COBALTADM-1.0

Cobalt Administration Guide

Isode

Table of Contents

1 Software version

This guide is published in support of Cobalt 1.0. It may also be pertinent to later releases. Please consult the release notes for further details.

2 Readership

This guide is intended for two classes of administrator:

- System administrators setting up Cobalt to provision data in an LDAP directory.
- Data administrators using Cobalt to manage data.

3 How to use this guide

It is recommended that all administrators read: Chapter 1, *Isode Cobalt Overview*. System administrators then need to use: Chapter 2, *Cobalt for System Administrators*. Data administrators can skip to: Chapter 3, *Cobalt for Domain Administrators*.

4 Typographical conventions

The text of this manual uses different typefaces to identify different types of objects, such as file names and input to the system. The typeface conventions are shown in the table below.

Object	Example
Applications	Cobalt
File and directory names	*/var/log/isode/cobalt*
Program and macro names	isode.cobalt
GUI elements	**Label**, **Menu**, **Menu Item**, **Sub-Menu**
User input	`hello!`
Cross references	see Section C.1, “RFCs”
Additional information to note, or a warning that the system could be damaged by certain actions.	Notes are additional information; cautions are warnings.

Note: This is an example of a note.

5 File system place holders

Where directory names are given in the text, they are often place holders for the names of actual directories where particular files are stored. The actual directory names used depend on how the software is built and installed. All of these directories can be changed by configuration.

Certain configuration files are searched for first in *(ETCDIR)* and then *(SHAREDIR)*, so local copies can override shared information.

The actual directory defaults vary, depending on whether the platform is *Windows* or *Unix*. The following table provides the platforms-specific defaults.

Name	Place holder for the directory used to store...	Windows	Unix
(ETCDIR)	System-specific configuration files	*C:\Isode\Cobalt\etc*	*/etc/isode/cobalt*
(SHAREDIR)	Configuration files that may be shared between systems	*C:\Program Files\Isode\Cobalt\share*	*/opt/isode/cobalt/share*
(BINDIR)	Programs run by users	*C:\Program Files\Isode\Cobalt\bin*	*/opt/isode/cobalt/bin*
(SBINDIR)	Programs run by the system administrators	*C:\Program Files\Isode\Cobalt\bin*	*/opt/isode/cobalt/sbin*
(LIBDIR)	Libraries	*C:\Program Files\Isode\Cobalt\bin*	*/opt/isode/cobalt/lib*
(DATADIR)	Storing local data	*C:\Isode\Cobalt*	*/var/isode/cobalt*
(LOGDIR)	Log files	*C:\Isode\Cobalt\log*	*/var/log/isode/cobalt*

6 Support queries and bug reporting

A number of email addresses are available for contacting Isode. Please use the address relevant to the content of your message.

- For all account-related inquiries and issues: customer-service@isode.com. If customers are unsure of which list to use then they should send to this list. The list is monitored daily, and all messages will be responded to.
- To provide keys necessary to activate products, send the generated string to support@isode.com along with information on what is being evaluated or what has been purchased.
- For all technical inquiries and problem reports, including documentation issues from customers with support contracts: support@isode.com. Customers should include relevant contact details in initial calls to speed processing. Messages which are continuations of an existing call should include the call ID in the subject line. Customers without support contracts should not use this address.
- Customers may also submit support queries through the customer section of the Isode web site using the URL provided. Customers with silver or gold support may also submit support queries by telephone.

- For all sales inquiries and similar communication: sales@isode.com.

Bug reports on software releases are welcomed. These may be sent by any means, but electronic mail to the support address listed above is preferred. Please send proposed fixes with the reports if possible. Any reports will be acknowledged, but further action is not guaranteed. Any changes resulting from bug reports may be included in future releases.

Isode sends release announcements and other information to the Isode News email list, which can be subscribed to from the address: https://www.isode.com/company/contact.html.

7 Export Controls

Cobalt uses *TLS* (Transport Layer Security) to encrypt data in transit. This means that Cobalt is subject to UK Export Controls. For some countries (currently EU, US, CA, AU, NZ, CH, NO, JP) these Export Controls can be handled by administrative process as part of evaluation or purchase. For other countries, a special Export License is required. This can be applied for only in context of a purchase order for Cobalt.

The TLS feature of Cobalt is enabled by a TLS Product Activation feature. This feature may be turned off, and Cobalt without this TLS feature is not export controlled. This can be helpful to support evaluation of Cobalt in countries that need a special export license.

Cobalt is used to administer sensitive data and so Isode strongly recommends that all operational deployments of Cobalt use the export-controlled TLS feature. You must ensure that you comply with these Export Controls where applicable, i.e. if you are licensing or re-selling Isode products. All Isode Software is subject to a license agreement and your attention is also called to the export terms of your Isode license.

Chapter 1 Isode Cobalt Overview

This chapter gives an overview of Isode Cobalt.

1.1 About Cobalt

Cobalt is a server, controlled by a web interface, for provisioning users and roles in an LDAP directory. It enables easy addition and management of information to support directory white pages, XMPP deployments, email deployments and military messaging deployments.

1.2 Information Provisioned by Cobalt

1.2.1 Domains

Cobalt groups information by domains (e.g., “example.com”). The term “domain” is used to mean Internet domains, typically registered in the *Domain Name System*. A Cobalt service can manage one or more domains. Cobalt names entities within domains (e.g., joe.soap@example.com) and ensures entity uniqueness within the domain.

1.2.2 Users: White Pages; XMPP; email; Military Messaging

A core Cobalt service is to provision users. This can be in support of XMPP, email or military messaging services or simply as a generic white pages provisioning to provide directory lookup and support by other applications.

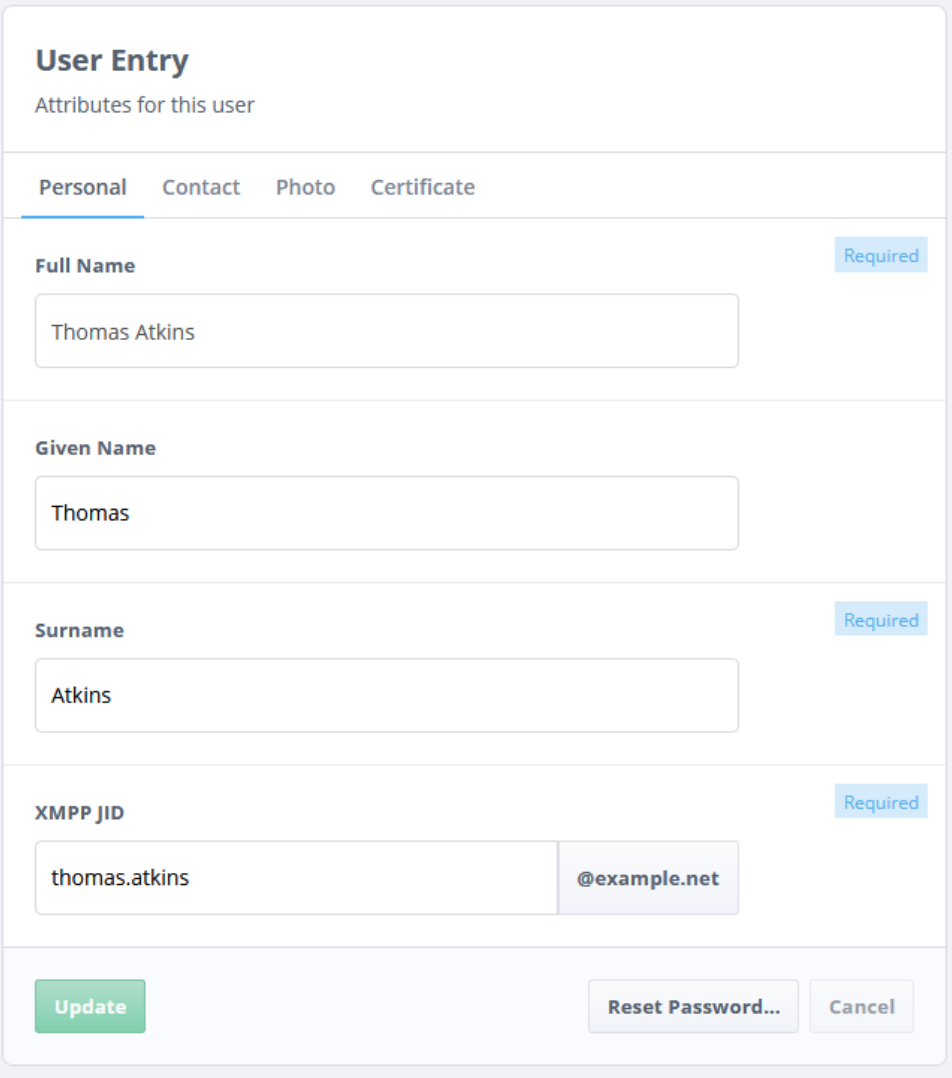
User management capabilities include:

- User creation
- Password assignment and reset
- Delete / Restore / Purge
- Account locking
- White pages information, including contact information and pictures
- X.509 PKI Certificates

1.2.3 XMPP Support

Provisioned users may be configured as XMPP users and a special attribute may be used for JID (Jabber ID). If this is chosen, the administrative UI uses XMPP terminology as shown in the figure below for Figure 1.1, “XMPP User Entry”.

Figure 1.1. XMPP User Entry



1.2.4 Email Support

Provisioned users may have a standard IMAP mailbox to support email service in conjunction with servers such as *M-Box*. Cobalt provides a number of options in support of this:

User management capabilities include:

- Primary email address of the user's mailbox.
- Alternate email addresses can be configured, which will be delivered to the same mailbox.
- IMAP Mailbox quota.
- Redirect option, so that the user's messages can be redirected to another address.
- Ensuring that all email addresses in the domain are unique.

Cobalt also provides provisioning options to be used in conjunction with Isode's *M-Switch* product to provide a full email services:

- **Redirections**- Enables configuration of addresses to point at other email addresses, which may be in the same or different domain. For example postmaster@example.com could be redirected to a user or distribution list.
- **Distribution lists**- Provision of flexible distribution lists. List members can be email addresses (users, redirections or distribution lists) provisioned in Cobalt or any other

email address. Controls are provided on who can submit messages to the list and information header addition following RFC 2369 is supported. There are also controls of military priority on distribution list expansion.

1.2.5 Military Messaging Support

Cobalt provides a range of capabilities to support formal Military Message Handling Systems (MMHS), with capabilities oriented towards support of systems using Isode's *Harrier*, *M-Box* and *M-Switch* products. Capabilities provided include:

- **Role based User Agents**. A key characteristic of MMHS is that mailboxes are role based, with multiple users able to access a role and users able to access multiple mailboxes. Cobalt enables configuration of role based mailboxes (UAs), which have mailbox and white pages information equivalent to the email service described above. A role based UA will also have a list of users that can occupy the role, which may be from the same domain or a different domain. A common approach will be for users to have a different domain, with users provisioned to have an email service and to be able to access an MMHS service.
- **ACP 127 Support**. UAs can be configured with ACP 127 attributes (RI and PLA) and also with line length, character set (ITA2/IA5) and attachment restrictions. Harrier will enforce these restrictions, which is important for messages that are transmitted using ACP 127.
- **Capability Checking**. Configuration of additional message capabilities of maximum message size and control of S/MIME signing/encryption.
- **Redirections**. See above (Section 1.2.4, "Email Support")
- **Military Address Lists**. Military address lists are similar to email address lists, but list members are split into Action and Info recipients, in support of MMHS processing. Recipient configuration follows ACP 133 schema.
- **Profiled Addresses (Organizations)**. MMHS messages flow between organizations. A message sent to a Profiled address will be distributed by a profiler, such as Isode's M-Switch Profiler, to role based mailboxes. Cobalt allows provision of such profiled addresses that represent organizations. It also allows configuration of roles that are allowed to send messages on behalf of an organization, which Harrier picks up and presents valid choices to the role.
- **Organization/Role address type**. To facilitate Harrier communication between Roles (for some functions) and Organizations (for released messages), Cobalt enforces User/Role/Organization type on managed entities. This is important, as it enables a distribution list to contain organizations only (or roles only) and then appear in address book as an organization (or role).
- **Routed UAs**. A routed UA is an address that belongs to a domain, but is not processed locally. It can be routed by M-Switch to a channel, domain or routing nexus. This is important to support domains where mailboxes reside at multiple locations – "flat domain" model.

1.3 Directory Support

1.3.1 M-Vault core server

Cobalt works with a primary M-Vault Server, which holds Cobalt's own configuration. Typically, this single directory server will also hold the data for all of the managed domains. For all configurations, this server needs to be present to hold Cobalt configuration information.

1.3.2 Additional Directory Servers

Cobalt can access data in other directory servers, so that domain information can be configured in multiple LDAP directory servers. This allows one Cobalt instance to manage domains with different purposes and in different directories.

In order to manage data in a directory, the schema set out in Appendix A, *Schema used by Cobalt* must be supported. M-Vault supports this schema.

1.3.3 Active Directory Support

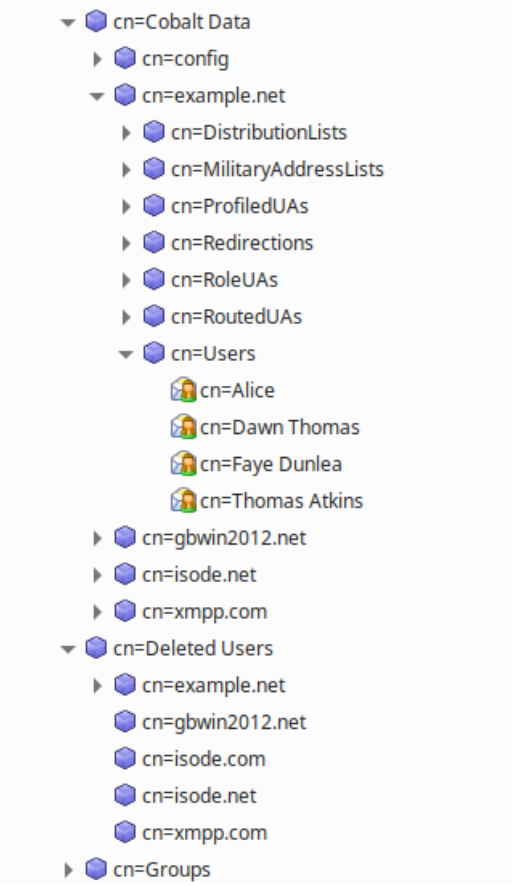
For a domain supporting *Users* only, Cobalt can access *Microsoft Active Directory* with no schema changes. For such a domain, Cobalt cannot add or modify users, but can view the users that are configured in Active Directory.

