information on how to update configurations for your environment’s specific requirements, see Chapter 4, “Agent Management Console Features and Functionality.”

Post-configuration tasks

**Security**
- In a simple login module:
  - Assign a password for the default administrative login, uafadmin. Enter a password that conforms to your password standards. This login allows you to connect to Unified Agent and perform administrative tasks when other security modules are not available.
  - Turn on encryption for the password. By default, encryption is not enabled.
  - Security Services – the security service provides encryption services used by various components such as the Simple Login Module. The default encryption algorithm is set to DES. To change the algorithm to RSA, see Appendix A, “Password Encryption and Security Configuration.”

**Discovery**
- Login modules – define role mappings for each login module defined in addition to the Anonymous Login Module and the Simple Login Module. Create mappings between native roles and the roles defined in Unified Agent. At a minimum, map to the “uaAgentAdmin” role to allow administrators to administer the agent without using the Simple Login Module.
  - Discovery Adapters – add discovery adapters to enable the agent to register with additional discovery services. You can also edit or delete existing adapters.

**Agent Properties**
- SNMP Service – the SNMP Service is a Simple Network Management Protocol Agent that responds to client requests and sends SNMP notification messages, called traps. The default port for the SNMP command responder is UDP 1498. The default notification port for SNMP traps is 49152. This service is not started unless it is required by another service or an agent plug-in, and its enablement depends on the plug-in’s ability to use it.
Configuring Unified Agent

- TDS™ Service – The TDS Service provides Tabular Data Stream remote procedure calls into the Unified Agent Framework. The default TDS port number is 9995. This service is not started unless it is required by another service or an agent plug-in, and its enablement depends on the plug-in’s ability to use it.

After you have made all the changes, restart the agent.
CHAPTER 3

Security

This chapter discusses general security concepts for Unified Agent. All administrative tasks are performed through the Agent Management Console. See Chapter 4, “Agent Management Console Features and Functionality” for details on how to administer security.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unified Agent security</td>
<td>9</td>
</tr>
</tbody>
</table>

Unified Agent security

Unified Agent security does not maintain its own security repository, except in supporting a UAF enabled from outside the machine running UAF. Instead, it delegates the authentication to existing systems such as Adaptive Server or the operating system. Security modules define these systems to which security has been delegated.

When a user is authenticated, roles for the user are retrieved by the security module and are mapped to Unified Agent defined roles. Permission to Unified Agent resources are granted to Unified Agent roles. Authorization is resolved through the mappings between the security module native roles and Unified Agent roles.

The Security Service authenticates logins and authorizes access to agent resources. It also provides password encryption services.

Encryption

The agent-config.xml and csi.properties files can contain passwords. By default, these passwords are not encrypted. To enable the encryption of these passwords, use the Agent Management Console to set the “encrypted” property to true for the Simple Login Module and for any LDAP discovery adaptors defined for the agent.
Unified Agent security

To allow installations to choose their own encryption algorithm and cipher provider, Unified Agent does not enable encryption by default. Therefore, it is important that you enable encryption and to set the encryption and cipher provider class.

**Note** Encryption is not enabled by default.

Use the attributes of the Security Service to specify the encryption algorithm and the cipher provider class. The attributes are `encryptionAlgorithm` and `cipherProviderClass`. The default encryption algorithm is DES and the cipher provider class is Sun’s JCE provider.

Authentication

Authentication is the process by which the system confirms the identity of each user. Unified Agent provides a set of five predefined login modules. All login modules are defined in the `$SYBASE_UA/conf/csi.properties` file or the `%SYBASE_UA%\conf\csi.properties` file. The syntax is defined by the Sybase Common Security Infrastructure (CSI) framework. By assembling and configuring different login modules, you can customize Unified Agent authentication to your strength requirements.

Login modules

- Anonymous Login Module – accepts a client connection whose user name is “anonymous,” and assumes the role of uaAnonymous. The anonymous login does not require a password. The Anonymous Login Module is required and cannot be removed.

- Simple Login Module – defines a user name, password, and a list of roles. The default user name is “uafadmin” with a blank password and a native role of uaAgentAdmin. By default, encryption is not enabled.

- Adaptive Server Login Module – delegates the authentication of users to the Adaptive Server. This module requires ASE Agent Plug-in. The user name must conform to the format of `ase_login@ase_servername`, for example, `sa@testmachine`. To authenticate, Adaptive Server must be running on the Agent’s machine and have an ASE Agent plug-in configured.
• UNIX Proxy Login Module – delegates authentication to the underlying UNIX or Linux operating system using Pluggable Authentication Modules (PAM).

**Note** On some platforms, the SYBASE_UA rlib proxy file needs to have execute permission or the UNIX Proxy Login Module will fail.

• NT Proxy Login Module – delegates authentication to the underlying Windows operating system. The user name must conform to the format of `username@nt-domain-name`. For example, `user@sybase`.

When you log in, each login module attempts authentication in the order specified in the `csi.properties` file until a successful module is found.

**Note** The Anonymous Login Module must always appear first.

### Security module control flags

Each security module has a control flag that determines how the overall authentication process behaves. The control flag value indicates whether the success of the security module is:

• **Required** – the LoginModule is required to succeed. If it succeeds or fails, authentication still continues to proceed down the LoginModule list.

• **Requisite** – the LoginModule is required to succeed. If it succeeds, authentication continues down the LoginModule list. If it fails, control immediately returns to the application (authentication does not proceed down the LoginModule list).

• **Sufficient** – the Login Module is not required to succeed. If it does succeed, control immediately returns to the application (authentication does not proceed down the LoginModule list). If it fails, authentication continues down the LoginModule list.

• **Optional** – the LoginModule is not required to succeed. If it succeeds or fails, authentication still continues to proceed down the LoginModule list.
Authorization

Authorization determines if a user has permission to access a service or plug-in resource. A resource is defined as an attribute or operation of an agent service or agent plug-in.

Unified Agent security is role-based. Permissions to agent service and agent plug-in resources are granted to unified agent roles. These unified agent roles are mapped back to roles returned by each security module.

The mapping of unified agent roles to security module roles allows the agent to tie its authorization model back to the authenticating security module.

When a client application attempts to access a resource, the authorizer compares the user's rights with the resource's access requirements.

Unified agent resources have default permissions set to the predefined unified agent roles. In most cases, only the role mappings need to be defined.

The predefined unified agent roles are:

- uaUser – for a typical end user
- uaGuest – for a temporary user
- uaAnonymous – for an unauthenticated user
- uaAgentAdmin – for the administrator of the Unified Agent
- uaPluginAdmin – for the administrator of agent plug-ins
- uaASEAdmin – for the administrator of the Adaptive Server
- uaOSAdmin – for the administrator of the operating system

Role mappings

Since Unified Agent delegates authentication through the security modules, role mappings between native security module roles and Unified Agent roles determine authorization. You must define mappings for each security module to allow access to agent resources.

Note On some platforms, administrators may need to know the groups that you are a member of in order to map to these groups.
Permissions

A set of permissions determines access to agent resources. Permissions are granted to Unified Agent roles for each resource. Attribute resources have read and write permission and operation resources have execute permission.

Agent resources have default permissions assigned to the Unified Agent roles. Most installations would not need to change the permissions for individual resources.
Unified Agent security
CHAPTER 4

Agent Management Console
Features and Functionality

This chapter contains information about Agent Management Console features and functionality.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managing look-up services</td>
<td>15</td>
</tr>
<tr>
<td>Managing Unified Agents</td>
<td>17</td>
</tr>
<tr>
<td>Managing discovery adaptors</td>
<td>20</td>
</tr>
<tr>
<td>Managing security modules</td>
<td>22</td>
</tr>
<tr>
<td>Managing role mappings</td>
<td>24</td>
</tr>
<tr>
<td>Managing agent services</td>
<td>25</td>
</tr>
<tr>
<td>Managing agent plug-ins</td>
<td>26</td>
</tr>
<tr>
<td>Managing attributes</td>
<td>29</td>
</tr>
<tr>
<td>Managing operations</td>
<td>31</td>
</tr>
</tbody>
</table>

The Agent Management Console is a Sybase Central plug-in that manages Unified Agent and its components. The Agent Management Console is a centralized tool for a system administrator to monitor, configure, administer, and manage the agent environment.

Managing look-up services

Unified Agent servers can broadcast themselves on a subnet using UDP, or register themselves with a look-up server such as JINI or LDAP. Within the Agent Management Console plug-in, the ‘Lookup Services’ folder allows the administrator to create look-up profiles to query such services. Each profile specifies the network protocol, host, port, and an optional plug-in name search filter.
Creating a network lookup profile

❖ Creating a network look up profile

1. Specify a unique user defined name for the lookup profile to be created.
2. From the drop-down menu, select the protocol. Your choices are:
   • JINI
   • LDAP
   • UDP
3. Specify the host name of the JINI or LDAP server. The UDP protocol does not require a host name and is therefore disabled when UDP is selected. If you do not know the name and port of the lookup server, contact your Unified Agent or network administrator.
4. The default port for JINI is 4160. However, if you do not know the port for the lookup server, contact your Unified Agent or network administrator.
5. Specify an optional plug-in name search filter. The discovery process uses this filter to perform a string pattern match on the Agent plug-ins registered on each agent. Only agents with positive matches are returned in the lookup query.

   For example, if you create a plug-in search filter called ASE, your search will return all agents with plug-ins having the letters ASE. This search is case sensitive.

Discovering Unified Agents

To discover Unified Agents, open the defined network lookup profile. The list of discovered agents are displayed on the right side details pane.

The list of discovered agents can be refreshed through the menu File | Refresh option.

You can establish a connection to a listed agent through the Connect or Connect As menu options. For more information on agents, see “Managing Unified Agents” on page 17.

