Chapter 2

Overview of LDAP Integration

A directory that implements the Lightweight Directory Access Protocol (LDAP) is the central repository for data shared between the various components in an SRC environment. You can integrate a number of supported third-party directory servers to provide LDAP support.

Topics in this chapter include:

- LDAP Overview on page 21
- Supported Directories on page 23
- Directory Security on page 23
- Provisioning the Directory on page 24
- Naming Directory Entries on page 24
- SDX Directory Schema and Object Model on page 25
- Directory Schema for SRC Software on page 27

LDAP Overview

The LDAP model is a standard that specifies directory access to servers that comply with the following RFCs:

- RFC 2255—The LDAP URL Format (December 1997)
- RFC 2254—The String Representation of LDAP Search Filters (December 1997)
- RFC 2253—Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names (December 1997)
- RFC 2252—Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions (December 1997)
- RFC 2251—Lightweight Directory Access Protocol (v3) (December 1997)

LDAP is optimized to support searching for information that meets specified criteria.

An LDAP directory is the central integration point for the systems that interact with the SRC software, such as network devices and RADIUS servers, and serves as a repository for customer information, service information, policies, and SRC configuration information, including licensing material. For information about how a directory can be deployed in an SRC configuration, see *SRC-PE Getting Started Guide, Chapter 31, Planning an SRC Installation on a Solaris Platform*.

Because a directory is a critical component in your SRC environment, you should have a good understanding of your directory server and of LDAP before using the SRC software. See the documentation for your directory server. This chapter provides information specific to directory configuration for the SRC software.

Directory Availability

Directory redundancy increases the level of availability and performance for an SRC deployment. A number of SRC components, such as the SAE, rely on access to the directory to obtain configuration and provisioning information. To maintain continuous access to the directory, an SDX directory client can be configured to use one directory server as the primary directory and to use any number of backup directories. The SRC software works with multiple servers in the following way:

- The first server specified is the primary or preferred directory server; any other servers comprise an ordered list of backup servers.
- If the primary directory server is not available or fails, the SRC software tries each of the backup servers in turn according to the ordered list. It switches directory connections to the first available backup directory.
- If a backup directory fails, the SRC software again tries each of the directory servers in turn, beginning with the primary server and proceeding through the ordered list. It switches directory connections to the first available backup directory.
- If the primary directory recovers or becomes available, the directory connection switches back to the primary directory server.

For sample deployments that use one or more backup directories, see *SRC-PE Getting Started Guide, Chapter 31, Planning an SRC Installation on a Solaris Platform*.

Directory Updates

When the SAE starts, objects such as policy and service definitions are loaded in to the directory. Directory data for some other objects, such as retailer and subscriber definitions, are loaded only when needed.

An SDX directory client runs in a number of components. Changes to data that is loaded by a directory client, but that is not loaded on an as-needed basis, can be updated for affected components. Therefore, you do not need to manually reload the data in the SDX directory client.

Depending on the configuration for an object, a client can detect data changes and make appropriate updates. In some cases, you can disable directory updates.

All SAEs in a configuration share the same data and receive the same updated directory information. As a result any SAE can manage a subscriber or a service. For example, when you create a new service, the service definition is stored in the directory, all SAEs are notified, and all active subscriptions to the service are adjusted to the new definition.

Supported Directories

You can directly integrate supported directory servers by installing the directory software to meet SDX specifications and then running a script provided by an SRC add-on component. The SRC software provides prepackaged integration with the following directory servers:

- DirX directory server—See *Chapter 6, Integrating the DirX Directory Server*.
- eTrust Directory—See *Chapter 3, Integrating eTrust Directory*
- Oracle Internet Directory —See *Chapter 4, Integrating Oracle Internet Directory*.
- Sun ONE Directory Server—See *Chapter 5, Integrating Sun ONE Directory Server*.

For information about which directory servers have been tested with the SRC software, see the *SRC-PE Release Notes*.