This setup is important for support of MMHS where users are provisioned in Active Directory. Cobalt can then be used to configure Role Based User Agents, where the role occupants are users configured in Active Directory. This enables use of Cobalt for MMHS configuration, while using Active Directory provisioned users and authentication.

1.3.4 DIT Layout

For each directory used by Cobalt, a selected point (*Cobalt data* in Figure 1.2, “Directory Structure”) in the Directory Information Tree (DIT) is configured to hold the data that Cobalt manages. Cobalt uses DIT structure to separate information for each domain. Different types of information object for each domain are given separate subtrees. This deep hierarchy is chosen to enable easy inspection with a DIT browser and to facilitate migration of selected Cobalt data.

Figure 1.2. Directory Structure



1.3.5 Deleted Users

Deleted Users entries are moved into a separate part of the DIT from active users, which enables deleted users to be restored. It also allows Cobalt to warn when a new user's email or XMPP address conflicts with a deleted user. This allows all Cobalt configured users to be searched from a single point in the DIT which does not include deleted users (as illustrated in the figure above).

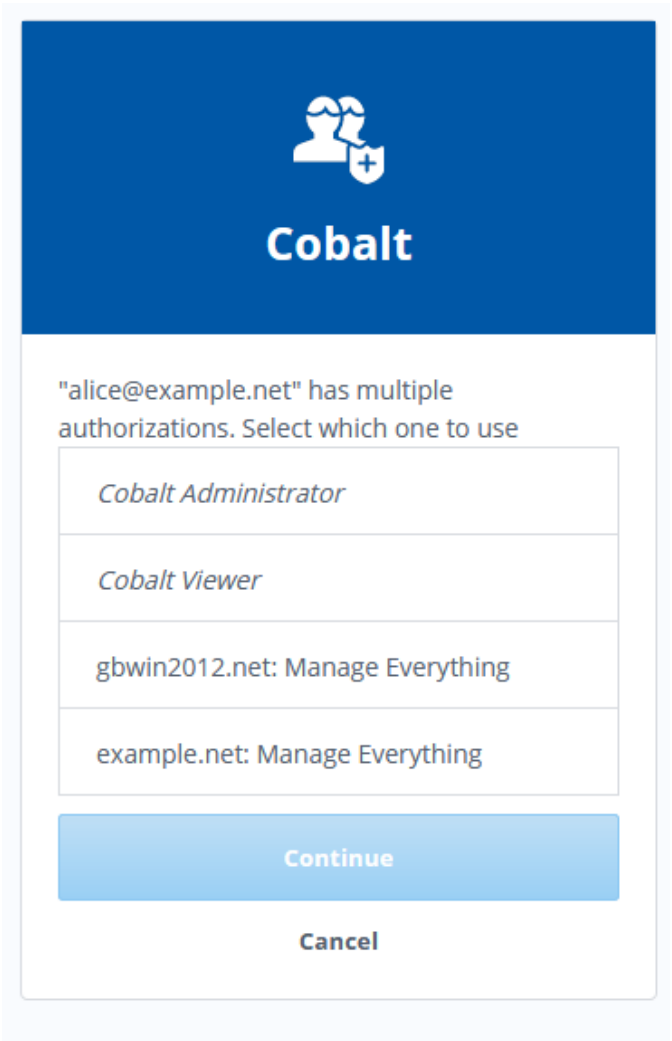
1.4 Roles and Access Control

1.4.1 Cobalt Server Access to Directory

Cobalt server binds to the primary M-Vault server and to other directories as a special privileged user, that is configured as part of setup. Cobalt requires user authentication of any user accessing the Cobalt service.

Cobalt maintains role based access control, recording which roles a given user has access to. When a user authenticates, a user with single role will automatically be made active in that role. A user with rights to multiple roles will be given a choice of roles as shown in Figure 1.3, "Select Authorization Role". A user can be active in only one role for a session.

Figure 1.3. Select Authorization Role

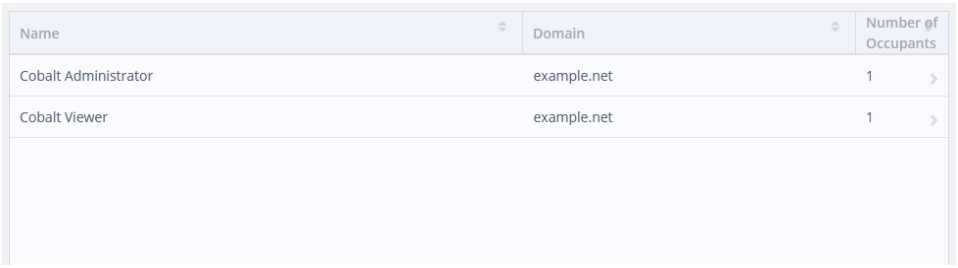


1.4.2 Cobalt Administrator Roles

Cobalt has two role types for its administration (see Figure 1.4, “Administrator Roles”):

- **Cobalt Administrator**. Full access to all Cobalt administrator functions.
- **Cobalt Viewer**. Can see Cobalt configuration, but no rights to modify.

Figure 1.4. Administrator Roles



A Cobalt Administrator can assign users from any domain to either of these roles. A user not assigned to one of these roles has no access to Cobalt configuration.

1.4.3 Domain Administrator Roles

For each domain created by Cobalt, the following role types as shown in figure Figure 1.5, “Domain Administrator Roles” are supported for each domain:

- **Manage Everything**. Full rights for the domain, including management of domain administrators.
- **Users Manager**. Can add, delete and modify users.
- **Roles Manager**. Can add, delete and modify other Cobalt managed information for the domain.
- **Users and Roles Viewer**. Can view information for the domain.

Note that a Cobalt Administrator can create and delete domains and manage the domain administrators. No other access to the domain information is granted.

Figure 1.5. Domain Administrator Roles

Chapter 2 Cobalt for System Administrators

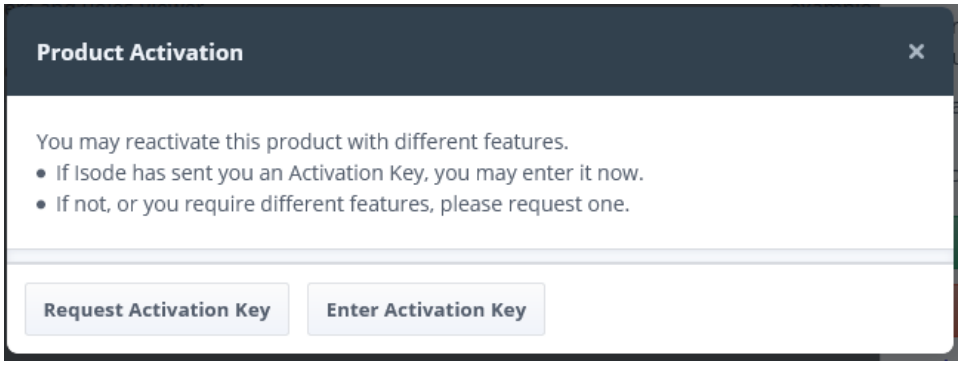
2.1 Cobalt Install and Initial Setup

The Cobalt installation process will lead to the Cobalt server running on port *8001* with access from a Web browser. The HTTP URL for accessing from a local system will be *https://localhost:8001*. The server will bootstrap itself with an auto generated certificate to offer HTTPS services. The browsers will display the page as insecure and give an option to add security exception. You will be able to set it up with a certificate trusted by the browsers and issued by trusted *CA* later (see Section 2.2.2, “TLS Configuration”).

2.1.1 Product Activation

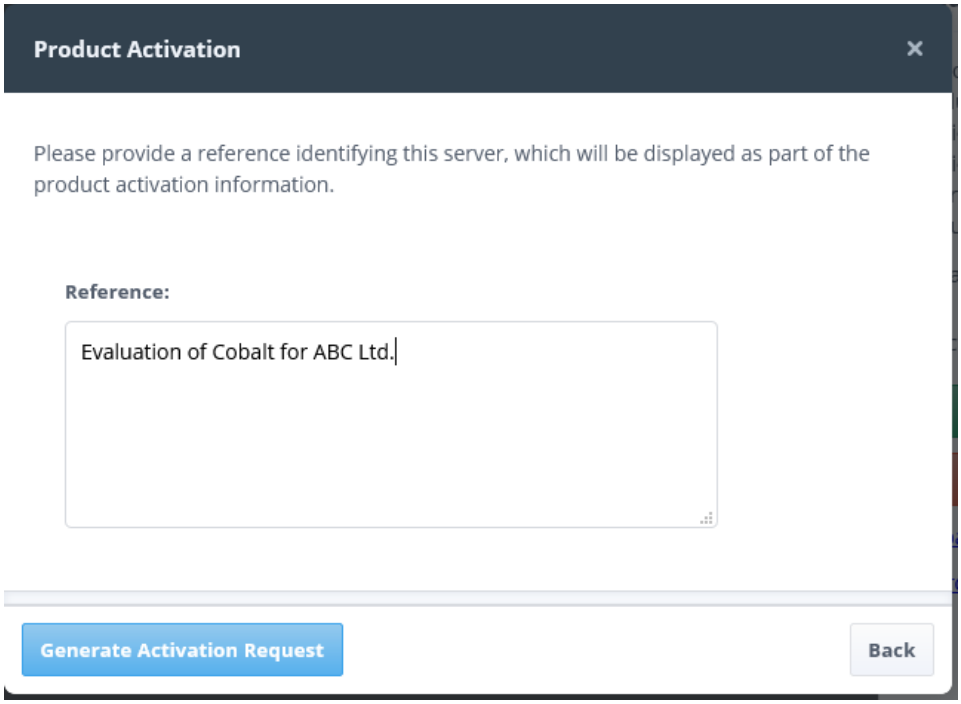
The first interaction with Cobalt is Product Activation (see Figure 2.1, “Product Activation”). A simple dialogue will lead to generation of an activation request string, which should be sent to support@isode.com, along with evaluation or purchase information.

Figure 2.1. Product Activation



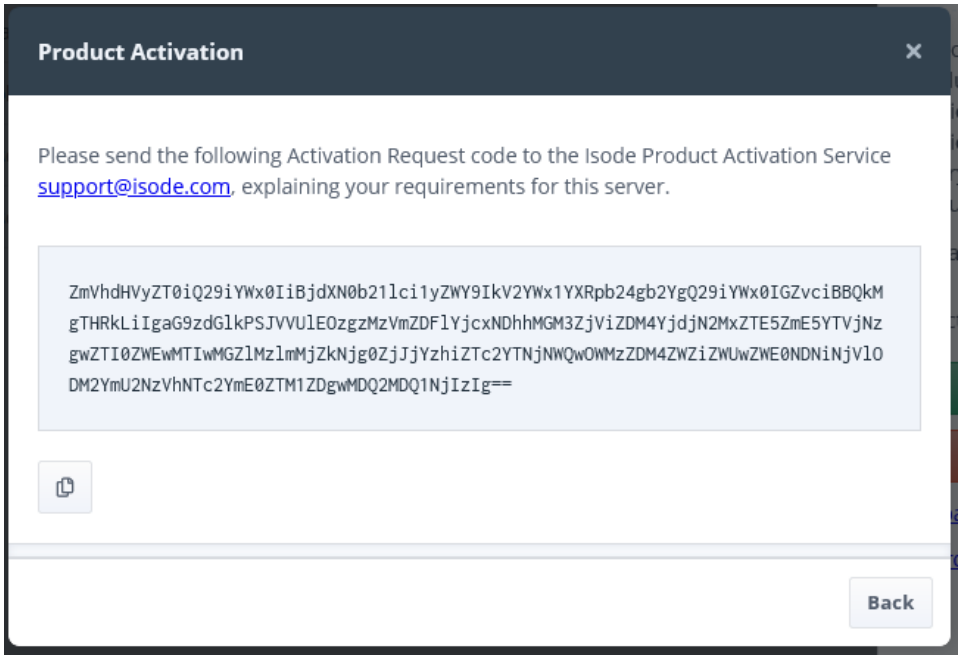
To request a key, you will be prompted to enter a reference for the request as shown in Figure 2.2, “Product Activation Request”. This reference value is a free form string and will be included inside any activation that is issued in response, so that you can use it to identify which server, department, etc. the request was for.

Figure 2.2. Product Activation Request



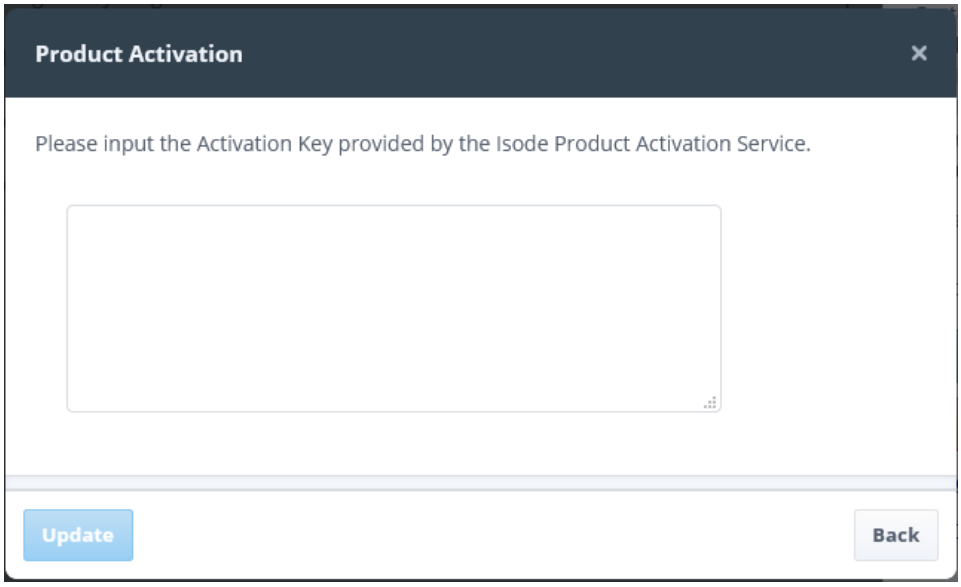
A request will be generated (see Figure 2.3, “Generated Request”) that should be sent to Isode in order to get the activation key.

Figure 2.3. Generated Request



An activation key will be returned by Isode, which should be input to Cobalt. Press the **Back** button to return to the landing page (Figure 2.1, “Product Activation”). Press the **Enter Activation Key** button to paste the activation key in the text box as shown below.