When a connection is made, the agent is added to the tree view on the left.
CHAPTER 4  Agent Management Console Features and Functionality

Deleting a network look-up profile

❖ Deleting a network lookup profile
1  Right click the network lookup profile.
2  Select Delete.
3  Select Yes on the confirmation dialog.

Changing a network look-up profile

The properties dialog for the network lookup profile has information about the protocol, the host name, the port number, and the plug-in search filter. The values here are the values you specified in the Add a new network profile wizard. You can edit these values in this dialog.

When you have finished editing the dialog, select OK to save any new values, and begin your plug-in search.

Specifying a network plug-in name filter

By specifying an optional plug-in name search filter, the discovery process will use this filter to perform a string pattern match on the agent plug-ins registered on each agent. Only agents with positive matches are returned in the lookup query.

Managing Unified Agents

A Unified Agent is a process on a server that is running a service or plug-in

Manually adding an agent

❖ Manually adding an agent the remembered agents in the tree view
1  Right click the Agent Management Console node in Sybase Central.
2  Select Add agent.
3 Enter a host name and port number of the agent server.
4 Click OK.

**Connecting to an agent**

- **Connecting to an agent**
  1 Right click on a remembered agent on the tree view or on an agent listed under a network lookup profile.
  2 Select Connect to connect to the agent anonymously, or:
     Select Connect As to create an authenticated connection. If you select Connect As you must provide a user ID and password.
  3 Click OK.

**Adding credentials**

When you connect to an agent, you are authenticated through a login module. You acquire the native roles of that login to create a set of credentials for your session. You can add credentials to your session by providing another login ID and password. The additional login can be through the same login module or a different login module. After the additional login is authenticated, native roles for the new login are added to your session. This allows you to build up a set of credentials for session to perform the desired tasks.

- **Adding credentials**
  1 Right click on a connected agent in the Agent Management Console in Sybase Central and select Add Credential, or select File | Add Credential.
  2 Enter a new user ID and password to add additional login information.
  3 Click OK.
  4 If successful, your session assumes additional privileges of the new login credentials.
CHAPTER 4    Agent Management Console Features and Functionality

Disconnecting from an agent

❖ Disconnecting from an agent
1 Right-click on a connected agent.
2 Select Disconnect.

Restarting an agent

Certain changes made to agents require that you restart the agent.

❖ Restarting an agent
1 Right-click on the agent.
2 Select Restart.

Retrieving the agent log

You can view the Unified Agent log through the Agent Management Console in Sybase Central. The log contains status information, activity, warnings, and errors.

❖ Retrieving the agent log
1 Right-click on the agent.
2 Select retrieve log.
3 The agent log is displayed in the bottom window in Sybase Central.

Viewing and changing agent properties

❖ Viewing agent properties
1 Right-click on the agent.
2 Select Properties.
3 The agent properties dialog contains three tabs that displays detail, encryption and session information.
Managing discovery adaptors

Discovery adaptors allow the agent to configure how it can be discovered.

Creating a discovery adaptor

You may register the agent to multiple discovery servers by creating multiple discovery adapters. Since UDP, unlike JINI and LDAP, is a connectionless protocol and broadcasts within a subnet, only one UDP adaptor is supported.

❖ Creating a discovery adaptor

1. Open the Discovery Adaptor wizard from the Discovery Adaptor folder for an agent.
2. Enter a unique ID for the discovery adaptor to be added.
3. Select the discovery adaptor class.
4 Click Next.
5 A screen displays notifying you that the new discovery adaptor will be created with default values for the selected adaptor class. Use the properties dialog to change any default values before restarting the agent. Click finish to complete the wizard.
6 See “Viewing and changing discovery adaptor properties” on page 21 to change any default values for the new discovery adaptor.

Deleting a discovery adaptor

❖ Deleting a discovery adaptor
1 To delete a discovery adaptor, right click delete.
2 Click Yes to confirm that you want to delete the selected discovery adaptor.

Viewing and changing discovery adaptor properties

❖ Viewing discovery adaptor properties
1 Right click on a discovery adaptor.
2 Select properties.
3 Click Yes to apply the changes.
4 The changes take effect when you restart the agent.

The properties page has the following information. This information is different depending on the discovery adaptor class:

• Heartbeat period – notifies the JINI server that the agent is still running. The default is every 900 seconds.
• Class
• Host name
• Port number

Once you have selected the values for the discovery adaptor, restart the agent. When you restart the agent, your connection is lost, so you must reconnect to the agent.
Managing security modules

There are three different types of discovery protocols:

- JINI
- LDAP
- UDP

Setting discovery adaptor password

If a discovery adaptor, such as LDAP, requires a password, the password can be set through a menu item.

- Setting a discovery adaptor password
  1. Right click on the adaptor.
  2. Select Set Password.
  3. Type and confirm the password.
  4. Click OK.

Managing security modules

The Unified Agent does not maintain its own security repository. Instead, it delegates the authentication to existing systems such as Adaptive Server or the operating system. Security modules define these systems to which security has been delegated. Unified Agent security is based on the JAAS framework.

Creating a security module

- Creating a security module
  1. Specify a unique name for the security module.
  2. Select the security provider class for the new module.
  3. When multiple security modules are configured, the control flag defines the authentication sequence. The options are optional, required, requisite and sufficient.
  4. Click Next.
5 Specify the order for the security modules by selecting a module and using the Move up and Move down buttons. The order is significant because authentication is performed sequentially adhering to the rules set by the control flags.

6 Click Next.

7 A security module wizard prompt reminds you to create valid role mappings for the security module using the role mapping wizard. You can view and change security module default property values in the properties dialog.

8 Click Finish.

Deleting a security module

❖ Deleting a security module
1 Right click on the security module.
2 Select Delete.
3 Click Yes in the confirmation box.

Ordering the security module authentication

The order of the security modules is significant. Authentication is performed sequentially adhering to the control flags defined for each security module.

❖ Modifying the security module authentication order
1 Select a security module by clicking on it.
2 Use the move up or move down button to change the position of the selected module.
3 Click OK when you have finished. Changes are not in effect until the agent is restarted.

Viewing and changing security module properties

The Security Module Properties dialog allows you to:
Managing role mappings

- View and change properties of the security modules.

**Note** The security module class cannot be modified.

- The list of properties shown is dependent on the defined security module class.

Setting a security module password

If a security module requires a password, it can be set through a dialog accessed from the menu.

- **Setting a security module password**
  1. Right click on the security module.
  2. Select Set Password.
  3. Type and confirm the password.
  4. Click OK.

Managing role mappings

Unified agent security is role-based. Resource permissions are granted to unified agent roles. The mapping of native roles or groups, from the security modules to the unified agent roles, determines authorization. A user ID authenticated through a security module possesses roles or groups. These native roles, or groups, are mapped to unified agent roles. Each security module contains a set of mappings.

Creating a role mapping

- **Mapping a new security role**
  1. Specify the name of the role or group for which you wish to create a mapping.
2 Then choose a Unified Agent role to which you want to map. Unified Agent comes with default permissions for these UA roles. Your choices are:

- uaUser – an authenticated user with general levels of access.
- uaGuest – a temporary user with limited levels of access.
- uaAnonymous – an unauthenticated user with low level access.
- uaAgentAdmin – an administrator for the agent with the highest level of access.
- uaPluginAdmin – a plug-in administrator with administrative access restricted to the plug-ins.
- uaASEAdmin – similar to uaPluginAdmin but restricted only to the ASE Agent plug-in. Usually mapped to Adaptive Server administrators.
- uaOSAdmin – an administrator with a high level of access. Usually mapped to an OS administrator.

3 Click finished to create the new role mapping.

Managing agent services

The AgentService folder includes a list of services, mandatory as well as optional, running on the agent. Each service provides a set of functionalities, used by the agent, other services, or agent plug-ins. The mandatory services listed are Agent Service, Configuration Service, Environment Service, and Session Service. Mandatory services cannot be started or stopped individually.

Starting an agent service

If optional agent services are not running, you can start them.

Starting an agent service

1 Right click on the agent service.
2 Select Start.
3 Select Yes in the confirmation box.
Managing agent plug-ins

Restarting an agent service

If you want to implement changes you have made in the agent service, you can select restart.

❖ Restarting an agent service.
1 Right click on the agent service.
2 Select Restart.
3 Select Yes in the confirmation box.

Stopping an agent service

Some of the optional agent services can be stopped.

❖ Stopping an agent service
1 Right-click on the agent service.
2 Select Stop.
3 Select Yes in the confirmation box.

Viewing agent service properties

❖ Viewing agent service properties
1 Right click on the agent service.
2 Select Properties.
3 The properties dialog displays with information about the name and class.

Managing agent plug-ins

An agent plug-in is a process that runs on the agent that performs tasks. Tasks can include monitoring, managing, and controlling a server such as the Adaptive Server.
CHAPTER 4    Agent Management Console Features and Functionality

Loading an agent plug-in
If an agent plug-in is not loaded, you can load them.

❖ Loading an agent plug-in
1 Right click on the agent plug-in.
2 Select Load.
3 Select Yes in the confirmation box.

Unloading an agent plug-in
Agent plug-ins can be unloaded.

❖ Unloading an agent plug-in
1 Right click on the agent plug-in.
2 Select Unload.
3 Select Yes in the confirmation box.

Reloading an agent plug-in
Agent plug-ins can be reloaded.

❖ Reloading an agent plug-in
1 Right click on the agent plug-in.
2 Select Reload.
3 Select Yes in the confirmation box.

Viewing agent plug-in properties
The agent plug-in properties page displays read-only information about the name and class.