Directory Security

You can help to secure data in your directory by configuring:

- Directory Access on page 23
- LDAPS Directory Connections on page 24

Directory Access

Directories specify different levels of access for users to particular information in the directory. Access control lists define access rights for users and clients.

From the SRC software, you can configure appropriate authorization for operators to access the directory and specific SRC components. Service providers can set up a multilayered access control scheme for operators. For instance, a network operator might be able to create configuration entries for network devices, but not for services or subscribers. See *SRC-PE Subscribers and Subscriptions Guide, Chapter 13, Configuring Subscribers and Subscriptions with SDX Admin*.

All clients that have the credentials of an SRC component are granted only the level of access required. For example, RADIUS requires access to read and compare user passwords that are part of the RADIUS profiles, but does not require access to other user passwords. RADIUS also does not require access to modify, create, or delete the entries.

For detailed information about directory access, see *Chapter 8, Access Control Scheme*.

Directories also provide audit control to track user activity. Audit control lets you trace the changes that a user makes to the directory. Because the SRC software can support directory access for a number of users, you can use a directory audit control mechanism to determine the actions that a user takes on SDX data, such as modifying directory entries.

LDAPS Directory Connections

LDAPS is LDAP that uses Secure Sockets Layer (SSL) to secure communications between an LDAP client and server. Most directories, including DirX directory server, eTrust Directory, Oracle Internet Server, and Sun ONE Directory Server support LDAP through SSL.

The SAE supports LDAPS connections to the directory server for components within the SAE. The SAE can provide simultaneous LDAP and LDAPS connections for different components. LDAPS connections are useful for protecting confidential data, such as attributes that contain passwords and keys. For public data that does not require the security of SSL, you can configure LDAP rather than LDAPS.

For information about configuring LDAPS connections, see *Chapter 7, Configuring LDAPS for SRC Components*.

Provisioning the Directory

You can provision the directory by:

- Using SRC applications such as SDX Admin and Policy Editor.
- Importing SDX directory files that are in LDAP Data Interchange Format (LDIF) into other programs.

  If you plan to import data into the SDX directory, you should have a good working knowledge of the SDX schema. See the LDAP schema documentation in the SRC software distribution in the folder */SDK/doc/ldap/* or on the Juniper Networks Web site at

  http://www.juniper.net/techpubs/software/management/sdx

- Using an external operations support system (OSS) to provision all or part of the directory information directly through the LDAP interface. Both mechanisms must follow the SDX LDAP schema.

Naming Directory Entries

When you add an entry to the directory, an asterisk (*) in the name can create more than one match and result in none of the associated configurations being used. Also, the directory does not distinguish between upper case and lower case characters in object names.

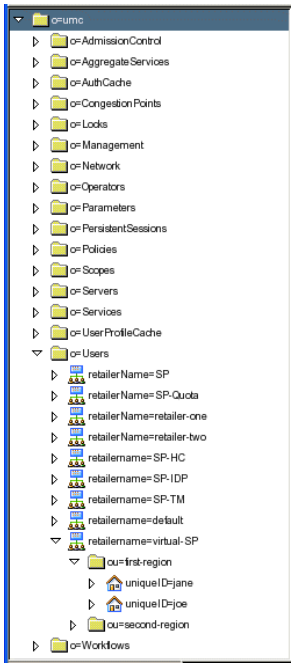
When you add an entry to the directory, do not use an asterisk (*) or other non-alphanumeric characters in the name. Do not specify object names that are the same but differ only in case use in the name. For example, do not use myrouter and MyRouter. Java applications and the enterprise service portals do not handle dots (.) and slashes (/) in subscriber names. When you enter the name of a subscriber, including a subscriber folder or a retailer name for a subscriber, do not use a dot or slash in the subscriber name.