Figure 2.4. Paste Product Activation Key



2.1.2 M-Vault Pre-Requisites

When operating with M-Vault R18.0, the following needs to be set up prior to use of Cobalt.

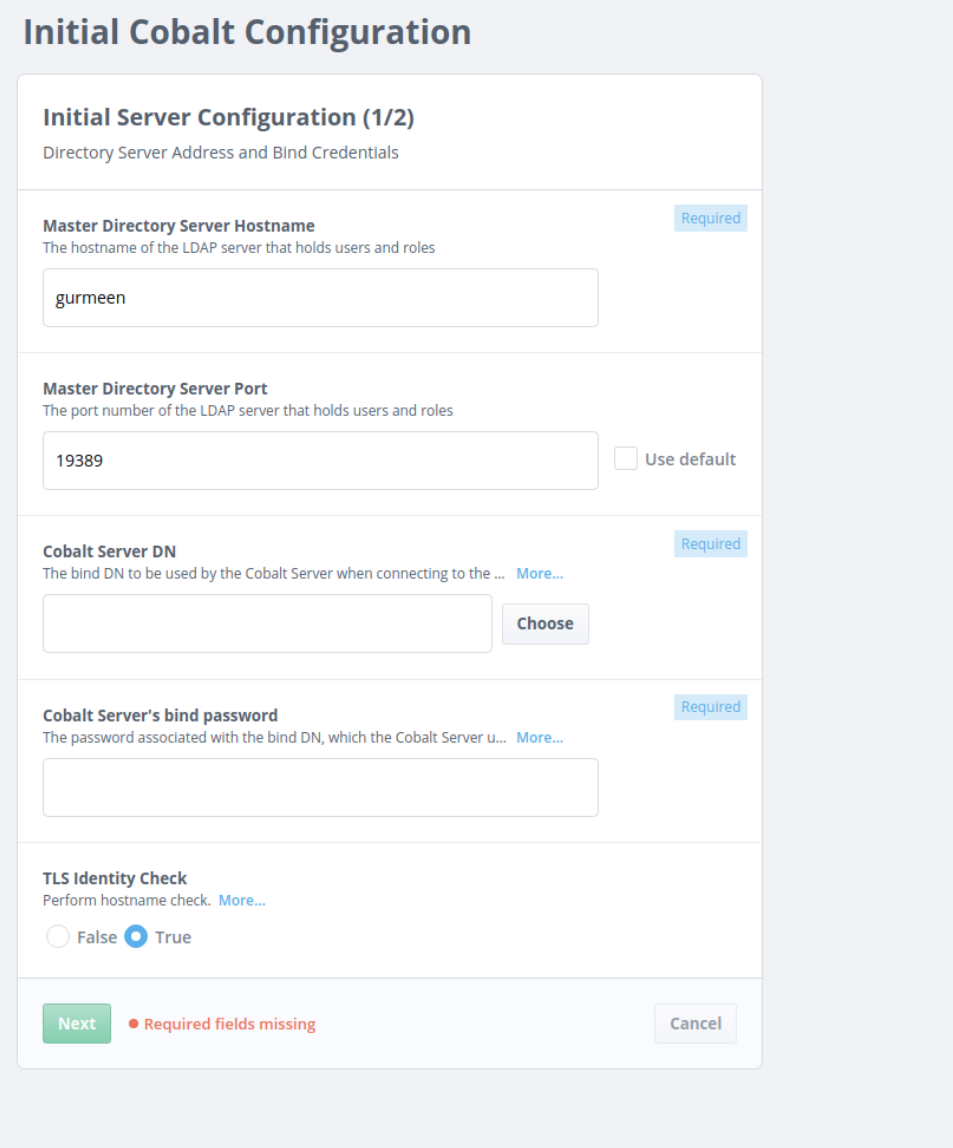
- An entry (which must be a naming context) in the DIT where Cobalt data will be stored. For example, you might have a naming context at o=Cobalt, and a user entry of "cn=Cobalt Server,cn=Users,o=Cobalt" with a suitable password that has read/write/add/delete/modify access to the directory.
- A Cobalt Server user, which has full read/write access to this part of the DIT.

Please contact support@isode.com, if you need help to set this up.

Future releases of M-Vault will install with Cobalt, so that these pre-requisites will be addressed as part of the install.

2.1.3 Directory Configuration

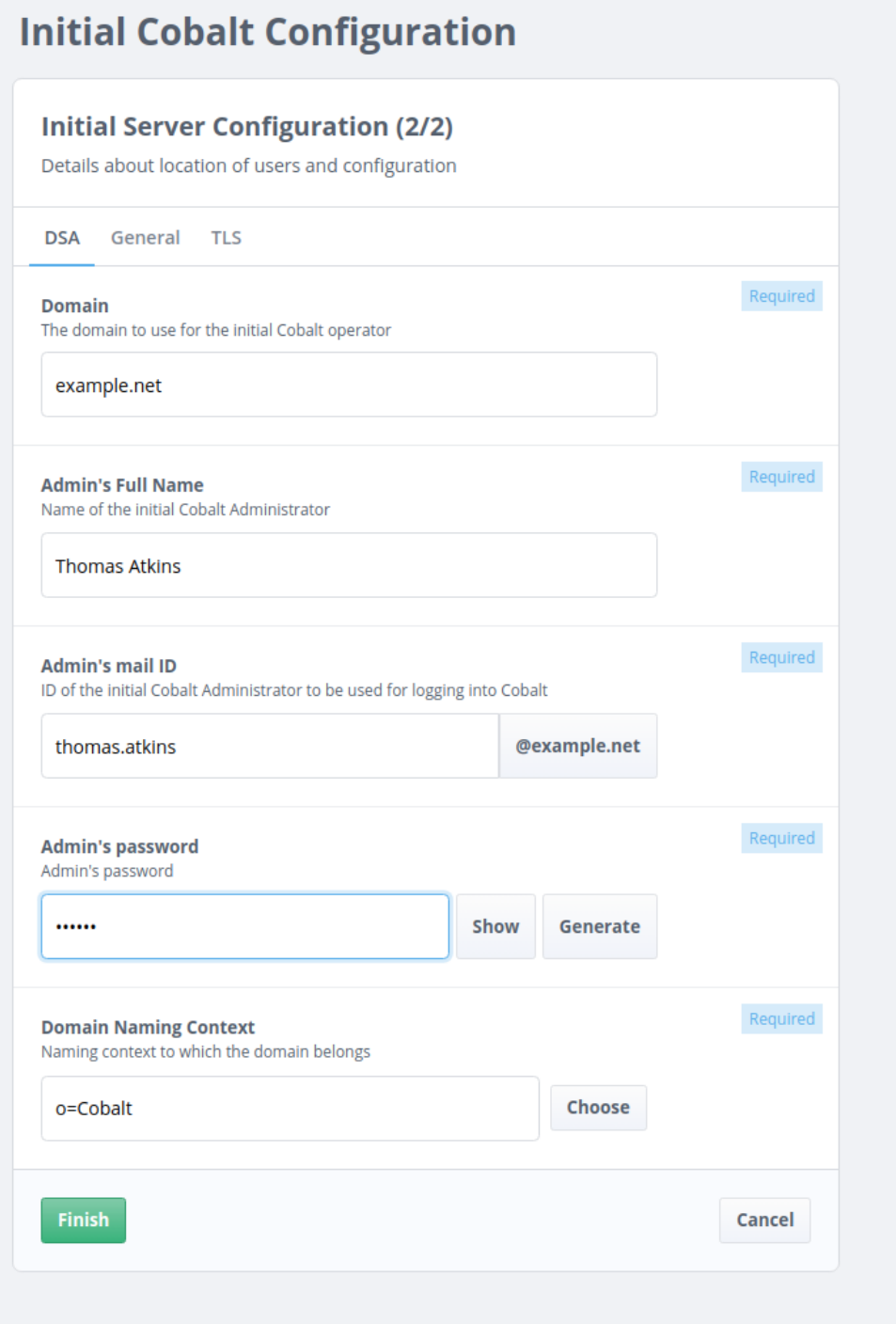
Figure 2.5. Directory Configuration Details



Cobalt will then guide the user to configure access to M-Vault, using the pre-requisite information as shown in the screen above (Figure 2.5, “Directory Configuration Details”).

2.1.4 Default Domain and Initial Users

Figure 2.6. Configure Default Domain and First User



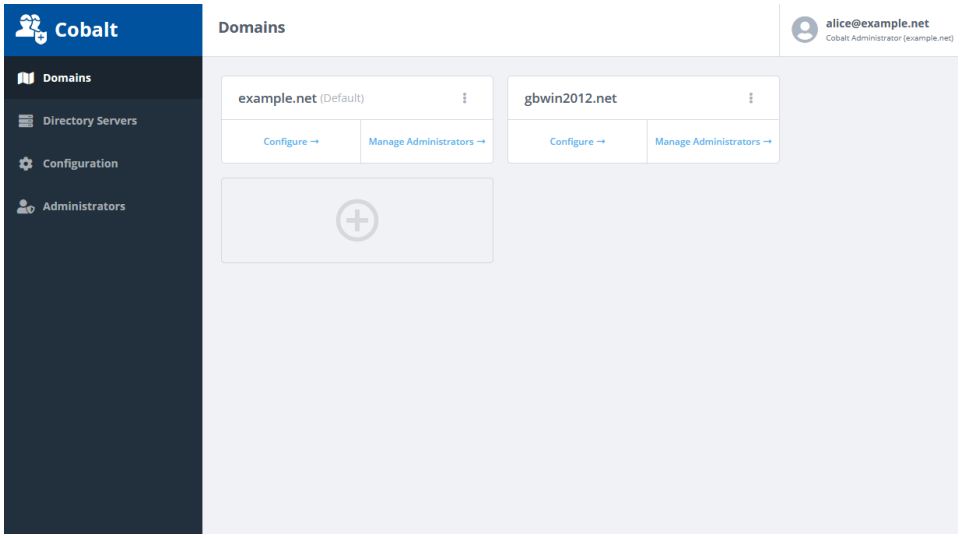
In the second stage of bootstrap (Figure 2.6, “Configure Default Domain and First User”), an initial domain and a single user within that domain are created, with a password for that user. This initial user is configured as a Cobalt Administrator and will have full rights to manage the initial domain.

Cobalt bootstrap is now complete, and the initial user can authenticate to Cobalt, to perform either Cobalt Administration or Domain Administration as described in the following sections.

2.2 Cobalt Administration

Cobalt provides a separate view (Figure 2.7, “Cobalt Administration View”) for its own administration and configuration. This view provides options to manage server configuration parameters (e.g. HTTP port, TLS Identity, etc), domains and Cobalt administrators.

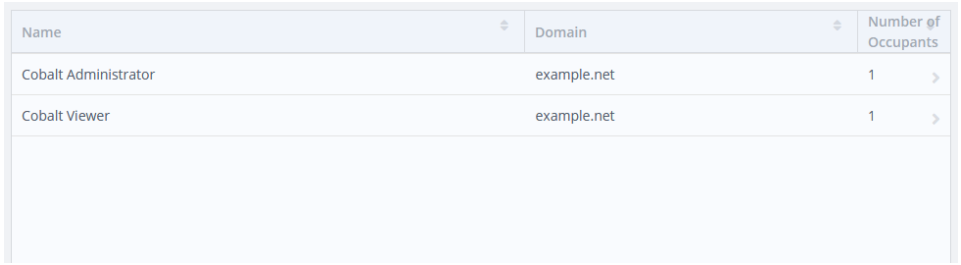
Figure 2.7. Cobalt Administration View



2.2.1 Cobalt Administrative Roles

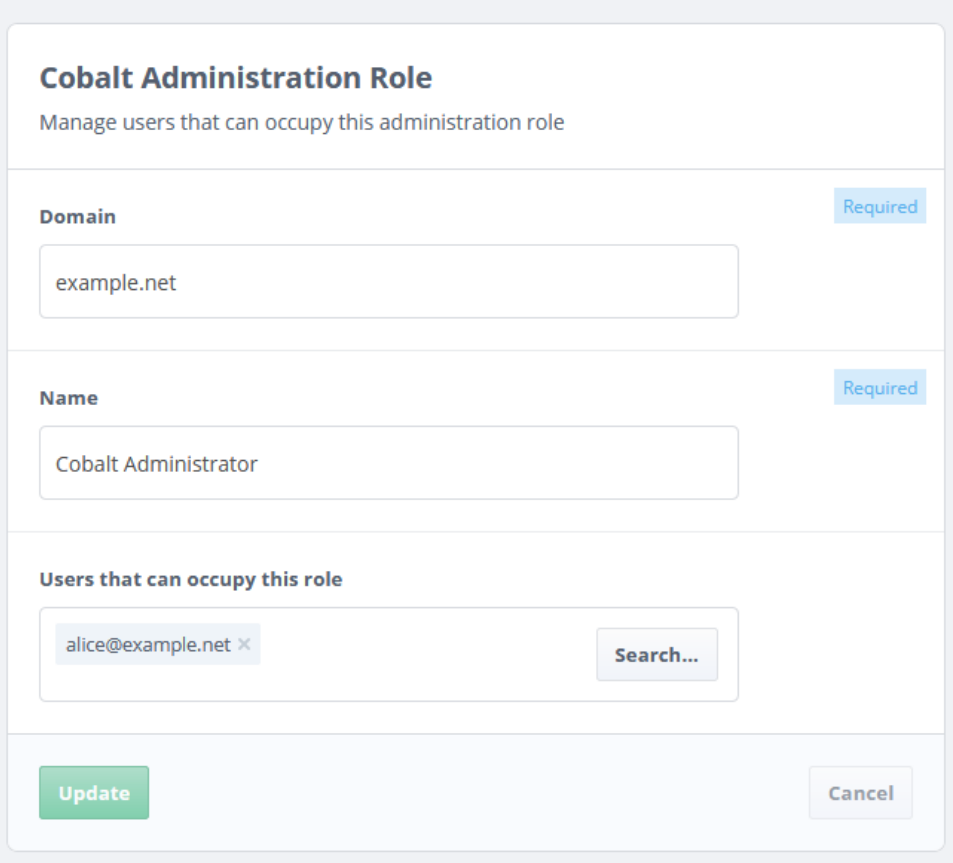
The following figure displays the Cobalt Administrative Roles as described in Section 1.4.2, “Cobalt Administrator Roles”