❖ Viewing agent plug-in properties
1 Right click on the agent plug-in.
2 Select Properties.
Managing agent plug-ins

3 The properties dialog displays with information about the name and class.

Server-based plug-ins

Server based plug-ins are plug-ins that manage servers. For server-based plug-ins there is server specific functionality that they can perform on the server.

Pinging the server

❖ Pinging a server
1 Right click on the server-based plug-in.
2 Select Ping Server.

Starting the server

❖ Starting the server
1 Right click on the server-based plug-in.
2 Select Start.
3 You are notified that the start command has been invoked on the server.
4 The server’s log is retrieved and displayed on the bottom panel to show the status of the server.

Restarting the server

❖ Restarting the server
1 Right click on the server-based plug-in.
2 Select Restart.
3 You are notified that the restart command has been invoked on the server.
4 The server’s log is retrieved and displayed on the bottom panel to show the status of the server.
## Stopping the server

- **Stopping the Server**
  1. Right Click on the server-based plug-in.
  2. Select Stop Server.
  3. You are notified that the stop command has been invoked on the server.
  4. The server’s log is retrieved and displayed on the bottom panel to show the status of the server.

## Retrieving the server log

- **Retrieving the server log**
  1. Right click on the server-based plug-in.
  2. Select Retrieve server log.
  3. The server’s log is retrieved and displayed on the bottom panel.

## Managing attributes

Agent Services and Agent Plug-ins contain attributes that can be managed individually. An attribute has a name, type, access level, description and value. Depending on the attribute’s access level, you can assign permissions for UAdefined roles.

## Viewing attributes

You can view an attribute through the details pane of Sybase Central. The details pane shows:

- **Name** – is the name of the attribute
- **Type** – is the Java class or datatype of the attribute
- **Access** – shows whether the attribute is readable and or writable:
  - RO – read only
  - RW – read/write
Managing attributes

- WO – write only
- Value – the display of the attribute’s value is dependent on its access as well as your login’s rights.
- Description – displays the description of the attribute

Updating attribute values

- **Updating attribute values**
  1. Right click on the attribute.
  2. If the attribute is writable and you have sufficient permissions, you can select Update.
  3. For an attribute where you have insufficient access, you are prompted for additional credentials.
  4. You are prompted for a new attribute value.
  5. Enter the new value and click OK.

Viewing properties and changing permissions

- **Viewing attribute properties**
  1. Right click on the attribute.
  2. Select properties.
  3. The properties dialog displays.
  4. The details tab has read-only information about:
     - Name
     - Type
     - Access
     - Description
  5. The permissions tab has a list of roles to which you can grant and revoke access.
  6. Click OK.
Managing operations

An operation is a method or function of an agent service or agent plug-in.

Viewing operations

- **Viewing operations**
  1. Select a service or a plug-in.
  2. On the details pane, select the operations tab.
  3. The operations tab has information about
     - Name – is the name of the operation.
     - Signature – the signature is the parameters for the operation.
     - Return Type – is the type of data or object that is returned by the operation.
     - Description – of the operation.
  4. Use the permissions tab to grant permission.
Password Encryption and Security Configuration

Password encryption utility

A password encryption utility named `passencrypt` is available in the agent `bin` directory. It can be used to generate encrypted passwords using a supported encryption algorithm.

Syntax of `passencrypt`:

```
passencrypt [-help] [-algorithm <encryption_algorithm>] [-provider <cipher_provider_class>] -text <clear_text>
```

where:

- help – print help
- algorithm – is the encryption algorithm, and is optional. The default value is DES. The supported algorithms are DES and RSA.
- provider – is the cipher provider class, optional. The default DES provider is Sun’s JCE. Default RSA provider is org.bouncycastle.jce.provider.BouncyCastleProvider.
- text – clear text, required.
Enabling Rivest-Shamir-Adleman (RSA) encryption

Due to United States export restrictions, Unified Agent does not ship with an RSA encryption provider. Unified Agent has been tested with the BouncyCastle encryption provider. To enable RSA base encryption complete the following steps:

2. Register Bouncy Castle with JRE by:
   a. Copying the bcprov-jdk14-124.jar to $SYBASE/shared-1_0/JRE-1_4/lib/ext or %SYBASE%\shared-1_0\JRE-1_4\lib\ext.
   b. Adding the following line to $SYBASE/shared-1_0/JRE-1_4/lib/security/java.security or the %SYBASE%\shared-1_0\JRE-1_4\lib\security\java.security after the line "security.provider.<number>=sun.security.jgss.SunProvider":

      security.provider.<nextnumber>=org.bouncycastle.jce.provider.BouncyCastleProvider

   Note Security providers are listed sequentially. Modify next number to be the next sequential number.

3. From within the Agent Management Console, change the following attributes values in the Security Service:

   encryptionAlgorithm = RSA
   cipherProviderClass = org.bouncycastle.jce.provider.BouncyCastleProvider

4. Restart the agent.
Adaptive Server agent plug-in server password encryption

The Adaptive Server agent plug-in requires a valid and privileged Adaptive Server login and password to perform its management functions. These functions include retrieving the server log and checking user roles for other parts of the Authentication systems. The password is encrypted using the Data Encryption Standard (DES). To change the password:

1. Create an encrypted form of the password using the passencrypt program. First, change directory to $SYBASE_UA/bin or %SYBASE_UA%in. Then, execute the passencrypt utility. For example, enter:

   `passencrypt -text <clear-text-password>`

Remember the encrypted string that was produced.

2. Start the Agent Management Console (AMC) which runs as a Plug-in under Sybase Central.

3. Connect to the Agent. If the agent is not listed as a child of AMC add the Agent by using the drop down menu on AMC and choosing “Add Agent.” A dialog will appear requesting the host and port name. The agent host name is the network name of the host that the agent is running. The port is typically 9999 but that can be changed on a per agent basis. Use the drop down menu over the Agent and choose “Connect as”. Enter a user name and password that provides an administrative role over the ASEAgentPlugin.

4. Locate the ASEAgentPlugin in the plug-ins tab. On the detail pane locate the “Attributes” tab and find the “password” attribute. Update the password attribute with the saved results of the passencrypt program from above by using the drop down menu over the password attribute and choosing “update”.

The ASE Agent Plug-in password has now been changed.

Pluggable Authentication Module (PAM) configuration

After enabling the UNIX login modules on UNIX platforms, complete the following steps using an ID with root privileges:
### Pluggable Authentication Module (PAM) configuration

<table>
<thead>
<tr>
<th>Platform</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solaris platforms</td>
<td>1. Append the contents of the <code>$SYBASE_UA/utility/sunos/pam.conf</code> file, provided with the Unified Agent, to the <code>/etc/pam.conf</code> file on your Solaris platform.</td>
<td>2. Restart the PAM services.</td>
</tr>
<tr>
<td>Linux platforms:</td>
<td>1. Copy the <code>$SYBASE_UA/utility/linux/sybase-ua</code> file, provided with the Unified Agent, to the <code>/etc/pam.d</code> directory on your Linux platform.</td>
<td>2. Restart the PAM services.</td>
</tr>
<tr>
<td>HP-UX platforms:</td>
<td>1. Append the contents of the <code>$SYBASE_UA/utility/hp-ux/pam.conf</code> file, provided with the Unified Agent, to the <code>/etc/pam.conf</code> file on your HP-UX platform.</td>
<td>2. Restart the PAM services.</td>
</tr>
<tr>
<td>IBM AIX platforms:</td>
<td>1. Create or edit the contents of the <code>/etc/pam.conf</code> file and include the content of the <code>$SYBASE_UA/utility/aix/pam.conf</code> file, provided with the UnifiedAgent.</td>
<td>2. Restart the PAM services.</td>
</tr>
</tbody>
</table>
APPENDIX B

SNMP Agent Plug-in

Overview


SNMP sends messages, called protocol data units (PDUs), to different parts of a network. SNMP-compliant devices, called agents, store data about themselves in Management Information Bases (MIBs) and return this data to the SNMP requesters.


Terminology

These acronyms are used throughout this document for the SNMP Agent Plug-in.

JMX

Java Management Extension. Supplies tools for managing and monitoring applications, system objects, devices, and service-oriented networks. This API allows Java classes to be dynamically constructed and changed. UAF is implemented using JMX.
Sybase Adaptive Server MIB

MIB

Management Information Base. A formal description of a set of objects that can be managed using SNMP. MIB-I refers to the initial MIB definition, and MIB-II refers to the current definition. There are MIB extensions for each set of related management entities, such as the RDBMS-MIB, which defines SNMP tables that provide information about managed database servers and their databases, and APPLICATION-MIBs, which provide general purpose attributes for each database server. Standard MIBs can be extended to include proprietary objects.

TDS

Tabular Data Stream™. Open Clients and Open Servers exchange information using this application protocol. All applications built using the Sybase Open Client libraries are also TDS applications, because the Open Client libraries handle the TDS interface. However, some applications, such as jConnect, are TDS applications even though they do not use the Open Client libraries—they communicate directly using TDS protocol.

UAF

Unified Agent Framework™. A JMX-based, service-oriented agent framework. It reduces the cost of developing a distributed agent by providing common services and interfaces for SNMP Agent Plug-in development and deployment. UAF includes the agent server, an agent framework library (both server and client APIs) and sample SNMP Agent Plug-ins.

USM

User-based Security Model. The Simple Network Management Protocol (SNMPv3) that defines the process for providing SNMP message-level security. USM includes an MIB to remotely monitor and manage the configuration parameters for this Security Model. The USM provides authentication for an SNMPv3 protocol.

VACM

View-based Access Control Model. The SNMPv3 that defines the process for controlling access to management information. It includes an MIB to remotely manage the configuration parameters for the VACM. While the USM provides authentication, the VACM provides authorization for an SNMPv3 protocol.