SDX Directory Schema and Object Model

The SDX directory schema is based on X.500/LDAP standards and the Common Information Model version 2.5 (CIM 2.5) schemas. The CIM provides definition of management information for systems, networks, applications, and services. The SDX schema extends the CIM to provide elements for modeling services; residential, enterprise, and retail customers; policies; network elements; and others.

The directory object model represents the way objects are stored in a directory. An object comprises data that is stored as entries and organized into a hierarchical structure called a directory information tree (DIT). A DIT contains a number of other trees called subtrees. Figure 4 shows the top-level objects, as well as some subordinate objects in the Users folder in the SDX directory tree as it appears in SDX Admin.

Figure 4: Directory Tree as Displayed by SDX Admin



Each entry has a number of attributes—special characteristics that provide information about the entry. An attribute can also be referred to as a property.

Each entry has an attribute to specify the name for the entry. A name for an entry must be unique within a specified level in the tree hierarchy; for example, each retailer name must be unique with the Users folder, as illustrated in Figure 4.

Naming Convention for Entries

The name for an entry can be expressed as either a relative distinguished name (RDN) or a distinguished name (DN). The RDN identifies a unique entry at one level in the directory tree. Each RDN identifies an attribute type with the associated value. The following list shows sample RDNs from Figure 4:

o = umc
o = Users
retailername = virtual-SP
ou = first-region
unique-id = joe

NOTE: Do not use the "#" character in DNs. It can cause various problems.

A DN is a comma-separated sequence of hierarchical entry names in the tree, concatenated from a specified entry backward to the base, or root, of the tree structure. In contrast to the RDN, the DN for an entry is unique within the entire directory. Each entry in the directory is identified and can be located by its distinguished name (DN). The DN for subscriber Joe would be the following:

unique-id = joe, ou = first-region, retailername = virtual-SP, o = Users, o = umc

NOTE: Throughout the SRC documentation, in text we show the elements of a DN separated by comma/space pairs. We do this for readability. The SRC software and the LDAP specifications require acceptance of the space, but the space is not necessary.

A base DN is the DN of an object that serves as the starting point for a directory search. For the directory as a whole, the base DN is *o = umc* for a default installation of the SRC software; it is the root object of the tree. For a search of policies, the base DN is the following:

o = policies, o = umc

A bind DN is the DN of a login to the directory. This DN must be entered (like a username) with a password to log in to the directory. In the SRC software, you use the bind DN and a password when you access the directory; for example, when you start SDX Admin to view or modify the contents of the directory.

Table 5 lists some of the common DN attribute types.

Table 5: Common DN Attribute Types

Attribute Type Abbreviation	Attribute Type Definition
cn	Common name
o	Organization name
ou	Organizational unit name (In the directory for SDX the organizational unit is typically a directory.)
uid	User ID

Directory Schema for SRC Software

An LDAP schema defines the content and structure of the directory tree. It determines the types of entries that can exist in the directory, the organization for entries, and the relationships between the different types of entries. The directory schema for the SRC software includes entries primarily for management configuration, network device configuration, policies, services, retailers or providers, and subscriber profiles.

Object Classes

Object classes define the different types of entries that can exist in the directory; each entry in the directory belongs to one or more object classes. An object class contains specified attributes to define the characteristics of the entry. For information about attributes, see *Attributes* on page 30.

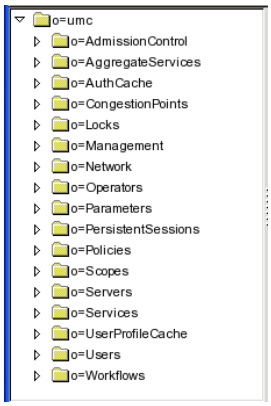
The following sections provide information about the objects in the SDX directory:

- Objects Representing Folders on page 27
- Subscriber Objects on page 28
- Service Objects on page 29
- Subscription Profile Objects on page 29
- Policy Objects on page 29
- Network Device Objects on page 29
- Configuration and System Management on page 30

Objects Representing Folders

Folders divide the DIT into logical subtrees. The content prefix of the SDX tree is *o = umc*. The tree is divided into subtrees, such as those for subscribers and service profiles, services, networks, and policies. Figure 5 shows the first-level folders under *o = umc* that are created during the setup of the directory. You can create the second-level folders by using SDX Admin.