Figure 2.8. Cobalt Administrators



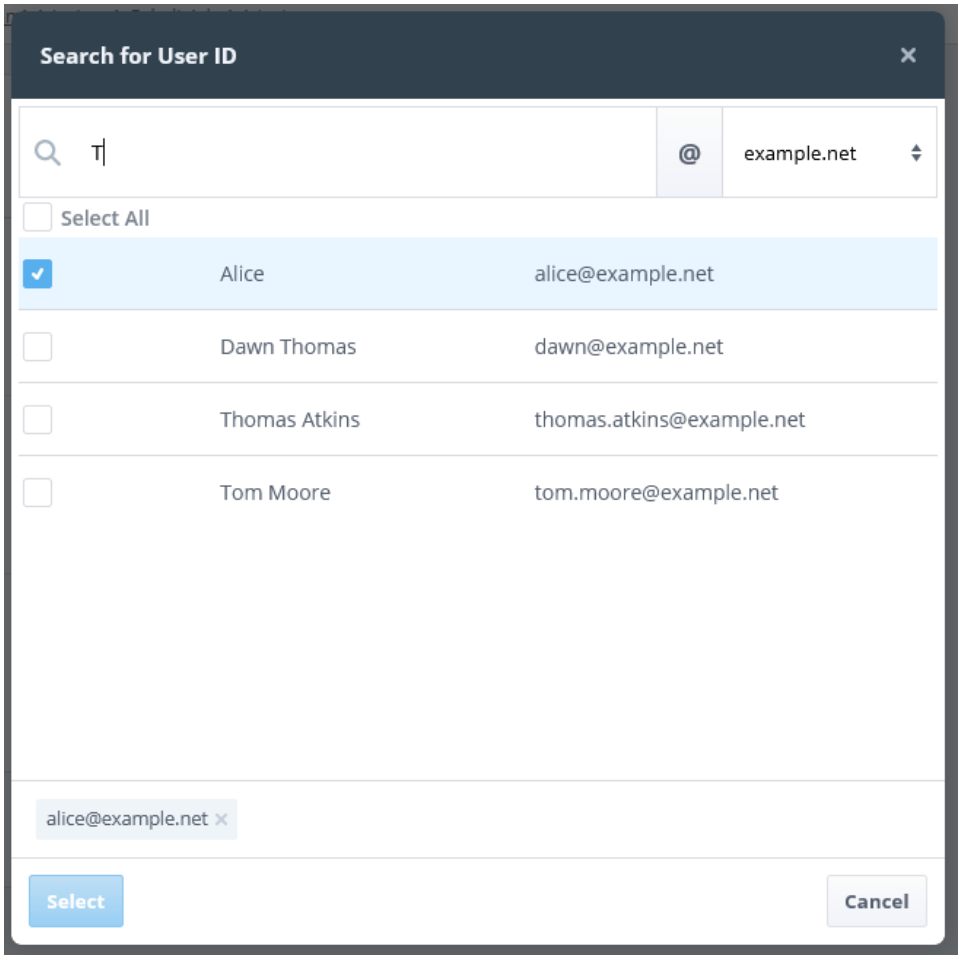
In order to add or remove users from the role, select the role that you wish to edit. The following form will be displayed.

Figure 2.9. Edit Administrators



Click the **Search...** button to display a dialog (Figure 2.10, “Search and Select”) to search and select users from any of the Cobalt configured domain. Select the domain from the dialog and type a few letters to search users in the selected domain. A list of users matching the search string will be presented. Select all users that you wish to add. Repeat this process with another search string and domain for adding more users.

Figure 2.10. Search and Select



Once users have been selected for the role, click the **Select** button to complete the selection and then press the **Update** button to submit changes on the form to update users in the selected role.

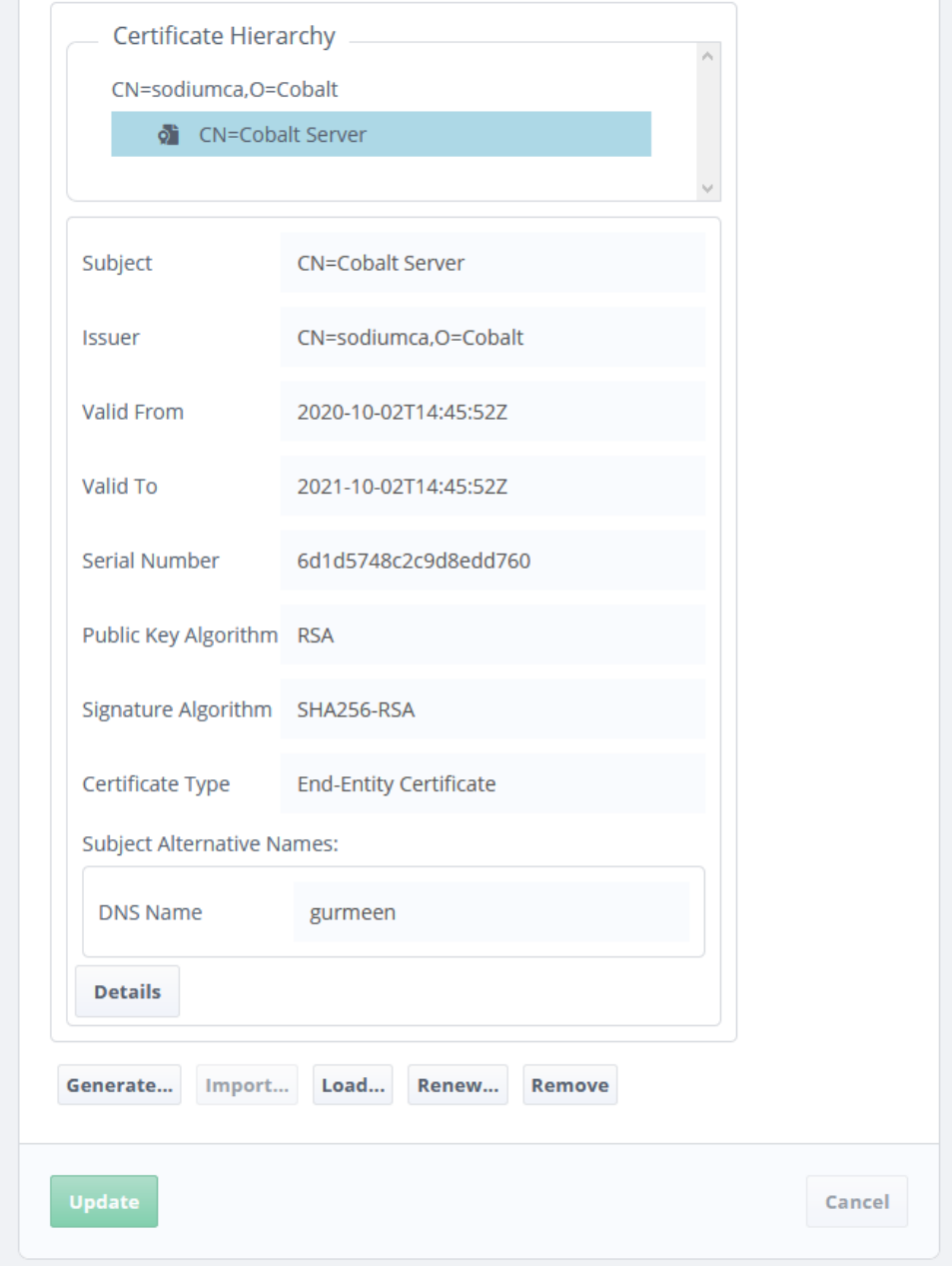
2.2.2 TLS Configuration

The **TLS** tab on the **Configuration** section displays the identity used by Cobalt for its HTTPS configuration (see Figure 2.11, "TLS Configuration Tab"). Use the **Generate...** button to create a new keypair and *CSR* to request a certificate from a *CA*. The *certificate chain* received from the CA can then be imported using the **Import...** button.

Alternatively, use the **Load...** button to load a *PKCS#12* or a concatenated *PEM* form of private key and *certificate chain*.

The **Renew...** button can be used to renew the current server certificate to replace an expired or revoked certificate.

Figure 2.11. TLS Configuration Tab

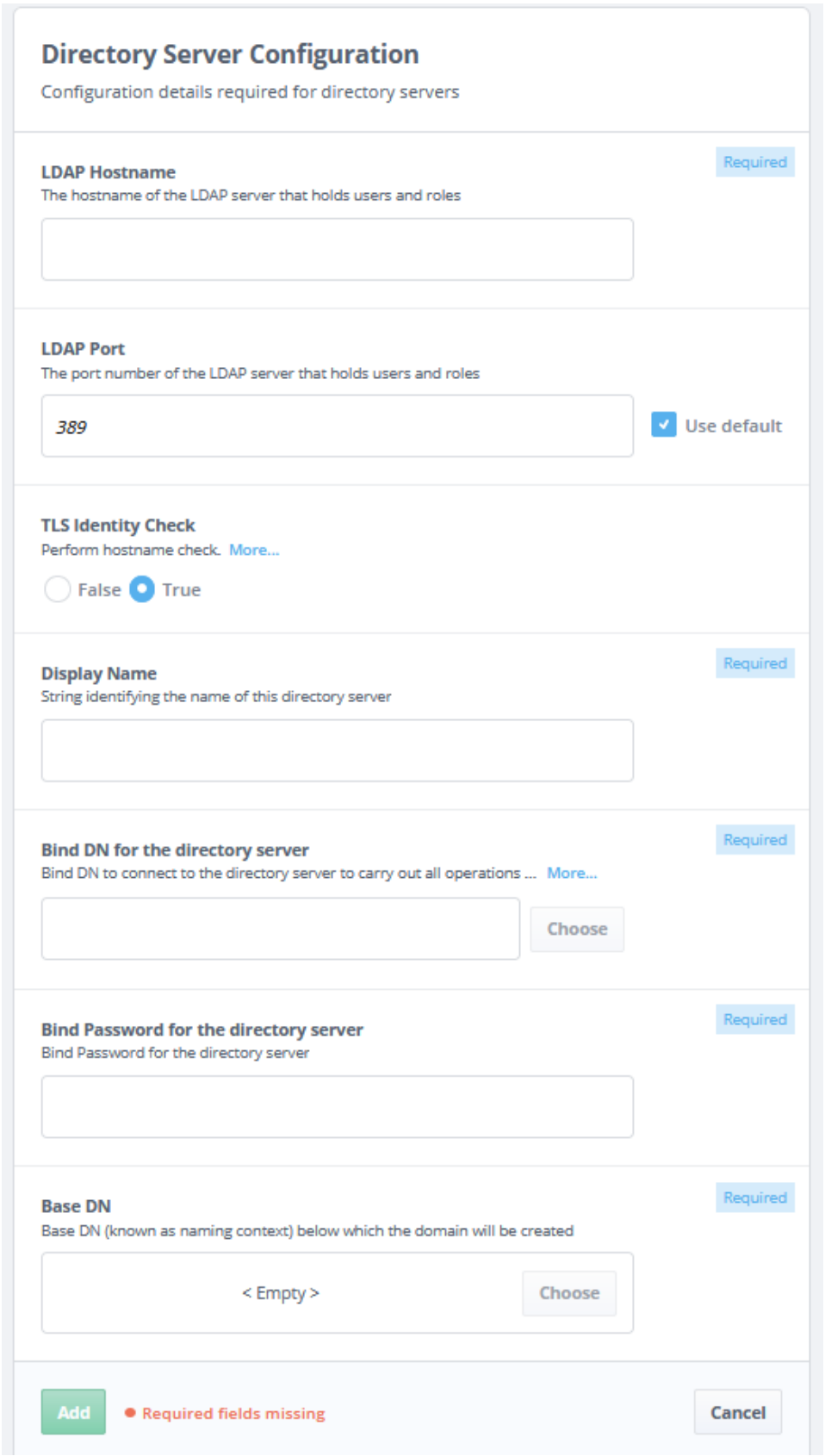


2.2.3 Directory Servers

Cobalt maintains a list of *directory servers* where its domain information is stored. The default directory server is the *M-Vault* server that holds Cobalt's configuration and data for the default domain. A single directory server can hold one or more domains.

In order to add a domain in another directory server, an entry for the directory server should be created here by clicking the **Add** button to see a form as shown in Figure 2.12, “Add Directory Server Form”. Provide host name, port, naming context, bind *DN* and password of a user entry that has suitable access over the directory tree of the DSA.

Figure 2.12. Add Directory Server Form



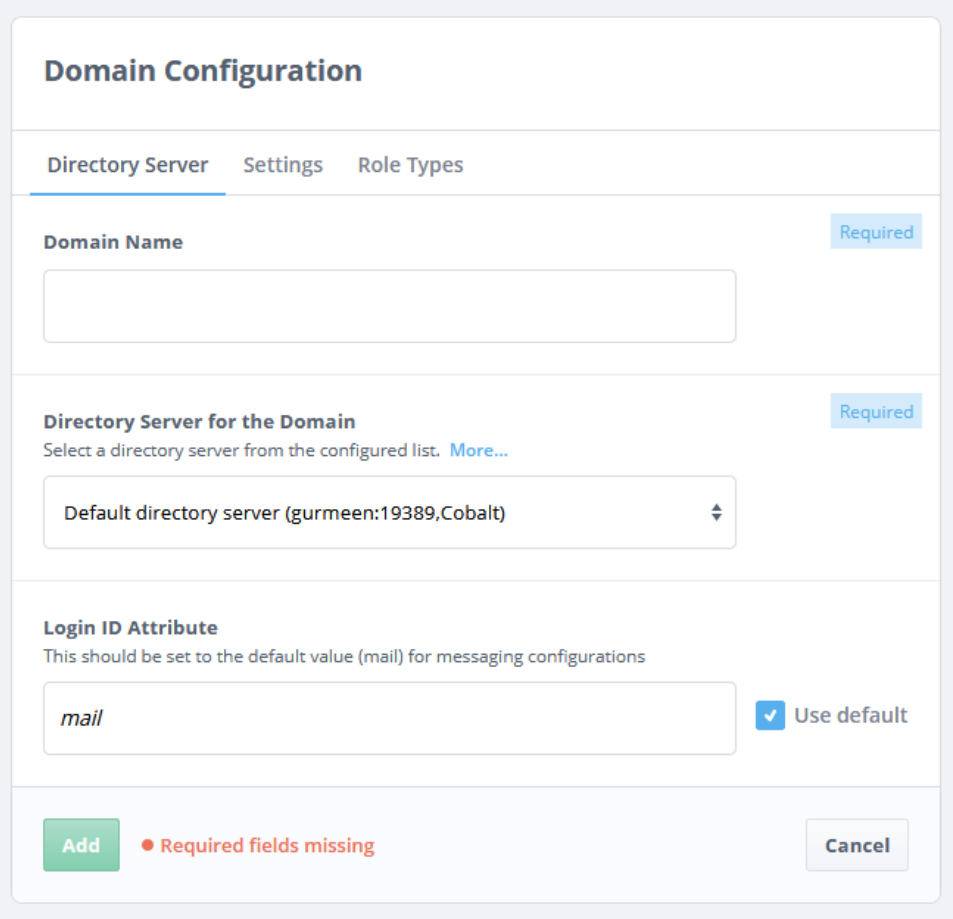
2.3 Setting up Domains

Cobalt manages one or more domains that can be setup and modified in the Cobalt Administrator mode. A default domain is always present and is stored in the Cobalt's own *directory server*.