Sybase Adaptive Server MIB

The Sybase Adaptive Server MIB describes information available to SNMP clients. The Adaptive Server Sybase MIB consists of an SNMP table and an SNMP error trap message. The table records the error trap messages. See “Sybase MIB” on page 48 for a sample MIB.
Unified Agent SNMP services and plug-ins

The Unified Agent Framework (UAF) is a server-based management infrastructure for Adaptive Server and other Sybase products. The server is a collection of application management extensions, called services and plug-ins.

UAF SNMP Service

The UAF SNMP Service consists of the following components:

- Dispatcher – dispatch tasks to the multiple version-specific message processing models, and to dispatch PDUs to various agent services and plug-ins.
Unified Agent SNMP services and plug-ins

- Message processing subsystems – interacts with the dispatcher to handle the version-specific SNMP messages. It contains one or more message processing models.
- Security subsystem – provides a framework to support the SNMPv3 protocol, including authentication and privacy on each message.
- Access control subsystem – a set of services that an application (such as a command responder or a notification originator application) can use for checking access rights.

UAF SNMP Service and UAF Adaptive Server SNMP Agent Plug-in managed objects (MOs)

- Command responder – interacts with the Adaptive Server SNMP Agent Plug-in to communicate the request of the SNMP client.
- Notification originator – initiates asynchronous messages on behalf of the Adaptive Server SNMP Agent Plug-in.

UAF SNMP Service MIBs

The UAF SNMP service also provides the following MIBs, which any SNMP client can query:

- SNMP-TARGET-MIB
- SNMP-NOTIFICATION-MIB
- SNMP-FRAMEWORK-MIB
- SNMPv2-MIB
- SNMP-COMMUNITY-MIB
- SNMP-USER-BASED-SM-MIB
- SNMP-VIEW-BASED-ACM-MIB
- SNMP-MPD-MIB
- NETWORK-SERVICES-MIB

These are common MIBs that may be provided with a standard SNMP package, or that you can find online at SNMP repositories.
Adaptive Server-to-SNMP client architecture

This architecture begins with an Adaptive Server stored procedure, and ends with a network management client. The initiation of an Adaptive Server message and the reception of that message as an SNMP trap require three services and plug-ins within UAF. They are:

- Adaptive Server SNMP Agent Plug-in – this plug-in has a `sybRaiseError` method that is exposed through the TDS service. An Adaptive Server stored procedure initiates a remote server connection to the UAF-TDS service, and calls `sybRaiseError`.

- UAF TDS Service – a methodology by which Adaptive Server connects and communicates with the UAF. The UAF TDS Service provides a remote procedure call (RPC), which constructs distributed client-server based applications and provides a calling interface that you can use to call any UAF service or plug-in method. Adaptive Server allows RPC calls using CIS (Component Integration Services).

- UAF SNMP Service – communicates the trap message to any SNMP network management client that was listening for traps.

Together, these three agent services and plug-ins provide the management infrastructure for Adaptive Server SNMP monitoring.

Security

Consider two types of security for SNMP monitoring:

- SNMP connections to Unified Agent
- Adaptive Server connections to Unified Agent

Security for connections from Adaptive Server

The UAF manages security for Adaptive Server connections by configuring the Adaptive Server SNMP Agent Plug-in for the data server.
The UAF must authenticate the Adaptive Server connection and ensure that the authenticated user is authorized to perform subsequent Adaptive Server SNMP operations. By configuring the UAF to use the Adaptive Server user name and password as one of its authentication and authorization modules, Adaptive Server can pass its own user name and password to the UAF.

To set the user name and password for connections to remote servers in Adaptive Server, use `sp_addexterlogin`. For more information see “Examples for configuring the Adaptive Server client” on page 70.

## Security for connections from SNMP

Security for SNMP depends on the protocol you use for the SNMP connections. Community names are used to authenticate the connection to the SNMP protocols v1 and 2c. “Public” and “private” are community names; they are passed to the SNMP agent as words that allow access to SNMP data.

SNMPv3 introduces significant increases in security for SNMP connections. Client connections are authenticated with a user name and password.

This version also introduces encrypted passwords and encrypted message content.

For the default login and user authentication for Unified Agent SNMP agent services, see “UAF SNMP Agent Service” on page 44.

### Sample connection

For example, you can use the program NET-SNMP to connect the SNMP service and to query the `aseErrorTable` from the Sybase Adaptive Server MIB. For more information, see the reference at [http://net-snmp.sourceforge.net/](http://net-snmp.sourceforge.net/).

In “Viewing the SNMP information from NET-SNMP” on page 48, “snmpwalk” is a NET-SNMP application using SNMP GETNEXT requests to query a network entity for a tree of information. In the SNMP protocol version 1 (“-v 1”), “-c public” specifies the community name. “localhost:1498” declares the network-addressable host name at port number 1498. The last keyword identifies the branch of the SNMP tree to query.

```
$ snmpwalk -v 1 -c public MYHOST:1498 enterprise
SNMPv2-SMI::enterprises.897.1.1.1.1.1.1 = INTEGER: 1
SNMPv2-SMI::enterprises.897.1.1.1.1.2.1 = STRING: "WEBSPINNER"
SNMPv2-SMI::enterprises.897.1.1.1.1.3.1 = INTEGER: 1
SNMPv2-SMI::enterprises.897.1.1.1.1.4.1 = INTEGER: 2
SNMPv2-SMI::enterprises.897.1.1.1.1.5.1 = INTEGER: 3
```

```
Configuring Adaptive Server for the SNMP Agent Plug-in

Adaptive Server generates the SNMP message and the SNMP network client receives it. Reception might be either passive, like receiving an SNMP trap, or active, like querying the UAF SNMP service for Adaptive Server messages. Successful configuration ensures that the components communicating Adaptive Server messages are running, and that the configuration values between components match.

For example, the SNMP client cannot receive a trap on port 49152 if the UAF SNMP service is not configured to send the trap on that port. When you configure Adaptive Server, consider these points:

- Which SNMP network client am I connecting to Adaptive Server?
- Is the default port for the UAF SNMP appropriate for my SNMP network client to use for contacting the UAF SNMP service?
- Is the default port number for the UAF SNMP traps an appropriate port number for my SNMP network client to use for listening to traps?
- Is the default port number for the UAF TDS service listener an appropriate port number for my SNMP network client to use for contacting UAF through TDS?
- The recommended UAF TDS server name in the Directory Services file, *interface* or *sql.ini*, is “<hostname>_UAF.” Is this the correct naming convention to follow? (This naming convention is the one used by Adaptive Server when naming its servers.)
- How much SNMP security do I need? Specifically, should I change the community name, the user name, or the pass phrase? Should I add a privacy pass phrase? Do I want the message contents to be encrypted?

Verify that all the following steps are complete when you configure Adaptive Server to generate these messages.
Configuring Adaptive Server to generate SNMP notifications

3. Start Unified Agent.
4. Install and configure Unified Agent SNMP and TDS services.
5. Install the Adaptive Server SNMP Agent Plug-in.
6. Make sure the UAF TDS service is identified in the Adaptive Server interfaces file.
7. Verify that an authenticated and authorized user name and password has been assigned to the Open Client connection.

TDS services

You can use the Agent Management Console to specify the TDS service default number as 9995, or to modify any of the numbers you use when you add a line in the interfaces file. For more information about TDS, see “Tabular Data Stream (TDS) Service” on page 55.

UAF SNMP Agent Service

The SNMP Agent Service is installed with the defaults listed in Table 4-1. Use many of these defaults for your initial connections to the SNMP agent from a client. You can modify these defaults from the Agent Management Console at any time.
Table 4-1: Unified Agent SNMP agent service default values

<table>
<thead>
<tr>
<th>Property</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent engine ID</td>
<td>SYBASE-UAF or 5359424153452d554146</td>
</tr>
<tr>
<td>Transport mapping</td>
<td>UPD: (0.0.0.0/1498)</td>
</tr>
<tr>
<td>Default notification port</td>
<td>49152</td>
</tr>
<tr>
<td>Management name</td>
<td>Sybase, Inc.</td>
</tr>
<tr>
<td>Management location</td>
<td>One Sybase Drive Dublin, CA 94568</td>
</tr>
<tr>
<td>Community name (v1 and v2c)</td>
<td>public</td>
</tr>
<tr>
<td>USM user name (v3)</td>
<td>snmpadmin</td>
</tr>
<tr>
<td>USM authentication pass phrase (v3)</td>
<td>Sybase4me</td>
</tr>
<tr>
<td>USM privacy pass phrase (v3)</td>
<td>null</td>
</tr>
</tbody>
</table>

Note  To maintain security, change the community name, user name, authentication pass phrase, and privacy pass phrase immediately.

Adaptive Server runtime

Adaptive Server supports connection to remote servers, including the UAF TDS Service. Typically, you place server names in the interfaces file to establish the remote host name and server port number. Only one TDS listener and one agent can be running on the host. In this example, the host name “MYHOST” is part of the server name, and the designation “_UAF” is appended to the end. The name for the TDS service is “MYHOST_UAF”.

This example shows an entry in the interfaces file representing this server. The syntax of master and query TCP varies depending on the platform and operating system, but resembles the following:
Adaptive Server runtime