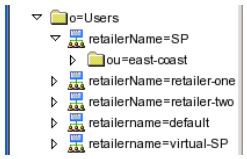
Figure 5: First-Level Folders



The standard object classes organization and organizationalUnit divide trees into subtrees. In some cases, these objects classes do not provide all the attributes required by an entry. An auxiliary object class (an object class that supplies additional information to augment structural object classes) supplies the additional attributes. This moreInformationAuxClass can be attached to the organization object class and the organizationalUnit object class.

You use the object class organizationalUnit to create folders under first-level folders. For example, in Figure 6, the *ou = east-coas*t folder is an organizationalUnit object.

Figure 6: Sample Retailer Folders



The DN for this folder is:

ou = east-coast, retailerName = SP, o = Users, o = umc

Subscriber Objects

The directory provides object classes for the categories of subscribers supported by the SRC software, such as:

- Residential users—umcUser
- Enterprises—umcEnterprise
- Sites—umcSite
- Retailers—umcRetailer
- Routers—umcRouterSubscriber

Folders under a umcRetailer object provide a convenient way to organize groups of subscribers. These folders can use the umcSubscriber auxiliary object class.

Subscriber objects are stored under *o = Users, o = umc*.

Service Objects

The umcService object class models is the base class in the service hierarchy. At a high level, SRC provides object classes for services.

Services are also referred to as SSP services (for residential and enterprise users)—sspService.

Service objects are stored under *o = Services, o = umc*. Services can also be stored under *l = < locality >, o = Scopes, o = umc*.

The SRC software supports parameter substitution for services. Parameter substitution requires the attachment of the auxiliary object class parameterAuxClass to an sspService object and to a locality if the sspService object is configured within a scope.

Subscription Profile Objects

When a subscriber subscribes to a service, the SDX subscription component creates an object for the subscription profile from the object class umcServiceProfile. The umcServiceProfile subscription object is created as a subordinate (child) of the subscriber object in the tree. The SRC software supports parameter substitution for service subscriptions, which means that the auxiliary object class parameterAuxClass can be attached to an instance of sspServiceProfile.

Subscriptions are stored under entries subordinate to *o = Users, o = umc*.

Policy Objects

The policy information model for SRC software is based on the Policy Core Information Model (PCIM) that is mapped to the Policy Framework LDAP core schema by the IETF. The SRC software extends this model to produce a policy model that is very close to the one that routers and other network devices use. A policy group object consists of one or more policy lists, which contain one or more policy rules. A policy rule consists of policy conditions and policy actions. The objects policy group, policy list, and policy rules are mapped to structural object classes. Each of these classes is derived from the object class policy.

Policy objects are stored under *o = Policies, o = umc*.

We recommend that you define policies in the directory before you create service objects. Service objects reference policies. The service definition interface should provide LDAP search functionality that retrieves all available policies.

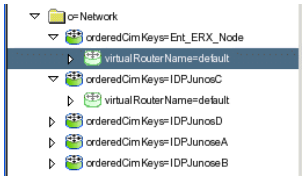
Network Device Objects

Physical network elements are modeled with the CIM Chassis object classes. The object class dlm1Chassis is used for any Juniper Networks equipment, SDX devices, and network devices in the connection path.

Some SRC components, such as the SAE, require additional information about the router, such as virtual routers or interface classifications. The SDX object class umcVirtualRouter is a structural object class that represents virtual routers on Juniper Networks routing platforms. The umcClassificationProfile is an auxiliary object class that is attached to the structural object class.

Network devices are stored in the folder *o = network, o = umc*. Virtual routers are stored subordinate to network devices. Figure 7 shows a sample network folder that contains a number of network devices.