2.3.1 Adding a Domain

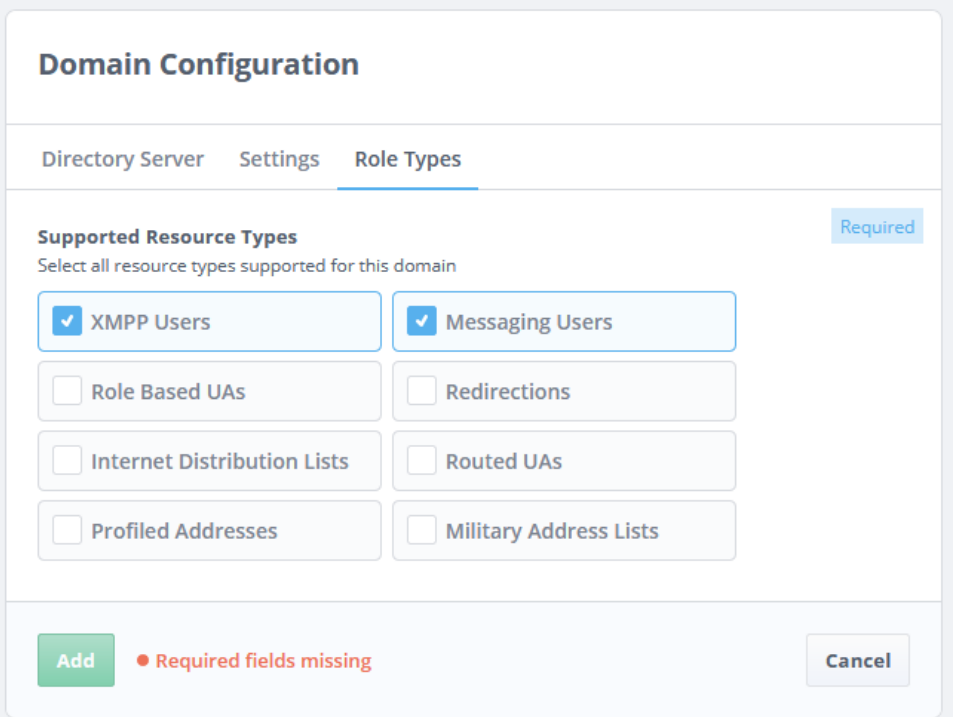
To add a new domain, select the **Domains** item from the sidebar on the left and click the **Add** button. The following form will be displayed. Enter the domain name and the directory server that it belongs to.

Figure 2.13. Add Domain Form



Select the roles supported for the domain on the **Role Types** tab as shown in the figure below. A messaging domain can also specify domain specific settings on the **Settings** tab.

Figure 2.14. Select Domain Role Types

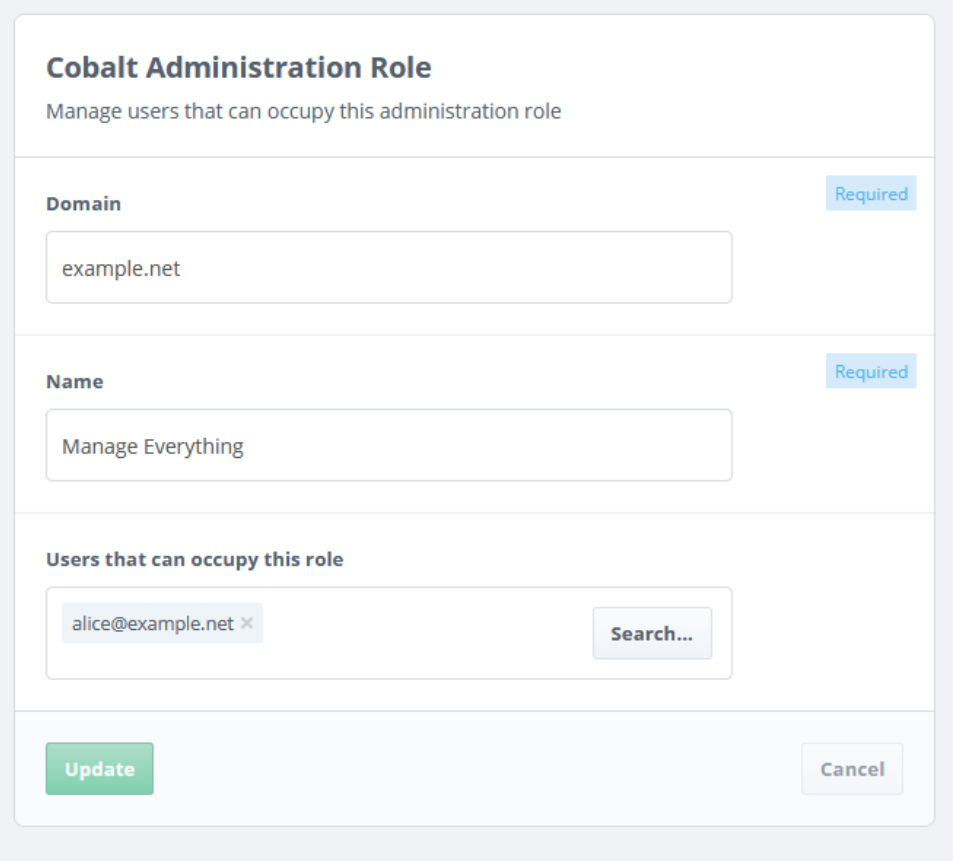


2.3.2 Domain Administrators

Users from any of the Cobalt managed domains can be assigned a role for managing a domain in one or more role types of a domain as described in Section 1.4.3, “Domain Administrator Roles”.

Click the role that you wish to assign a user to. The following form will be presented. Click the **Search** button to search and add users to this role as described in the figure Figure 2.10, “Search and Select”. Click the **Update** button for the change to be submitted to take effect.

Figure 2.15. Edit Domain Administrators



Chapter 3 Cobalt for Domain Administrators

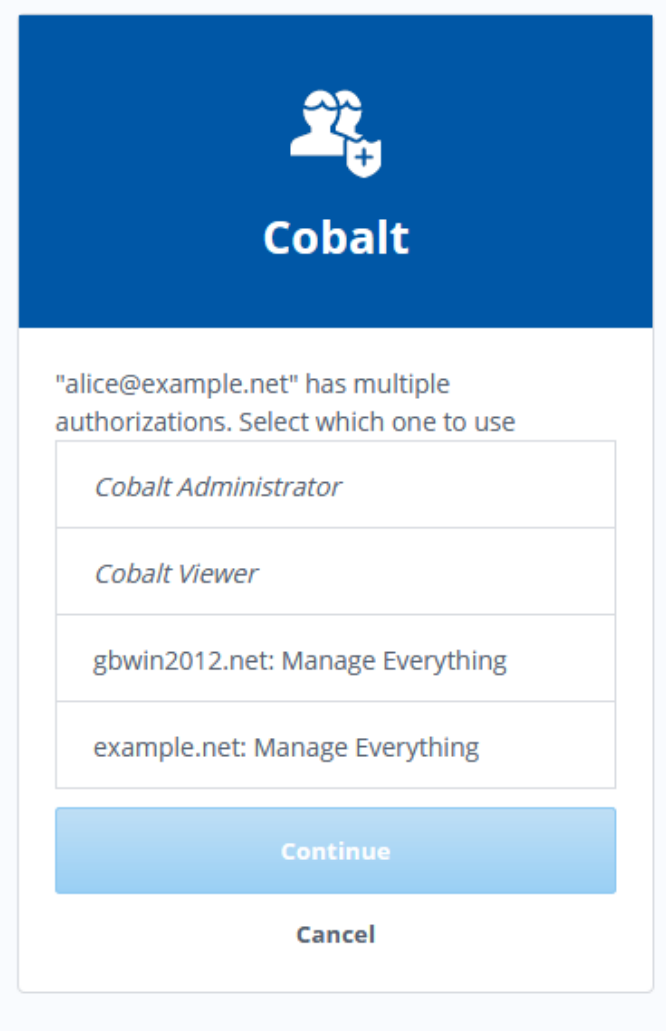
3.1 Accessing Cobalt

After providing a valid username and password, Cobalt will show the user which roles they are authorized to use (unless the user is only assigned to one role). A user can only be active in one role at any given instance. A user with multiple authorizations can switch role by clicking the top right user ID icon and selecting **Switch View** button.

The following figure illustrates the page presented to a user with multiple Cobalt roles.

3.1.1 Selecting and Switching View

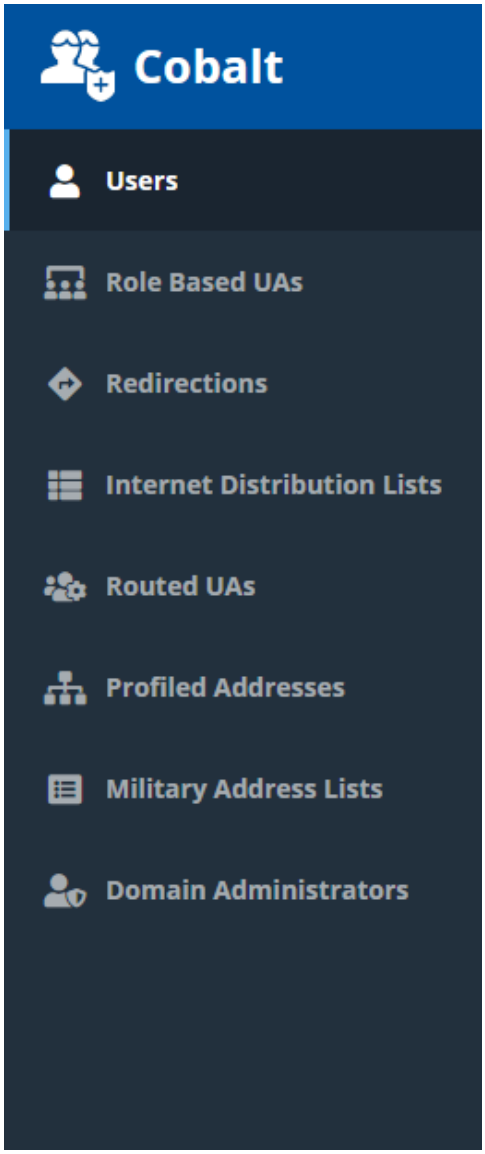
Figure 3.1. Select View



Once a role to manage a domain is selected, Cobalt will present a filtered view based on selected authorization. A *Users Manager* for a domain will be presented with a view that displays a list of users whereas a *Roles Manager* will be presented with a list of configured roles for that domain. A user in *Manage Everything* role will be able to see users, configured roles and domain managers for the domain.

The Left Hand Side (LHS) sidebar as shown below will display all resources as items that can be managed for a domain after role is selected.

Figure 3.2. LHS Sidebar



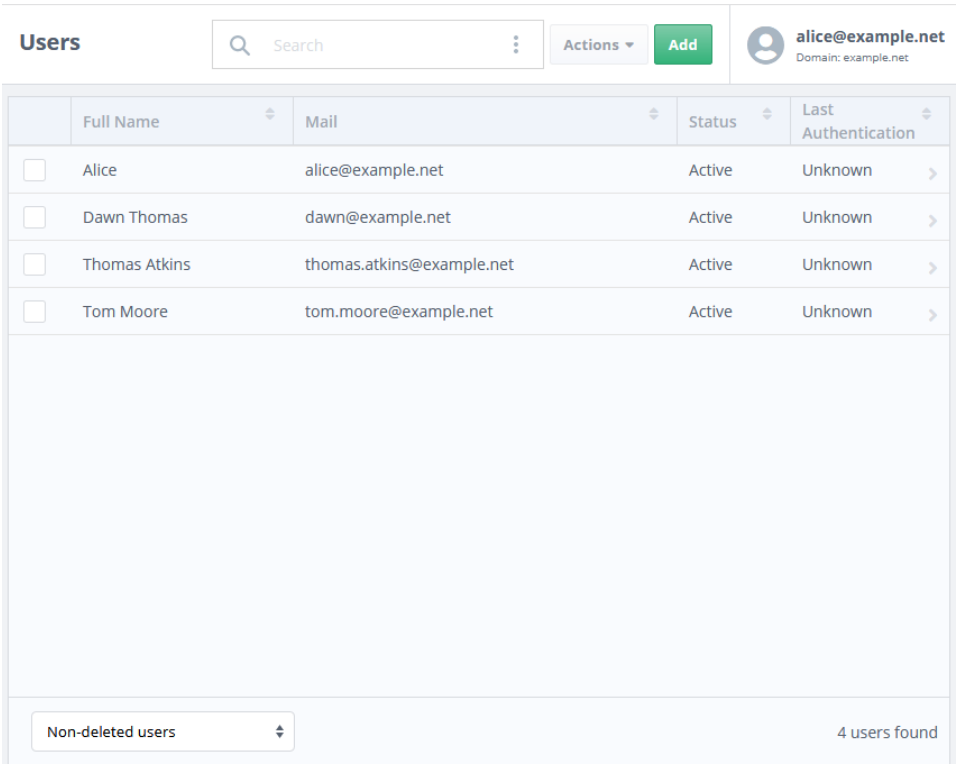
3.2 Managing Users

Cobalt presents a list of users (as shown in Figure 3.3, “Users View”) configured in a domain that supports users. The search box can be used to make the list only show users matching a specific string.

The **Actions** dropdown provides a range of operations that can be performed on selected users.

The filter box at the bottom of the page is used to only show users matching a specific status (all users, deleted users, etc).