```
myhost_uaf
  master tcp ether myhost 9998
  query tcp ether myhost 9998
```

Adaptive Server uses CIS to connect to the TDS service. Enable CIS by using one of these commands:

- `sp_configure "cis rpc handling," 1`
- `set cis_rpc_handling on`

Restart your machine to activate CIS.

Adaptive Server needs information on connecting to this remote server. UAF supports many login modules. This example uses an Adaptive Server that is already running, to provide authentication and authorization through Adaptive Server login and user roles.

The login to UAF passes an “@” character to its login name, followed by the Adaptive Server name. An authenticated Adaptive Server login name and password grants access to the UAF, based on established authorization Adaptive Serve roles for that login, and on mapped UAF functionality for that role. The UAFv1.5 contains these mappings:

<table>
<thead>
<tr>
<th>Adaptive Server role</th>
<th>Unified Agent role</th>
</tr>
</thead>
<tbody>
<tr>
<td>sa_role</td>
<td>uaAgentAdmin</td>
</tr>
<tr>
<td>sa_role</td>
<td>uaASEAdmin</td>
</tr>
</tbody>
</table>

An Adaptive Server login with sa_role privileges can access any UAF agent, services, and plug-ins that previously granted access using the roles uaAgentAdmin or uaASEAdmin.

You can use two Adaptive Server stored procedures—`sp_addserver` and `sp_addexternallogin`—to connect the TDS remote server to an Adaptive Server. Executing these stored procedures requires a client that can connect to the Adaptive Server database and execute SQL commands. These two commands display the Adaptive Server command line utility `isql`:

```
C:\Sybase>isql -U sa -P Sybase4me -S myserver
sp_addserver MYHOST_UAF
----------
sp_addexternallogin
  MYHOST_UAF,sa,sa@myserver,Sybase4me
----------
exit
```
Using the Adaptive Server SNMP Agent Plug-in

The Sybase Adaptive Server MIB defines the sybRaiseError error trap. This trap can be generated from a stored procedure:

```sql
create procedure snmpSybRaiseError
    @errno int,
    @severity int,
    @state int,
    @spid int,
    @engine int,
    @msg varchar(255) as
declare @user varchar(255)
begin
    select @user = suser_name()
    exec MYHOST_UAF...uaf_invoke_plugin_operation
        "com.sybase.ase.snmp",î2.0.0",1,
        "sybRaiseError",@servername,@errno,@severity,
        @state,@user,@spid,@engine,@msg
end
```

The stored procedure takes a set of arguments and calls the Adaptive Server SNMP Agent Plug-in to generate the SNMP notification. The call to this stored procedure follows this format:

```
$ isql -Usa -P -Smyserver
1> exec snmpSybRaiseError 1,2,3,4,5,"Hello SNMP World"
sybRaiseError
-------------
1
```

For example:

```
exec snmpSybRaiseError 1,2,3,4,5,"Hello SNMP World"
sybRaiseError
-------------
1
```

The stored procedure returns the SNMP table index:

```
(0 rows affected)
(return status = 0)
```
Viewing the SNMP information from NET-SNMP

The error message is sent as an SNMP trap and added to the `aseErrorTable` in the Adaptive Server SNMP Agent Plug-in. You can use an SNMP client to query the table to see whether messages have been logged. For example, an open source SNMP network client application called NET-SNMP could be used to see the reported messages:

```bash
$ snmpwalk -v 1 -c public MYHOST:1498 enterprise
SNMPv2-SMI::enterprises.897.1.1.1.1.1.1 = INTEGER: 1
SNMPv2-SMI::enterprises.897.1.1.1.1.2.1 = STRING: "WEBSPINNER"
SNMPv2-SMI::enterprises.897.1.1.1.1.3.1 = INTEGER: 1
SNMPv2-SMI::enterprises.897.1.1.1.1.4.1 = INTEGER: 2
SNMPv2-SMI::enterprises.897.1.1.1.1.5.1 = INTEGER: 3
SNMPv2-SMI::enterprises.897.1.1.1.1.6.1 = STRING: "sa"
SNMPv2-SMI::enterprises.897.1.1.1.1.7.1 = INTEGER: 4
SNMPv2-SMI::enterprises.897.1.1.1.1.8.1 = INTEGER: 5
SNMPv2-SMI::enterprises.897.1.1.1.1.9.1 = STRING: "Hello SNMP World"
```

```bash
$ snmptable -m SYBASE-MIB -v 1 -c public MYHOST:1498 aseErrorTable
SYBASE-MIB::aseErrorTable