Figure 7: Sample Network Folder



Congestion points in a connection path are modeled by the object class networkInterface, which is subordinate to network device objects in the directory tree. The network devices used for grouping the congestion points are stored in the folder *o = AdmissionControl, o = umc*.

Configuration and System Management

Some of the SDX configuration information, such as license configuration data, is stored in the directory.

The CIM object class dlm1UnitaryComputerSystem represents the hosts on which SAE and system management components, such as an SNMP server, are installed. For management components, the CIM object class dlm1UnitaryComputerSystem replaces the object class umcHost.

The dlm1UnitaryComputerSystem object class stores an IP address in the CIM attribute dlmIdentifyingDescriptions. The location (for example, POP A) is specified in dlmOtherIdentifyingInfo.

All configuration and management objects are stored under *o = Management, o = umc*.

Attributes

An attribute contains a characteristic and the values for that characteristic. Attributes for an object class can be required or optional. An attribute type provides the syntax, sorting, and comparison rules to be used for an attribute.

The following example shows the characteristics and values for the sspType attribute:

- Description—Specifies the provider type (content provider or Internet service provider or others).
- Object identifier—1.3.12.2.1107.1.3.101.10.4.35

- Attribute syntax—Directory String
- Equality matching rule—Case Ignore Match
- Multivalued—False

Structure Rules

DIT structure rules are rules specific to an object class; they specify the location of the object class in the DIT and a name form which identifies naming attributes for an object class. Structure rules state which object classes can be located superior (as a parent) and subordinate (as a child) to other object classes. For example, service profiles are subordinate to users. The DIT structure rules prevent the addition of entries that belong to an object class in an unsupported location in the DIT. If you try to add an entry in an unsupported location, you receive an error message.

The structure rules are used to model dependencies in the DIT. For example, structure rule (SR) 1—organizationalNameForm—allows the creation of the root directory *o = umc*.

Content Rules

A DIT content rule defines which attributes an entry for a specified object class can contain, such as:

- Mandatory attributes that an entry must contain
- Optional attributes that an entry can contain
- Auxiliary object classes that can be associated with the object class
- Optional attributes from the structural and auxiliary object class definitions that an entry must not contain

Where to Find More Information About the Object Model and Directory Schema

The SRC software provides detailed documentation of the object model and the directory schema, including graphical representations of the schema models. See the LDAP schema documentation in the SRC software distribution in the folder */SDK/doc/ldap/* or on the Juniper Networks Web site at

http://www.juniper.net/techpubs/software/management/sdx

The documentation for the SDX object model and schema provides the following topics:

- Attribute types—Lists the attributes types and the associated object classes that appear in the directory.
- Object classes—Provides a link to the directory that contains HTML files that describe each object class. You can also view a list of object classes from the Attribute Types page.
- Content rules—Lists and describes the content rules for structural object classes and the associated auxiliary classes.
- Structure rules—Lists and describes the structure rules for the schema and provides examples of how structure rules are used.
- Name forms—Provides a link to the directory that contains HTML files that describe each name form. You can also view a list of name forms from the structure rules page.
- Schema models—Provides links to graphical representations of the schema models for network, operator, policy, services, and user in GIF and PDF formats.

The SRC software also provides sample data files in LDIF, an easy-to-read format. The location of the LDIF files on your system depends on the directory integrated with the SRC software. Table 6 lists the location of the LDIF files for the various directories.

Table 6: Location of LDIF Files

Directory Server	Directory That Contains LDIF Files
DirX directory server	<*DIRX_HOME*>*/customize/data* where <DIRX_HOME> is the DirX home directory
eTrust Directory	*/opt/UMC/conf/etrust/sdx_ldif*
Oracle Internet Directory	*/opt/UMC/conf/OID*
SUN ONE Directory Server	*/opt/UMC/conf/iDS*

For detailed information about LDAP, see the documentation for your LDAP server.