Figure 3.3. Users View

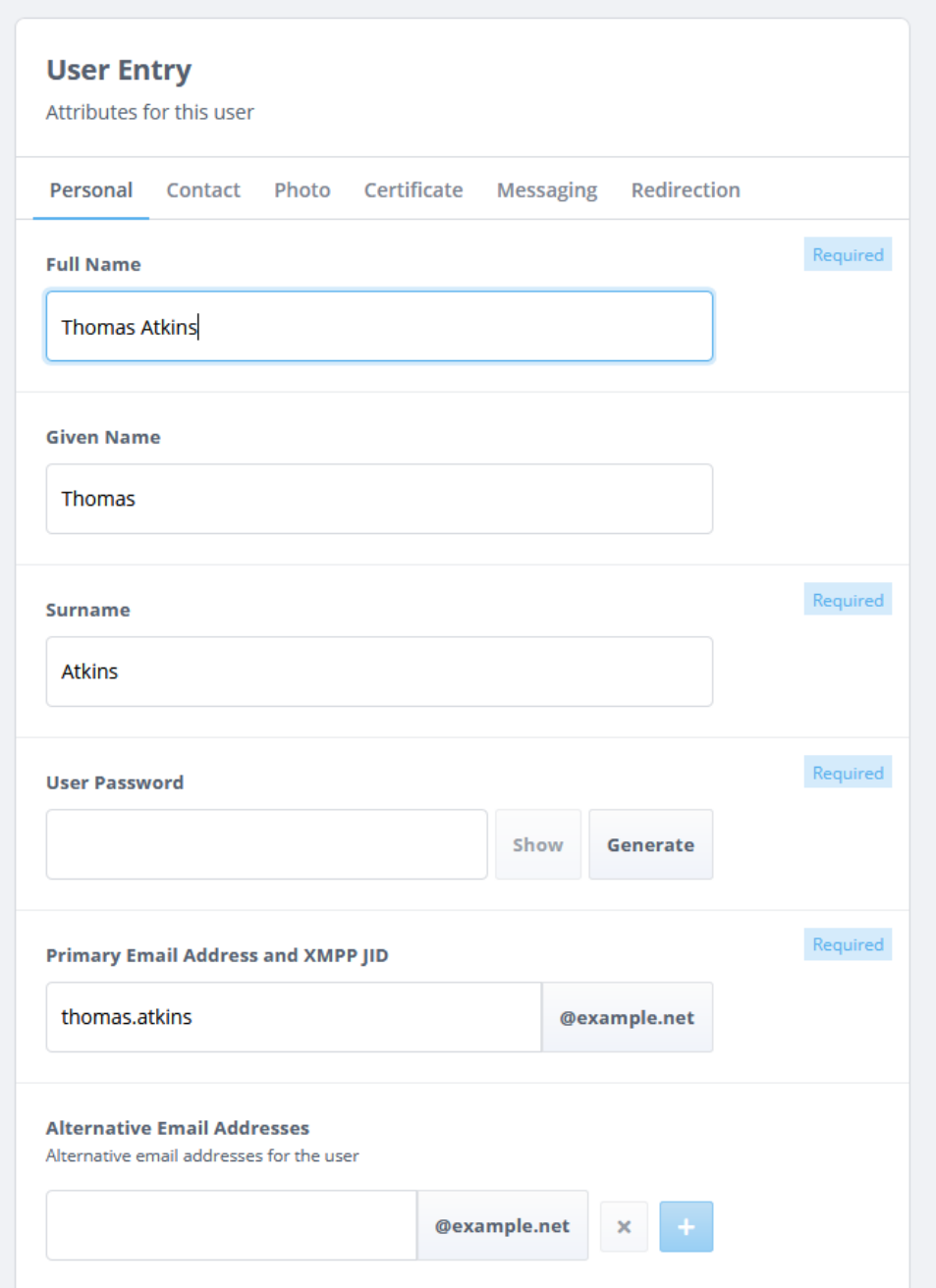


In order to delete a user, check the box next to the username, and then select **Delete** action from the **Actions** dropdown. Alternatively, use the right click context menu to select an action. When Cobalt deletes a user, the user's entry is moved to the *Deleted* section of the DIT (see Section 1.3.4, “DIT Layout”). A *Deleted* user can be restored by selecting the **Restore** action from the **Actions** dropdown menu. A user entry can be removed from the directory altogether by using the **Purge** action. Note that a purged user cannot be restored, and Cobalt will not prevent any new user with the same name as the purged user from being added.

A user can be a locked either as a result of password policy or manually by the administrator by selecting the **Lock** action from the **Actions** dropdown menu. Use the **Unlock** action to unlock the user.

To view details for a user, click on the appropriate row. The details will be displayed, with attributes grouped into tabs. A new user can be added using the **Add** button. The following form will be displayed for adding a user in a domain.

Figure 3.4. Add User



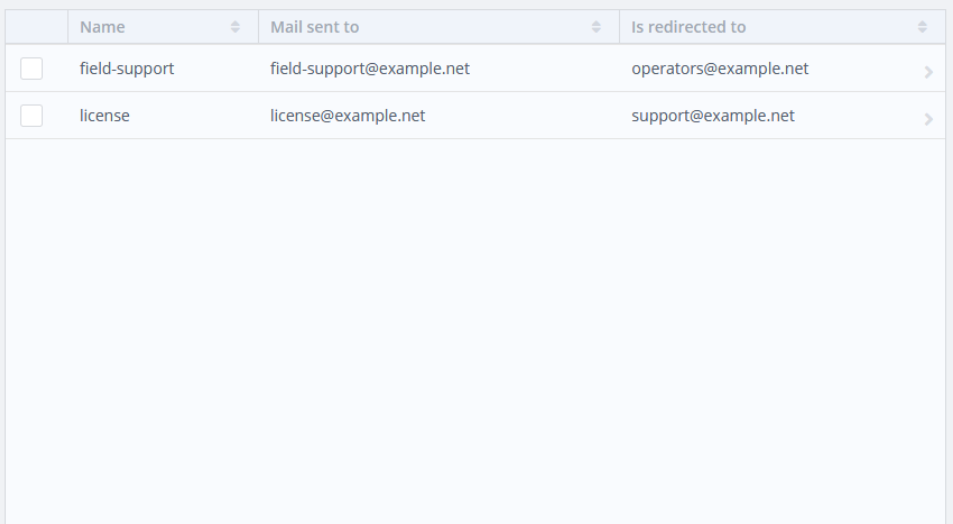
A user's password can be reset using the **Reset Password...** button on the user form.

3.3 Managing Redirections

A **Redirection** role provides the functionality to specify a redirection for one or more email addresses to a given email address. See Section 1.2.4, “Email Support”.

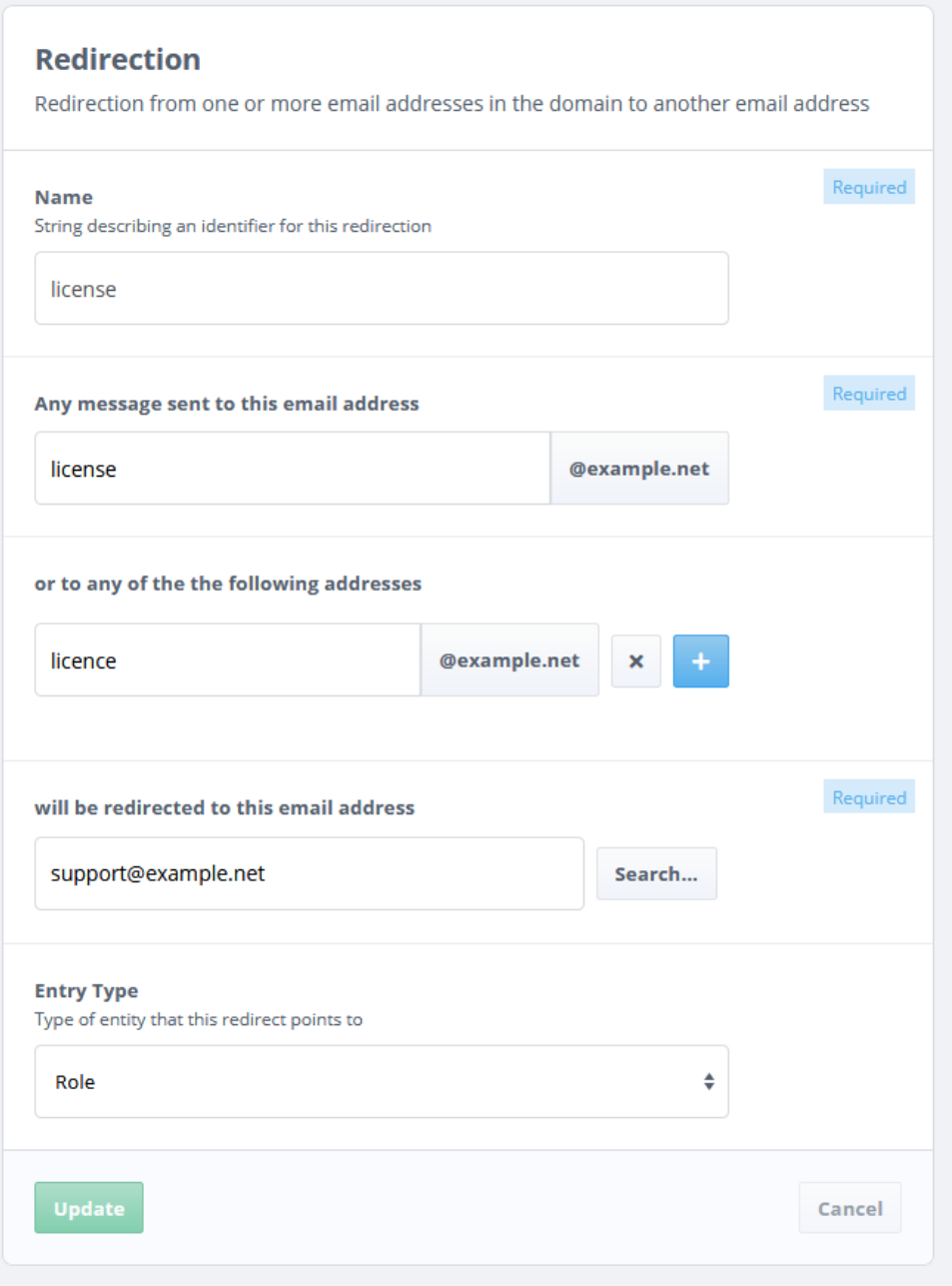
Select the **Redirections** item from the left sidebar (Figure 3.2, “LHS Sidebar”) to display the configured redirections for the domain as shown in figure below.

Figure 3.5. Redirections



Use **Add** to add a new entry and click in a row to view it. The following form will be displayed.

Figure 3.6. Redirection Form



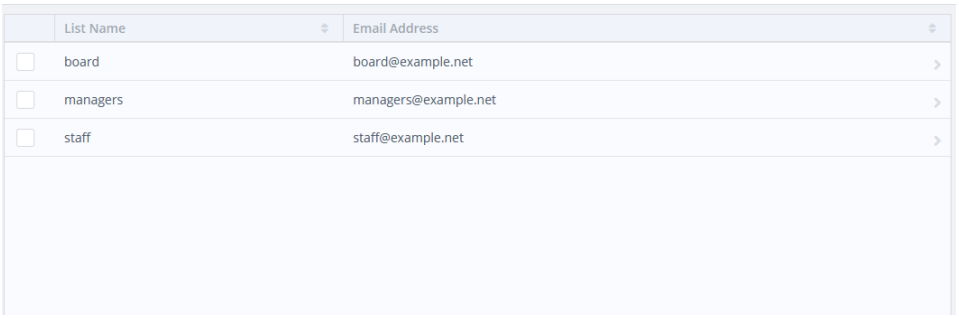
The **Entry Type** attribute describes the type of entity that this redirection points to and can take one of the following values:

- User
- Role
- Organization

3.4 Managing Internet Distribution Lists

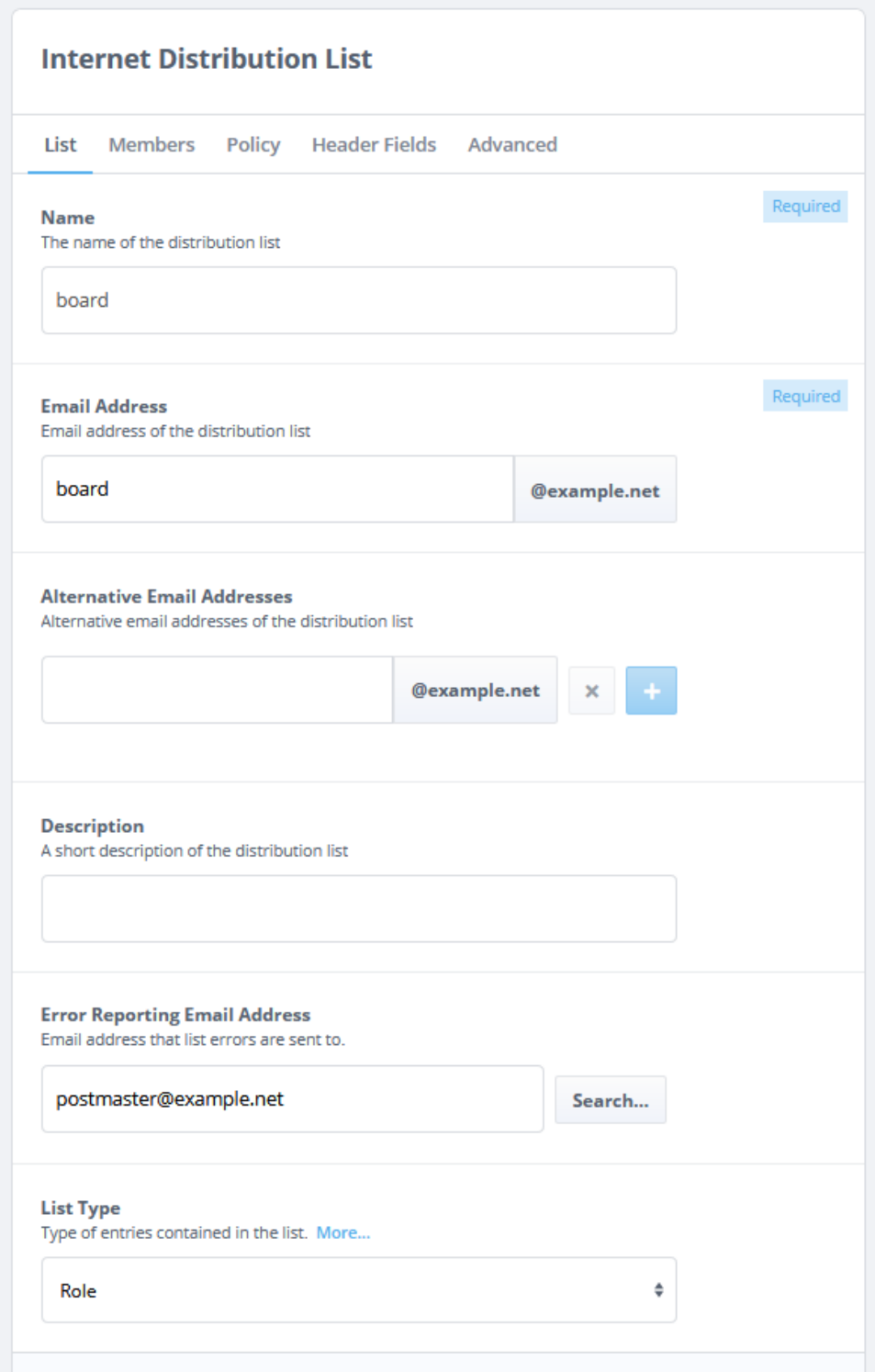
Cobalt provides the functionality to manage Internet Distribution Lists for a domain (see Section 1.2.4, “Email Support”). Select **Internet Distribution Lists** item from the left sidebar (Figure 3.2, “LHS Sidebar”) to view them as shown in the figure below.