sybErrorIndex sybServerName sybErrorNumber sybErrorSeverity sybErrorState
    sybUserName sybServerProcess sybEngine sybErrorMessage
1 WEBSPINNER 1 2 3 sa 4
5 Hello SNMP World
```

---

Sybase MIB

This is the Sybase MIB file. The values here are described in the sections above.

---Sybase-Common-MIB

This is a Sybase private MIB definition to be used in conjunction with SNMP and CSMI (Client Server Management Interface) tools. It is intended to be a common MIB for all Sybase products.

It is Sybase's intention to make this information freely available and encourage widespread use of this MIB in connection with the agents and adaptors that monitor and administer Sybase products. However, Sybase retains exclusive ownership and title.

-- Confidential Property of Sybase, Inc.
APPENDIX B    SNMP Agent Plug-in

-- (c) Copyright Sybase, Inc., 2002
-- All rights reserved
-- Use, duplication or disclosure by the Government is subject to restrictions
as set forth in subparagraph (c) (1) (ii) of DFARS 52.227-7013 for the DOD and
as set forth in FAR 52.227-19 (a) - (d) for civilian agencies.
-- Sybase, Inc. 5000 Hacienda Drive, Dublin, CA  94568

Notes:
Sybase is a registered IAB number {897} under the enterprises node.

===============================================================================
SYBASE-MIB DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
   Counter32, Gauge32, Integer32, enterprises
   FROM SNMPv2-SMI
   DisplayString, DateAndTime, AutonomousType
   FROM SNMPv2-TC;
sybase   MODULE-IDENTITY
LAST-UPDATED "0206182100Z"
ORGANIZATION "Sybase Inc."

CONTACT-INFO
" Bill Cox
   Sybase Inc.
   5000 Hacienda Drive
   Dublin, CA 94568
   US
   Tel USA (603) 230 7106
   E-mail: bcox@sybase.com"

DESCRIPTION
   "The MIB module to describe objects for Sybase proprietary products"
 ::= { enterprises 897 }

-- ------------------------------------------------
Adaptive Server Enterprise related objects

---------
ase OBJECT IDENTIFIER ::= { sybase 1}

aseObjectsOBJECT IDENTIFIER ::= {ase 1}
aseErrorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF AseErrorEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"A table relating databases and servers present on a host."
::= { aseObjects 1 }

aseErrorEntry OBJECT-TYPE
SYNTAX      AseErrorEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
""
INDEX { sybErrorIndex }
::= { aseErrorTable 1 }

AseErrorEntry ::= SEQUENCE {
    sybErrorIndex    INTEGER,
sybServerName     DisplayString,
sybErrorNumber    INTEGER,
sybErrorSeverity  INTEGER,
sybErrorState     INTEGER,
sybUserName       DisplayString,
sybServerProcess  INTEGER,
sybEngine         INTEGER,
sybErrorMessage   DisplayString
}

sybErrorIndex OBJECT-TYPE
SYNTAX      INTEGER (1..2147483647)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"An index value used to uniquely identify an entry in this table."
::= {aseErrorEntry 1}
sybServerName  OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
"Name of the Adaptive Server Enterprise reporting an error condition"

::= {aseErrorEntry 2}
sybErrorNumber  OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
"Number of error condition being raised."

::= {aseErrorEntry 3}
sybErrorSeverity  OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
"Severity level of error condition being raised."

::= {aseErrorEntry 4}
sybErrorState  OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
"SQL State of error condition."

::= {aseErrorEntry 5}
sybUserName  OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
"Database user name"
Sybase MIB

sybServerProcess OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Server process number (spid)"
 ::= {aseErrorEntry 7}

sybEngine OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Server engine number"
 ::= {aseErrorEntry 8}

sybErrorMessage OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Text message associated with an error"
 ::= {aseErrorEntry 9}

sybASEtraps OBJECT IDENTIFIER ::= { ase 2 }
DESCRIPTION

"An error condition was raised within an instance of ASE. The condition could have been raised either internally, or from within a user written storedprocedure."

 ::= { sybASEtraps 1 }

-----------------------------------------------
APPENDIX C

Tabular Data Stream (TDS) Service

What is TDS?

TDS, or Tabular Data Stream, is an application protocol by which Open Clients and Open Servers exchange information. Applications that use the Sybase Open Client libraries are also TDS applications, as are some (such as Sybase jConnect) that do not use the Open Client libraries.

As part of the Unified Agent Framework, TDS:

1. Starts when you start the UAF (Unified Agent Framework).
2. Reads the TDS port number from its own configuration file. This file is set by another application, such as Agent Management Console.
3. Obtains the Agent RMI (Remote Method Invocation) port number from UAF Agent Services.
4. Starts a TCP/IP socket listener on the designated TDS port number, which waits for login, logout, and remote procedure requests, then:
   - Upon receiving a login request, the service creates a UAF session for that user, using the RMI client interface.
   - Upon receiving a logout request, the service closes the UAF session.
Using the TDS service

- Upon receiving an RPC request, the service processes it and returns a result set, which may contain only the returned status of the request.

5 Shuts down the listener when the session is closed.

Using the TDS service

The TDS service provides one remote procedure call, uaf_list_rpc, which returns information about remote procedures. Table C-1 shows the columns this function returns.

**Table C-1: Columns returned by uaf_list_rpc**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>java.lang.String</td>
<td>Name of RPC</td>
</tr>
<tr>
<td>Arguments</td>
<td>java.lang.String</td>
<td>Arguments for RPC</td>
</tr>
<tr>
<td>Description</td>
<td>java.lang.String</td>
<td>Description of RPC</td>
</tr>
</tbody>
</table>

Table C-2 on page 56 shows the output from this function.

**Table C-2: Output from uaf_list_rpc**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Arguments</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>uaf_list_rpc</td>
<td>&lt;none&gt;</td>
<td>Returns information about remote procedure calls</td>
</tr>
<tr>
<td>uaf_list_plugins</td>
<td>&lt;none&gt;</td>
<td>Returns information about agent plug-ins</td>
</tr>
<tr>
<td>uaf_list_services</td>
<td>&lt;none&gt;</td>
<td>Returns information about services</td>
</tr>
<tr>
<td>uaf_list_plugin_operations</td>
<td>Plugin ID, Plugin Version</td>
<td>Returns information about operations available for a particular plug-in</td>
</tr>
<tr>
<td>uaf_list_plugin_operation_arguments</td>
<td>Plugin ID, Plugin Version, Plugin Operation &lt;operation arguments&gt;</td>
<td>Returns information about the arguments for a particular plug-in operation</td>
</tr>
<tr>
<td>uaf_invoke_plugin_operation</td>
<td>Plugin ID, Plugin Version, Instance Number, Plugin Operation &lt;operation arguments&gt;</td>
<td>Invokes a plug-in operation</td>
</tr>
<tr>
<td>uaf_list_plugin_attributes</td>
<td>Plugin ID, Plugin Version</td>
<td>Returns the list of attributes for a plug-in</td>
</tr>
</tbody>
</table>
### UAF services

A set of remote procedure call (RPC) functions accesses UAF plugins, operations and attributes:

- `uaf_list_services` – lists the UAF services.
- `uaf_list_service_operations` – lists the operations for a service.
- `uaf_list_service_operation_arguments` – lists the arguments for a service operation.
- `uaf_invoke_service_operation` – invokes an operation in a service.
- `uaf_list_service_attributes` – lists the attributes of a service.
- `uaf_get_service_attribute` – gets the value of an attribute of a service.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Arguments</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>uaf_get_plugin_attribute</code></td>
<td>Plugin ID, Plugin Version, Instance Number, Attribute Name</td>
<td>Updates the value of the requested plug-in attribute</td>
</tr>
<tr>
<td><code>uaf_set_plugin_attribute</code></td>
<td>Plugin ID, Plugin Version, Instance Number, Attribute Name</td>
<td>Updates the value of the provided plug-in attribute</td>
</tr>
<tr>
<td><code>uaf_list_service_operations</code></td>
<td>Service ID</td>
<td>Returns information about operations available for a particular service</td>
</tr>
<tr>
<td><code>uaf_list_service_operation_arguments</code></td>
<td>Service ID, Service Operation, &lt;operation arguments&gt;</td>
<td>Returns information about the arguments for a particular service operation</td>
</tr>
<tr>
<td><code>uaf_invoke_service_operation</code></td>
<td>Service ID, Service Operation, &lt;operation arguments&gt;</td>
<td>Invokes a service operation</td>
</tr>
<tr>
<td><code>uaf_list_service_attributes</code></td>
<td>Service ID</td>
<td>Returns the value of the requested service attribute</td>
</tr>
<tr>
<td><code>uaf_get_service_attribute</code></td>
<td>Service ID, Attribute Name</td>
<td>Returns the value of the requested service attribute</td>
</tr>
<tr>
<td><code>uaf_set_service_attribute</code></td>
<td>Service ID, Attribute Name, Attribute Value</td>
<td>Updates the value of the provided service attribute</td>
</tr>
</tbody>
</table>
UAF services

- `uaf_set_service_attribute` – sets the value of an attribute.

These services can be called from these interfaces:
- JDBC
- .NET/ADO.NET
- ODBC
- OLED
- CT_LIB
- DB_LIB

For this reason, syntax is omitted in documenting the services.

**uaf_list_service_operations**

Returns information about operations available from a particular service. The valid service identifiers are listed in Table C-4, though this table is dynamic. Table C-6 uses the identifier RemoteShellService from Table C-4.

Argument – *Service ID*
### Table C-3: Columns returned

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>string</td>
</tr>
<tr>
<td>Description</td>
<td>string</td>
</tr>
<tr>
<td>ReturnType</td>
<td>java.lang.Boolean</td>
</tr>
</tbody>
</table>

### Table C-4: Sample output from uaf_list_service_operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
<th>ReturnType</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>Starts the service</td>
<td>void</td>
</tr>
<tr>
<td>stop</td>
<td>Stops the service</td>
<td>void</td>
</tr>
<tr>
<td>restart</td>
<td>Restarts the service</td>
<td>void</td>
</tr>
<tr>
<td>saveArl</td>
<td>Saves Access Requirement List</td>
<td>void</td>
</tr>
<tr>
<td>saveConfig</td>
<td>Saves the configuration settings of the service</td>
<td>void</td>
</tr>
<tr>
<td>incrementReferenceCount</td>
<td>Increases the reference count by 1</td>
<td>void</td>
</tr>
<tr>
<td>decrementReferenceCount</td>
<td>Decreases the reference count by 1</td>
<td>void</td>
</tr>
<tr>
<td>executeWait</td>
<td>Executes a command asynchronously; that is, the control flow immediately reverts back to before the calling process</td>
<td>com.sybase.ua.util.ProcessInfo</td>
</tr>
</tbody>
</table>

**uaf_list_services**

Returns information about services. Table C-3 shows the columns returned by this function. This service has no arguments.
Table C-5: Columns returned

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Identifier</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Initied (initialized)</td>
<td>java.lang.Boolean</td>
</tr>
<tr>
<td>Started</td>
<td>java.lang.Boolean</td>
</tr>
</tbody>
</table>

Table C-6: Sample output

<table>
<thead>
<tr>
<th>Name</th>
<th>Identifier</th>
<th>Initied (Initialized)</th>
<th>Started</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent service</td>
<td>Agent</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Bootstrap service</td>
<td>BootstrapService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Configuration service</td>
<td>ConfigService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Environment service</td>
<td>EnvironmentDiscoveryService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>File transfer service</td>
<td>FileTransferService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Plugin registration service</td>
<td>PluginRegisterService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>RMI service</td>
<td>RMIService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Remote shell service</td>
<td>RemoteShellService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Security service</td>
<td>SecurityService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Self discovery service</td>
<td>SelfDiscoveryService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Service registration service</td>
<td>ServiceRegistrationService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Session service</td>
<td>SessionService</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>TDS service</td>
<td>TdsService</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

uaf_list_service_operation_arguments

Returns information about the arguments for a particular service operation.

Arguments – Service Id, Service Operation

Table C-7 returns columns from the service identifiers in Table C-4, and the service operations in Table C-6.
APPENDIX C Tabular Data Stream (TDS) Service

<table>
<thead>
<tr>
<th>Table C-7: Columns returned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column name</td>
</tr>
<tr>
<td>Argument name</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table C-8: Sample output from uaf_list_service_operation_arguments RemoteShellService, executeNoWait</th>
</tr>
</thead>
<tbody>
<tr>
<td>Argument</td>
</tr>
<tr>
<td>command</td>
</tr>
<tr>
<td>workingDirectory</td>
</tr>
<tr>
<td>processStatusChangeListener</td>
</tr>
<tr>
<td>processResponseListener</td>
</tr>
<tr>
<td>timeoutSeconds</td>
</tr>
</tbody>
</table>

**uaf_invoke_service_operation**

Invokes a service operation.

**Arguments** – *Service Id, Service Operation, <operation arguments>*

Only one column is returned, the column name; the datatype returned depends on the operation executed.

**uaf_list_service_attributes**

Returns the value of the requested service attribute.

**Argument** – *Service Id*
Table C-9: Columns returned

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>string</td>
</tr>
<tr>
<td>Type</td>
<td>string</td>
</tr>
<tr>
<td>Description</td>
<td>string</td>
</tr>
</tbody>
</table>

Table C-10: Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>modelerType</td>
<td>string</td>
<td>The type of the source modeled. You can set this attribute only once.</td>
</tr>
<tr>
<td>id</td>
<td>string</td>
<td>Service ID.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>Service name.</td>
</tr>
<tr>
<td>mBeanName</td>
<td>string</td>
<td>Service MBean name.</td>
</tr>
</tbody>
</table>

**uaf_get_service_attribute**

Returns the value of the requested service attribute.

*Arguments – Service Id, Attribute Name*

Only one column, Value, is returned; the datatype returned depends on the operation executed.

**uaf_set_service_attribute**

Updates the value of the service attribute provided.

*Arguments – Service Id, Attribute Name, Attribute Value*

This service returns no columns.

**UAF plug-ins**

This set of RPC (Remote Procedure Call) plug-in functions accesses the UAF:

- uaf_list_plugins – lists the UAF plug-ins.
- uaf_list_plugin_operations – lists operations for a plug-in.
• uaf_list_plugin_operation_arguments – lists arguments for a plug-in operation
• uaf_invoke_plugin_operation – invokes an operation for a plug-in.
• uaf_list_plugin_attributes – lists the attributes of a plug-in.
• uaf_get_plugin_attribute – gets the value of a specified attribute of a plug-in.
• uaf_set_plugin_attribute – sets the value of a specified attribute of a plug-in.

**uaf_list_plugins**

Returns information about plug-ins.

This function has no arguments.

**Table C-11:**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>string</td>
</tr>
<tr>
<td>Identifier</td>
<td>string</td>
</tr>
<tr>
<td>Version</td>
<td>string</td>
</tr>
<tr>
<td>Instance</td>
<td>string</td>
</tr>
<tr>
<td>Home directory</td>
<td>string</td>
</tr>
<tr>
<td>Provider</td>
<td>string</td>
</tr>
<tr>
<td>Loaded</td>
<td>java.lang.Boolean</td>
</tr>
<tr>
<td>Enabled</td>
<td>java.lang.Boolean</td>
</tr>
</tbody>
</table>

Sample output:

<table>
<thead>
<tr>
<th>Name</th>
<th>Identifier</th>
<th>Version</th>
<th>Instance</th>
<th>Home Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASE Agent Plugin</td>
<td>com.sybase.ase</td>
<td>15.0.0</td>
<td>1</td>
<td>/opt/sybase...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Provider</th>
<th>Loaded</th>
<th>Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sybase.Inc</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**uaf_list_plugin_operations**

Returns information about operations available to a particular plug-in.
Arguments – Plugin Identifier, Plugin Version

Table C-12: Columns returned

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>string</td>
</tr>
<tr>
<td>Description</td>
<td>string</td>
</tr>
<tr>
<td>ReturnType</td>
<td>java.lang.boolean</td>
</tr>
</tbody>
</table>

Table C-13: Sample output from:

```
uaf_list_plugin_operations"com.sybase.ase","15.0.0"
```

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
<th>ReturnType</th>
</tr>
</thead>
<tbody>
<tr>
<td>loadPlugin</td>
<td>Loads the plug-in</td>
<td>void</td>
</tr>
<tr>
<td>unload</td>
<td>Unloads the plug-in</td>
<td>void</td>
</tr>
<tr>
<td>reload</td>
<td>Reloads the plug-in</td>
<td>void</td>
</tr>
<tr>
<td>saveArl</td>
<td>Saves the ARL (Access Requirement List)</td>
<td>void</td>
</tr>
<tr>
<td>saveConfig</td>
<td>Saves the configuration settings of the plug-in</td>
<td>void</td>
</tr>
<tr>
<td>retrieveServerLog</td>
<td>Returns the server log identified by the attribute ASELogfile</td>
<td>void</td>
</tr>
</tbody>
</table>

uaf_list_plugin_operation_arguments

Returns information about the arguments of a particular plug-in operation.

Arguments – Plugin Id, Plugin Operation

Table C-14: Columns returned

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Argument name</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Description</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Type</td>
<td>java.lang.String</td>
</tr>
</tbody>
</table>

The output for the argument "maxLines", for instance, is the maximum number of lines returned to the function that called it, in datatype java.lang.Integer.

The output for these arguments is dynamic, based on the plug-ins installed in UAF. To display a list of valid arguments, query a running program.
**uaf_invoke_plugin_operation**

Invokes a plug-in operation.

**Arguments** — Plugin Id, Plugin Operation, <operation arguments>

This plug-in returns a value, the datatype of which depends on the operation executed. For example, when the operation invoked returns an integer, the datatype returned is also an integer.

Sample output from:
```
uaf_invoke_plugin_operation "com.sybase.ase","15.0.0",1,getASELogTail,4
```

Value
```
00.00000.00001:2005/11/28 10:23:29:53 server 'bin_iso_1' (ID=50)
00.00000.00001:2005/11/28 10:23:29:53 server Master device size: 30 megabytes, or 15360 virtual pages. (A virtual page is 2048 bytes.)
```

**uaf_list_plugin_attributes**

Returns the value of the requested plug-in attribute.
**UAF plug-ins**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Returned datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Type</td>
<td>java.lang.String</td>
</tr>
<tr>
<td>Description</td>
<td>java.lang.String</td>
</tr>
</tbody>
</table>

**Table C-15: Columns returned**

**Table C-16: Sample output using “com.sybase.ase”, “15.0.0”**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>modelerType</td>
<td>java.lang.String</td>
<td>The datatype of the modeled resource, which can be set only once.</td>
</tr>
<tr>
<td>ID</td>
<td>java.lang.String</td>
<td>The ID of the plug-in</td>
</tr>
<tr>
<td>name</td>
<td>java.lang.String</td>
<td>The name of the plug-in</td>
</tr>
<tr>
<td>version</td>
<td>java.lang.String</td>
<td>The plug-in version, as defined in the agent-plugin.xml file</td>
</tr>
<tr>
<td>instanceNumber</td>
<td>java.lang.String</td>
<td>The number of the instance plug-in</td>
</tr>
</tbody>
</table>

**uaf_get_plugin_attribute**

Returns the value of the requested plug-in attribute.

Arguments – **Plugin Id**

The datatype of the returned value depends on the operation executed.

**uaf_set_plugin_attribute**

Updates the value of the provided plug-in attribute.

Arguments – **Plugin Id, Attribute Name, Attribute Value**

This function returns no columns.
TDS listener client examples

This section shows client examples, and provides information on setting them up.

Setting up examples for the Adaptive Server client

Adaptive Server supports the TDS service as a connection to remote servers. These server names are typically placed in the interfaces file, to establish the remote server port number and the remote host namer.

Only one agent is expected to run on the remote host, so only one TDS listener is expected. In this example, “MYHOST” is part of the server name, with the designation “_UAF” at the end. The TDS service name is, therefore, “MYHOST_UAF.”

*Figure C-1: Interfaces file entry*

```
MYHOST_UAF
    master tcp ether myhost 9998
    query tcp ether myhost 9998
```

Adaptive Server uses Component Integration Services to connect to the TDS service. Enable CIS by using one of these commands:

```
sp_configure "cis rpc handling",
set cis_rpc_handling on
```

You must restart the database to activate CIS.

Adaptive Server requires specific information to connect to the remote server MYHOST_UAF. See “Using the Adaptive Server SNMP Agent Plug-in” on page 47 and “Installing and Configuring Unified Agent and Agent Management Console” on page 3, for more information about authenticating a client connection.
In the first example in “Examples for using the Adaptive Server client” on page 68, the running Adaptive Server provides authentication and authorization through an Adapted Server Enterprise login and user roles. The login to UAF passes “@” to its login name, followed by the Adaptive Server name. An authenticated login name and password grants access to the UAF, based on established authorization roles for that login, and mapped UAF functionality for that role:

<table>
<thead>
<tr>
<th>Adaptive Server role</th>
<th>UAF role</th>
</tr>
</thead>
<tbody>
<tr>
<td>sa_role</td>
<td>uaAgentAdmin</td>
</tr>
<tr>
<td>sa._role</td>
<td>uaASEAdmin</td>
</tr>
</tbody>
</table>

An Adaptive Server login, with an sa_role, can access any UAF agent, services, and plug-ins granted access to the roles uaAgentAdmin or uaASEAdmin.

**Examples for configuring the Adaptive Server client**

You can use any of these methodologies to configure the TDS listener port number:

- The installation program prompts for the TDS port number, and automatically modifies the TDS service XML file.
- The Agent Management Console changes the port number value and restarts the service.
- The UAF administrator edits the TDS service XML file by hand.

Valid values for this port are between 1024 and 32768. This value is stored in the service configuration file, using the property “tdsPort.”

**Examples for using the Adaptive Server client**

Two stored procedures, sp_addserver and sp_addexternlogin, add the TDS remote server connection to Adaptive Server. You can execute these stored procedures using any client that can connect to the Adaptive Server database and execute SQL commands. This example uses the Adaptive Server command line utility isql.

C:\Sybase> isql -U sa -P Sybase4me -S myserver
   sp_addserver MYHOST_UAF
   go
The remote server representing the UAF TDS service is complete, and you can now call other stored procedures. This example shows a stored procedure that lists the available services in the agent:

```sql
sp_addexternlogin MYHOST_UAF, SA, SA@MYSERVER, SYBASE4ME
```

```sql
go
```

```sql
C:\Sybase> isql -U sa -P Sybase4me -S myserver
```

```sql
MYHOST_UAF...uaf_list_services
go
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>ServiceDescription</th>
</tr>
</thead>
<tbody>
<tr>
<td>FileTransferService</td>
<td>1.0.0</td>
<td>Transfers files between client and agent.</td>
</tr>
<tr>
<td>SelfDiscoveryService</td>
<td>1.0.0</td>
<td>Registers the agent and its services and plug-ins with discovery servers.</td>
</tr>
</tbody>
</table>

```sql
MYHOST_UAF...sp_uaf_serviceinvoke "SelfDiscoveryService",discoverAgents
go
```

```
Host Port Version
---------- --------- ----------
MYHOST 9999 1.5.0
MYOTHERHOST 14578 1.5.0
```

(2 rows returned)

The remote server representing the UAF TDS service is complete, and you can now call other stored procedures. This example shows a stored procedure that lists the available services in the agent:

The calling interface to plug-ins requires information, such as version and instance number.

For example, this next example shows a plug-in called “com.sybase.ase” in version 15.0, the first plug-in instance of that name. To display the last 20 lines of the log and return to the calling procedure, enter:

```sql
C:\Sybase> isql -U sa -P Sybase4me -S myserver
```

```sql
MYHOST_UAF...uaf_invoke_plugin_operation
"com.sybase.ase", "15.0.0",1, getASELogTail,20
go
```

ASE Log

```
00:00000:00001:2005/10/13 16:31:31:26 server Master device size: 30 megabytes, or 15360 virtual pages.(A virtual page is 2048 bytes.)
```
Examples for configuring the Adaptive Server client

You can use any of these methodologies to configure the TDS Lister port number:

- The installation program prompts for the TDS port number, and modify the TDS service XML file.
- The Agent Management Console automatically changes the port number value and restarts the service.
- The UAF administrator manually edits the TDS service XML file.

Valid values for this port are between 1024 and 32768. This value is stored in the service configuration file, using the property “tdsPort.”