Figure 3.7. Internet Distribution Lists



Click in a row in the table to view a list and its attributes (see Figure 3.8, “Internet Distribution List Form”). **Add** button is used to set up a new distribution list.

Figure 3.8. Internet Distribution List Form



A list requires an email address and can have one or more member email addresses (see Figure 3.9, “Internet Distribution List Members”). There are a number of attributes for a distribution list that are grouped in tabs on the form. The email addresses of the members, submitters and error-reporting can be searched and selected using the **Search...** button (see Figure 2.10, “Search and Select”).

The information header addition as per RFC 2369 can be set on the **Header Fields** tab. A *policy* can be specified to control how the list behaves, and who is allowed to submit messages to it (see Figure 3.10, “Distribution List Policy”).

Figure 3.9. Internet Distribution List Members

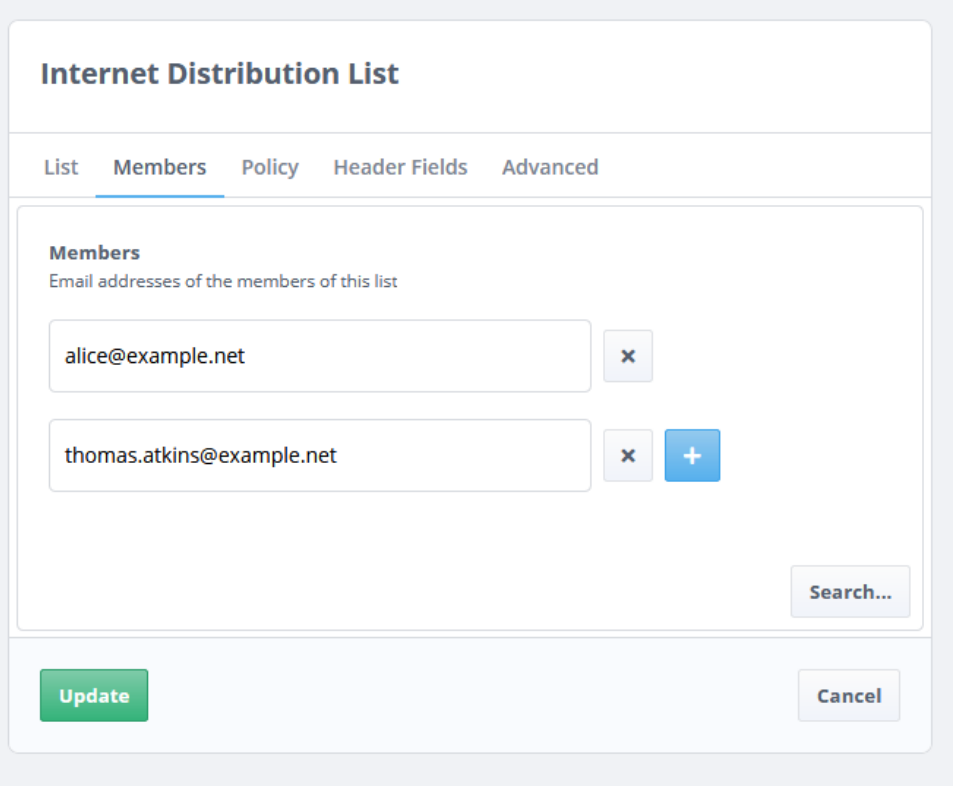
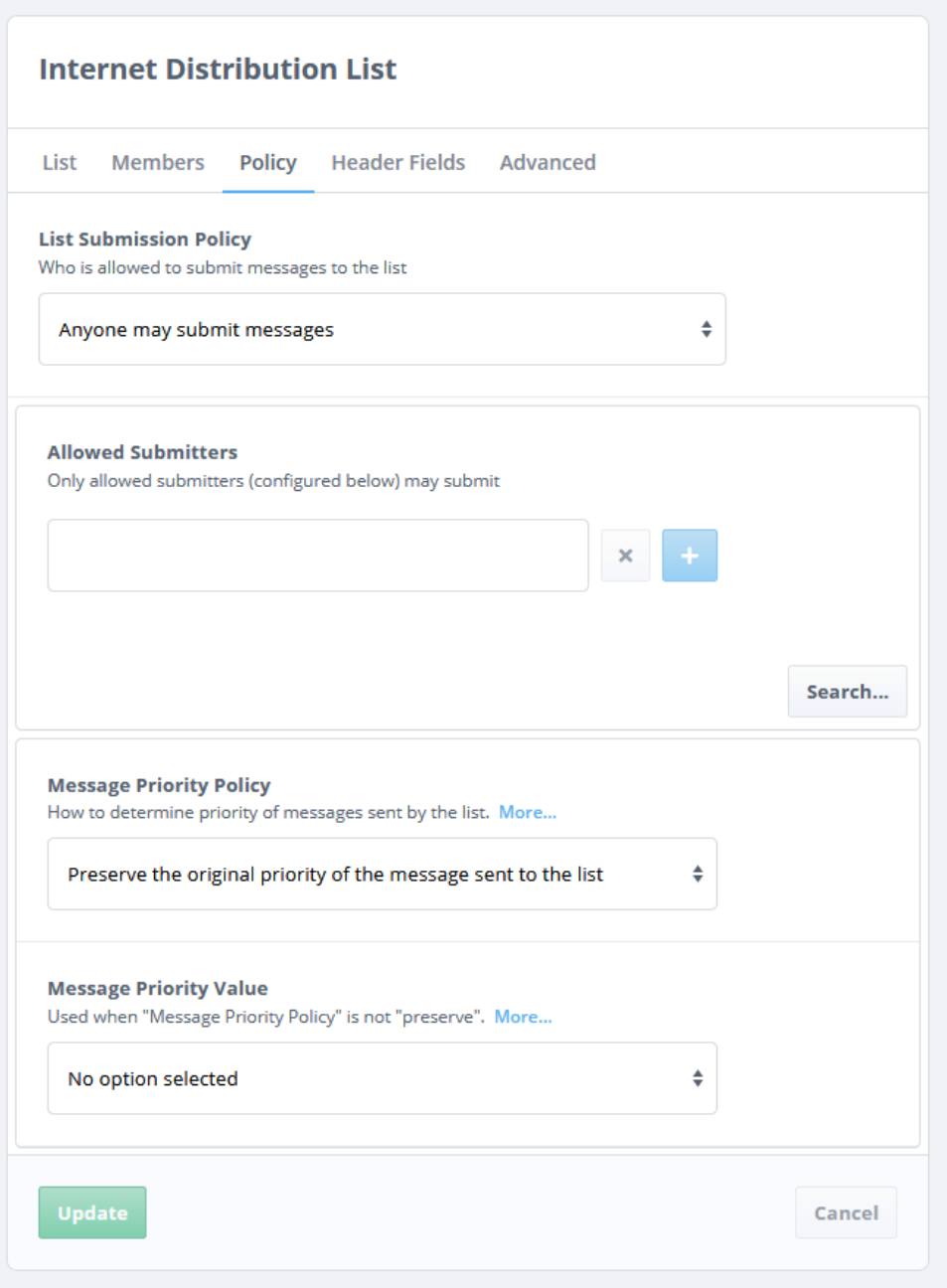


Figure 3.10. Distribution List Policy

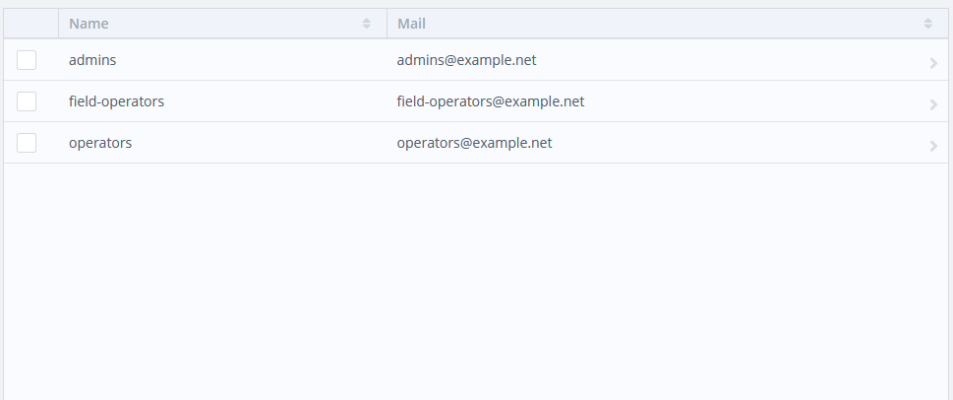


3.5 Managing Role Based User Agents

A Role Based User Agent (see Section 1.2.5, “Military Messaging Support”) can be created and managed by selecting the **Role Based User Agents** item from the left sidebar (Figure 2.10, “Search and Select”).

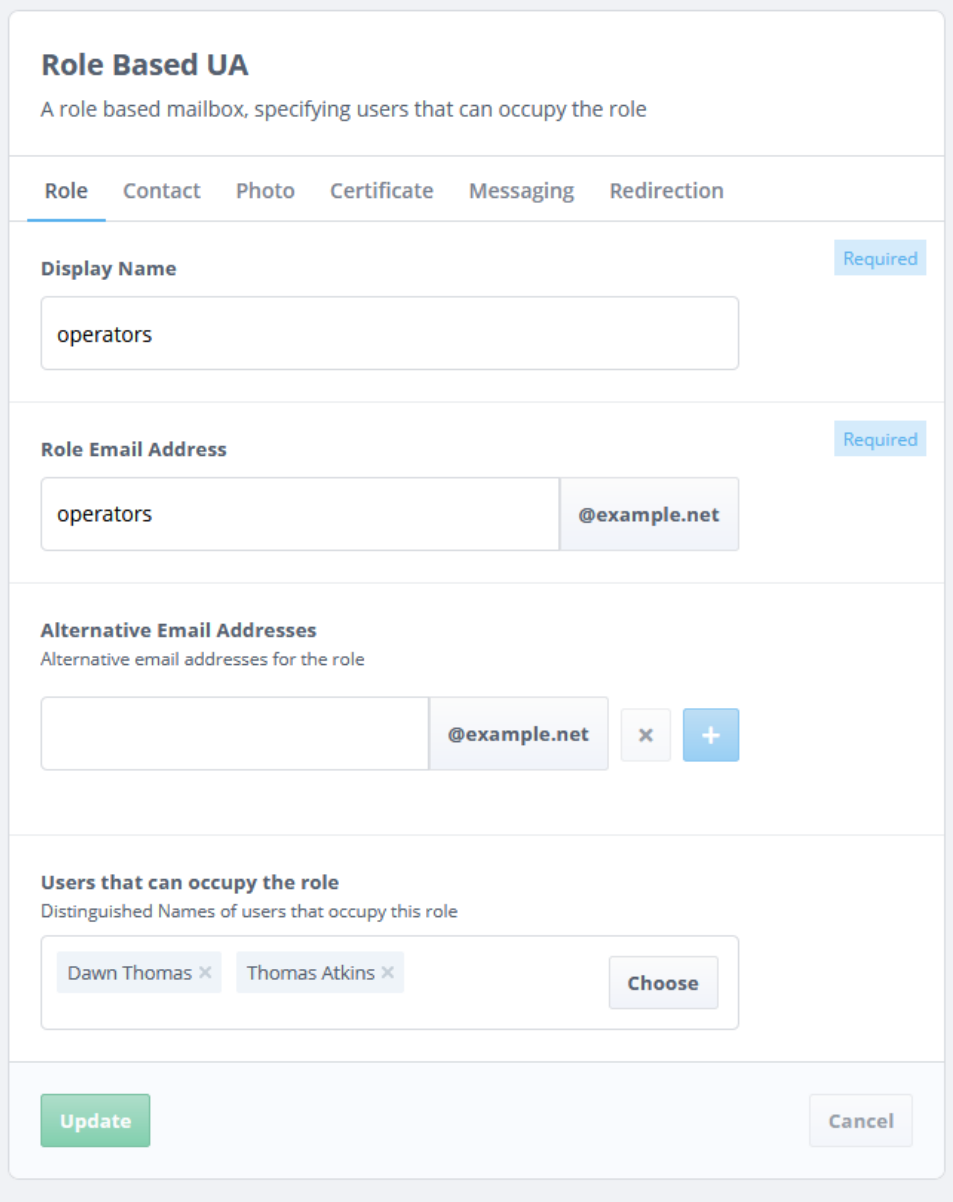
A table as shown in figure below will be displayed.