Reporting functionality and error recovery

A TDS service places all logging messages in the Agent log file, using existing logging services.
Symbols
::= (BNF notation) in SQL statements xiv,
, (comma) in SQL statements xiv
{} (curly braces) in SQL statements xiv
() (parentheses) in SQL statements xiv
[ ] (square brackets) in SQL statements xiv

A
accessing UAF agent 46
acronyms, SNMP definitions 37
Adaptive Server
client, configuring 68
client, using 68
client, configuring examples 70
configuring for generating SNMP notifications 44
security for connections from 41
Adaptive Server client, examples, configuring 70
Adaptive Server client, examples, using 68
Adaptive Server login module 10
Adaptive Server SNMP Agent Plug-in, using with MIB 47
Adaptive Server Sybase MIB 38
adaptors, creating 20
adaptors, discovery 20
adding agent manually 17
agent
properties 19
SNMP 37
Agent Management Console 2, 44
features 15
starting 6
agent plug-in
loading 26, 27
reloading 27
unloading 27
viewing 27
agent service
restarting 26
starting 25
stopping 26
viewing properties 26
Anonymous Login Module 11
anonymous login module 10
architecture, client, Adaptive Server to SNMP 41
attributes
properties, viewing and changing 30
values, updating 30
viewing 29
authentication
ordering security module 23

B
Backus Naur Form (BNF) notation xiii, xiv
BNF notation in SQL statements xiii, xiv
brackets. See square brackets [ ]

C
case sensitivity
in SQL xv
changing network profile 17
CIS
(Component Integration Services) enabling 46
used to connect to TDS 46
client
architecture, Adaptive Server to SNMP 41
connections, authenticating 42
client, Adaptive Server, configuring 70
Index

client, Adaptive Server, using 68
columns output from uaf_list_rpc 56
comma (,) in SQL statements xiv
command
  set cis_rpc_handling on 46
community names 42
Component Integration Services (CIS) 46
components, of UAF SNMP Service 39
configuring
  Adaptive Server 43
  Pluggable Authentication Module 35
  Unified Agent 7
configuring Adaptive Server client 68
connection
  client, authenticating 42
to remote servers 45
to UAF TDS Service 45
control flags, security modules 11
conventions
  See also syntax
  Transact-SQL syntax xiii
  used in the Reference Manual xiii
curly braces ({}) in SQL statements xiv

D
default values, of UAF Agent Service 44
definition of acronyms 37
deleting network profile 17
disconnecting from agent 19
discovery
  Jini Registry 3
  User Datagram Protocol (UDP) 3
discovery adaptors
  creating 20
deleting 21
  managing 20
  setting password 22
  viewing and changing 21

E
enabling RSA encryption 34
encrypted
  message content 42
  passwords 42
encryption
  RSA (Rivest-Shamir-Adleman) 34
  simple login module password encryption 35
error recovery, TDS 70

F
features, Agent Management Console 15
file, Sybase MIB 48
function
  uaf_get_plugin_attribute 66
  uaf_get_service_attribute 62
  uaf_invoke_plugin_operation 65
  uaf_invoke_service_operation 61
  uaf_list_plugin_attributes 65
  uaf_list_plugin_operation_arguments 64
  uaf_list_plugin_operations 63
  uaf_list_plugins 63
  uaf_list_service_attributes 61
  uaf_list_service_operation_arguments 60
  uaf_list_services 59
  uaf_set_plugin_attribute 66
  uaf_set_service_attribute 62
function uaf_list_rpc 56
functionality, TDS 70

I
identity confirmation 10
implementing, Adaptive Server Management
  Information Base (MIB) 37
installing
  Unified Agent 4
  installing Unified Agent 3
isql, command line utility 46

J
Java method
  sybRaiseError 41
Unified Agent and Agent Management Console
Index

Jini Registry discovery  3
Jini server
    starting  4
JMX, definition  37

L
listener client, TDS, examples  67
listener client, TDS, examples, configuring  68
listener client, TDS, examples, setting up  67
loading agent plug-in  26
login module authentication  10
login modules  10
    Adaptive Server login module  10
    anonymous login module  10
    NT Proxy login module  11
    simple login module  10
    UNIX Proxy login module  11
look-up profile, creating  16
look-up services, managing  15

M
message content, encrypted  42
MIB
    definition  37
    implementing  37
    Management Information Base  38
    provided by UAF SNMP Service  40
module, security, creating  22

N
name filter, network, specifying  17
native roles Unified Agent
    roles  12
NET-SNMP, error message  48
network look-up profile
    changing  17
network look-up profile, creating  16
network look-up profile, deleting  17
network plug-in name filter, specifying  17
NTProxy login module  11

O
operations
    managing  31
    viewing  31
outputs, from uaf_list_rpc  56

P
parentheses ()
    in SQL statements  xiv
password
    authentication  10
    encrypted  42
    encryption  9
    encryption utility  33
    setting security module  24
permission  12
permissions, security  13
Pluggable Authentication Module (PAM)  35
plug-in
    loading  27
    reloading  27
    unloading  27
    viewing  27
plug-in, server-based
   pinging  28
plug-ins, UAF, list of  62
pre-installation tasks  3
properties
    viewing agent service properties  26
    viewing and changing in attributes  30
properties, viewing and changing
    security module  23

R
remote procedure call, uaf_list_rpc  56
remote procedure calls, list  57
restarting agent  19
retrieving agent log  19
Rivest-Shamir-Adleman (RSA) encryption  34
role mapping, creating  24
role-based authorization  12
roles
Index

mapping 12
native 12
SNMP login 46
Unified Agent 12
RSA (Rivest-Shamir-Adleman) 34
runtime, Adaptive Server 45

S
sa_role 12
sa_role privileges 46
security
  authenticating client connections 42
  authentication 10
  authorization 12
  encrypted passwords and message content 42
  encryption 9
  for connections from Adaptive Server 41
  for connections from SNMP 42
  managed by UAF 41
  modules, control flags 11
  permissions 13
  protocol version 3 42
  protocol versions 42
  role mapping 12
  two types to monitor SNMP 41
Unified Agent 9
security module
  creating 22
  deleting 23
  ordering authentication 23
  setting password 24
  viewing and changing 23
server-based plug-in
  pinging 28
  restarting 28
  retrieving 29
  starting 28
  stopping 29
server-based plug-in, restarting 28
server-based plug-in, retrieving 29
server-based plug-in, starting 28
server-based plug-in, stopping 29
service, Unified Agent, Windows 5
services, provided by UAF 1
services, TDS, default values 44
set cis_rpc_handling on command 46
setting password, discovery adaptors 22
simple login module 10
  password encryption 35
Simple Network Management Protocol (SNMP) 37
SNMP 37
  acronyms 37
  acronyms for understanding 37
  agent plug-in 37
  Agent Service, default values 44
  configuring Adaptive Server for plug-in 43
  definition 37
  error message sent as trap 48
  login roles 46
  Management Information Base (MIB) 38
  notifications, to configure Adaptive Server for 44
  sample connection 42
  table 38
Tabular Data Stream (TDS) 38
terminology 37
  trap 38
two types of security to monitor 41
UAF service MIBs, list of 40
Unified Agent Framework (UAF) 38
  User-based Security Model (USM) 38
View-based Access Control Model (VACM) 38
viewing information from NET-SNMP 48
SNMP v3 42
SNMPv3 42
square brackets [ ] in SQL statements xiv
starting
  Jini server 4
  Unified Agent 4
starting Agent Management Console 6
stored procedure
  sp_addexternalllogin 46
  sp_addserver 46
  sp_configure, for enabling CIS 46
Sybase MIB file 48
sybRaiseError Java method 41
symbols
  in SQL statements xiii, xiv
syntax conventions, Transact-SQL xiii
syntax, master and query TCP, example of 45

Unified Agent and Agent Management Console
T

table, SNMP 38
Tabular Data Stream (TDS) 55
TDS

ersor recovery 70
functionality 70
listener client examples 67
listener client examples, setting up 67
starts with UAF 55
TDS (Tabular Data Stream) 38
TDS (Tabular Data Stream) 55
service defaults 44
TDS service 56
terminology for SNMP 37
trap
SNMP 38
Trap Message, communicated by UAF SNMP service 41
trap, SNMP
error message sent as 48

U

uaAgentAdmin 46
uaAgentAdmin role 46
uaASEAdmin 46
UAF
agent, accessing 46
ASE SNMP agent plug-in MO 40
definition 37
login 46
login, authenticating 46
manages security 41
SNMP engine components 39
SNMP Service 40
SNMP Service components 39
SNMP Service MIBs 40
SNMP service MIBs, list of 40
SNMP service, communicates Trap Message 41
Unified Agent Framework 38, 39
UAF plug-ins 62
UAF services
remote procedure calls, listed 57
UAF, TDS component of 55
uaf_get_plugin_attribute 66
uaf_get_service_attribute 62
uaf_invoke_plugin_operation 65
uaf_invoke_service_operation 61
uaf_list_plugin_attributes 65
uaf_list_plugin_operation_arguments 64
uaf_list_plugin_operations 63
uaf_list_plugins 63
uaf_list_rpc 56
uaf_list_rpc_outputs 56
uaf_list_service_attributes 61
uaf_list_service_operation_arguments 60
arguments 61
uaf_list_service_operations operations 59
uaf_list_service_operations, function
uaf_list_services 58
uaf_list_services 59
uaf_list_services, sample output 60
uaf_set_plugin_attribute 66
uaf_set_service_attribute 62
Unified Agent
configuring 3, 7
disconnecting 19
installation 4
installing 3
Java Management Extensions 1
restarting 19
retrieving log 19
security 9
SNMP 44
SNMP service entity components 39
starting 4
starting as a service, Windows 5
starting on UNIX 5
viewing and changing properties 19
Unified Agent Framework 1
Unified Agent Framework (UAF), management server 39
Unified Agent Security
authentication 10
authorization 12
encryption 9
unified agents
adding credentials 18
connecting to 18
unified agents, adding manually 17
unified agents, managing 17
Index

United Agents
  discovering  16
UNIX    35
  Proxy login module  11
  starting Unified Agent   5
  updating attribute values  30
    usASEAdmin privileges  46
User Datagram Protocol (UDP)  3
USM (User-based Security Model)  38
USM, definition  37
utility, isql  46
utility, password encryption  33

V

VACM
  definition  37
VACM (View-based Access Control Model)  38