Figure 3.11. Role Based User Agents



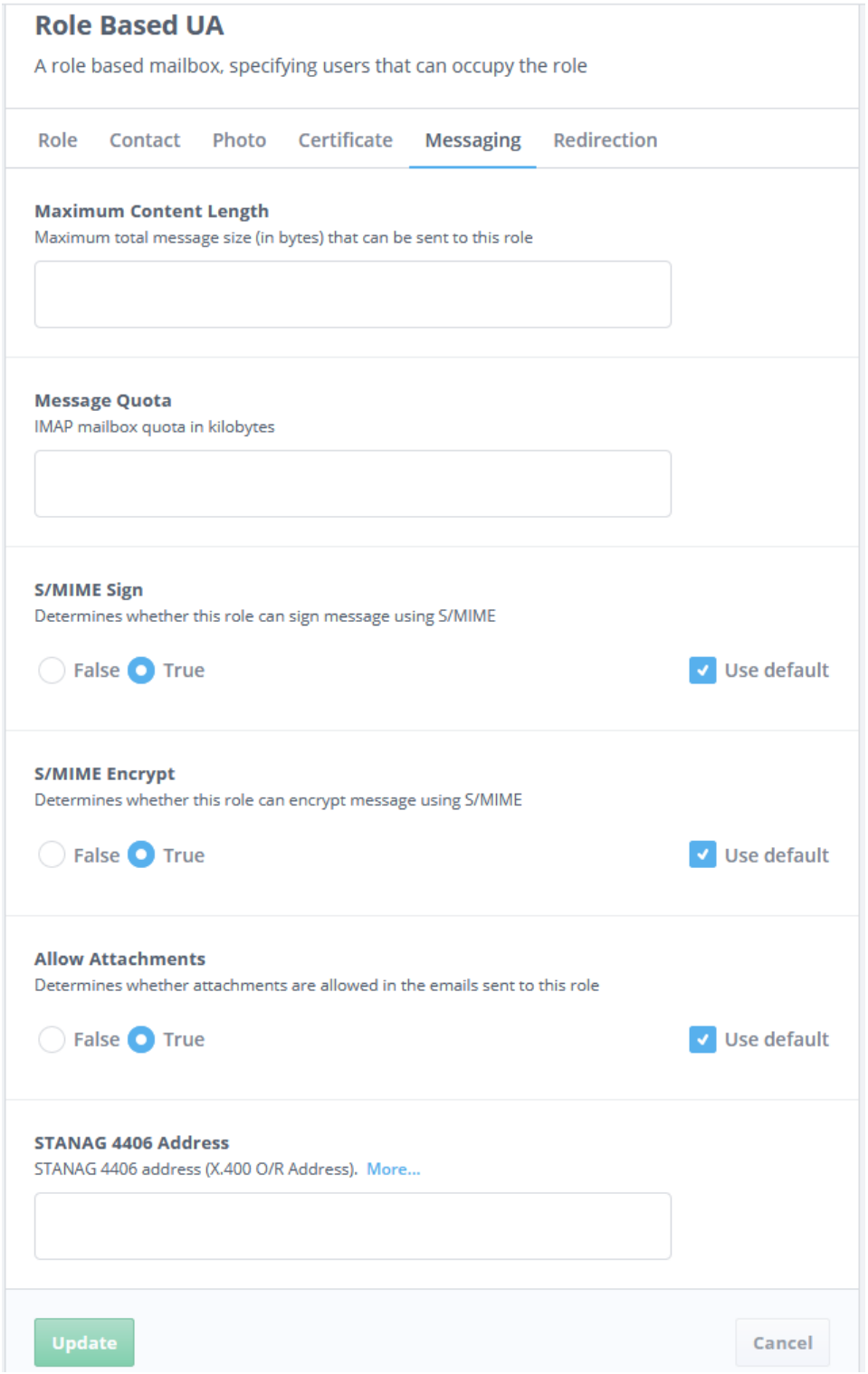
In order to add a Role Based User Agent, select **Add** and fill in the form as shown below to specify the email address for the mailbox. Search and select (Figure 2.10, “Search and Select”) one or more users that can occupy this role.

Figure 3.12. Role Based User Agent Form



Contact tab contains the contact information like address and phone number. **Messaging** tab (see Figure 3.13, “Role Based User Agent Messaging Tab”) specifies the information related to the role's mailbox. Other attributes (photo, certificate, redirection) associated with this role can be found on the respective tabs.

Figure 3.13. Role Based User Agent Messaging Tab

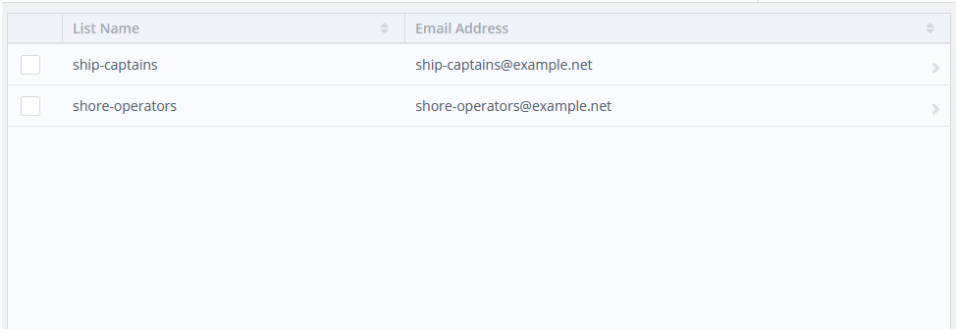


3.6 Managing Military Address Lists

Military Address Lists (see Section 1.2.5, “Military Messaging Support”) can be created and managed by selecting the **Military Address Lists** item from the left sidebar (Figure 3.2, “LHS Sidebar”).

A list of **Military Address Lists** will be displayed as shown below.

Figure 3.14. Military Address Lists

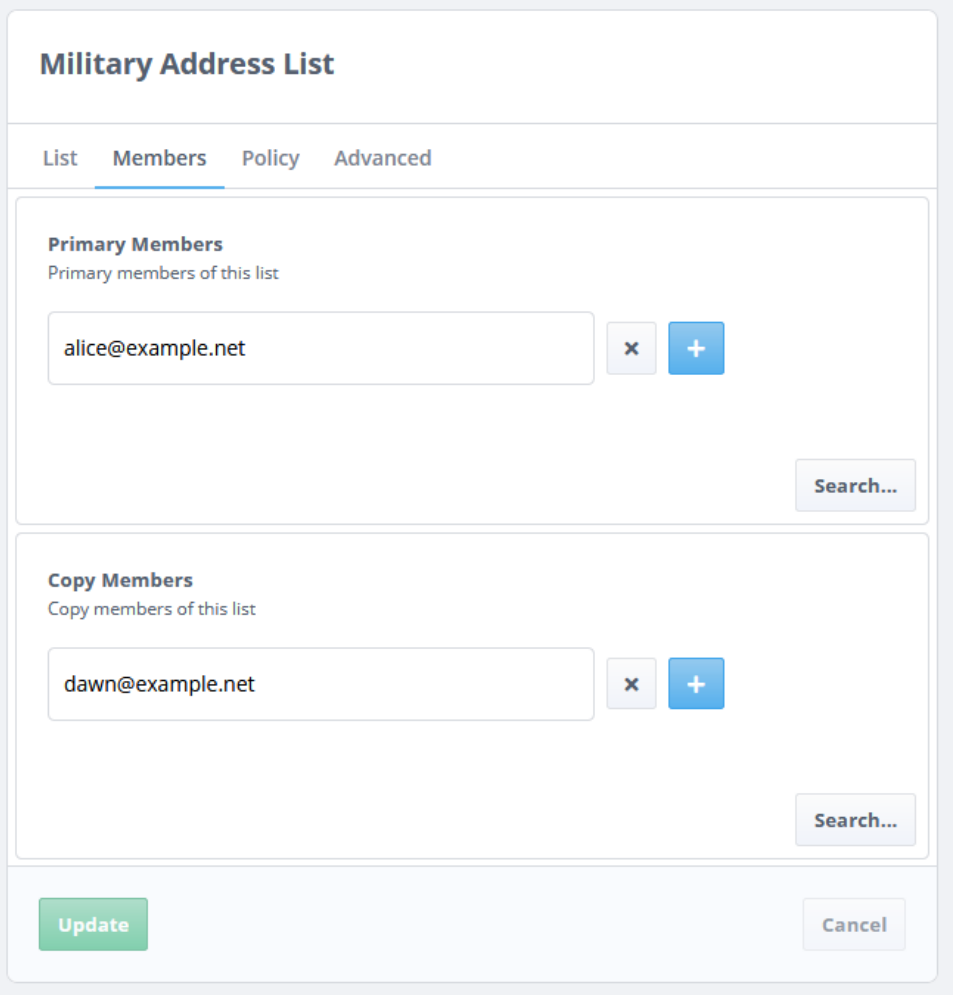


Military Address Lists can be set up like Internet Distribution Lists by specifying an email address and *Primary* and *Copy* members for *Action* and *Info* recipients respectively.

A *policy* can be specified to control how the list behaves, and who is allowed to submit messages to it (see Figure 3.10, “Distribution List Policy”).

The figure below displays the form that appears on selecting a **Military Address List** entry.

Figure 3.15. Military Address List

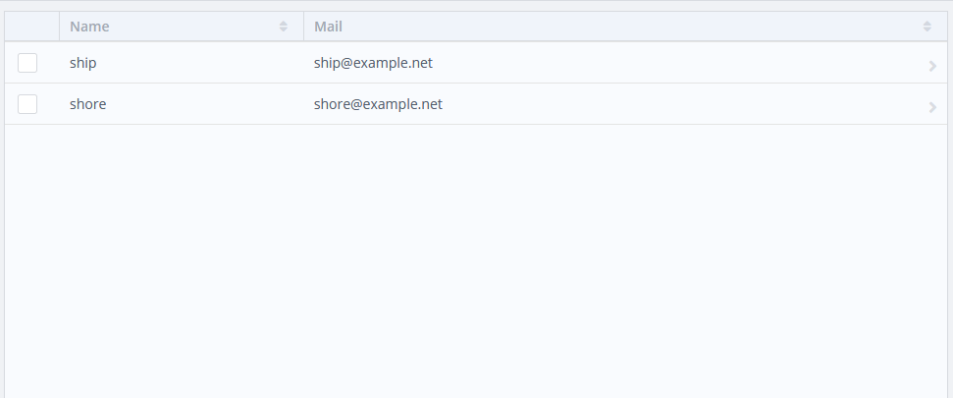


3.7 Managing Profiled Addresses (Organizations)

Profiled Addresses (see Section 1.2.5, “Military Messaging Support”) can be managed by selecting the **Profiled Addresses** item from the left sidebar (Figure 3.2, “LHS Sidebar”).

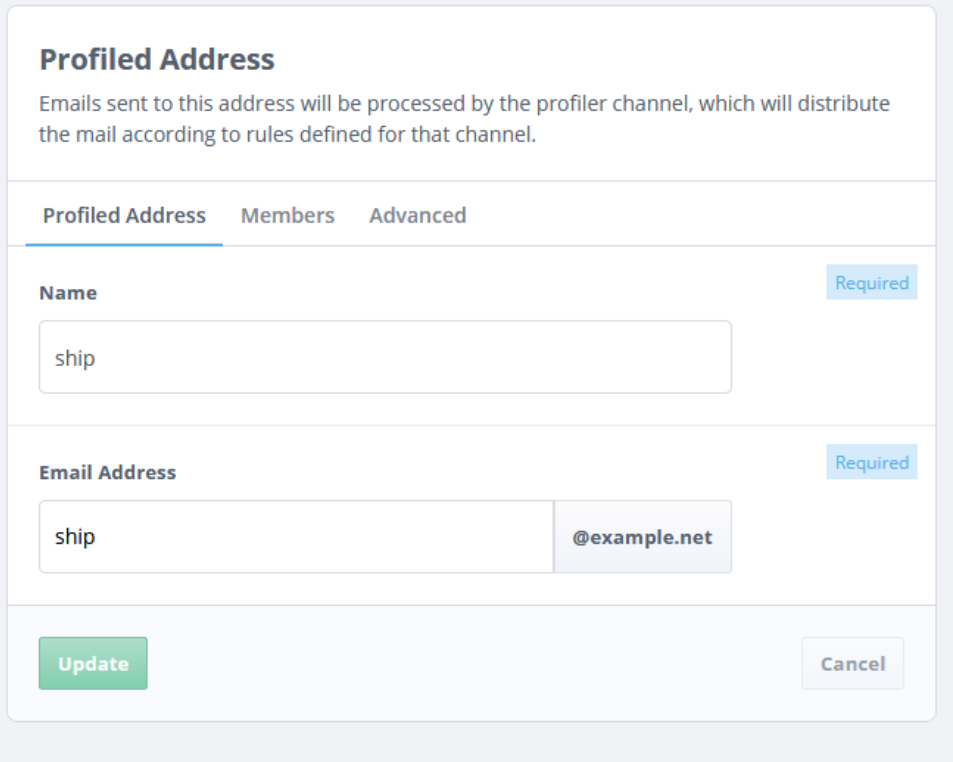
The figure below displays a list of **Profiled Addresses** for a domain.

Figure 3.16. Profiled Addresses



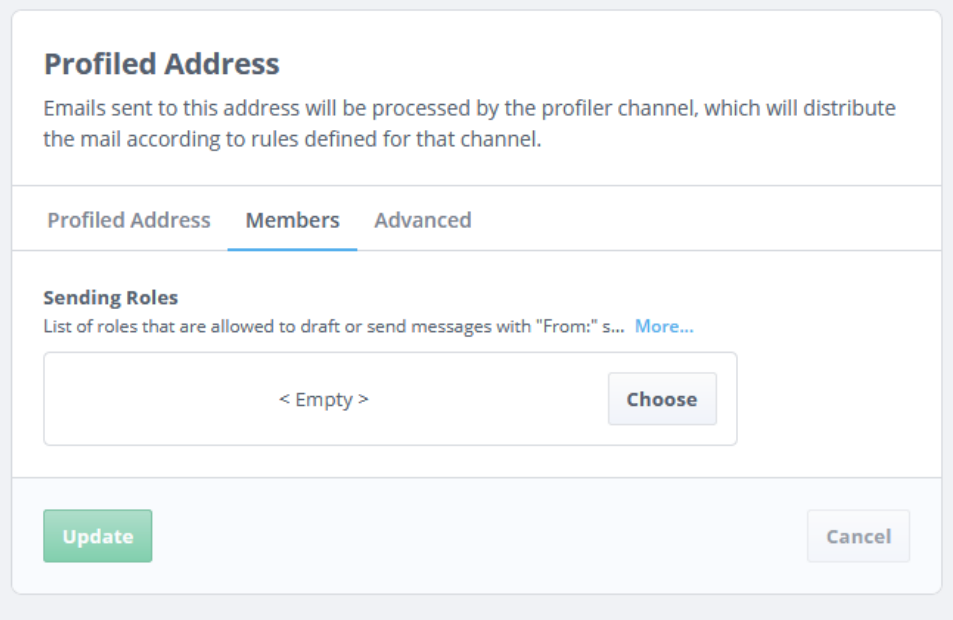
A profiled address requires an email address for creation. The figure below displays the form for a **Profiled Address**.

Figure 3.17. Profiled Address Form



One or more roles can be selected as *members* by using the select dialog that appears by clicking the **Choose** button (see figure below).

Figure 3.18. Profiled Address Members

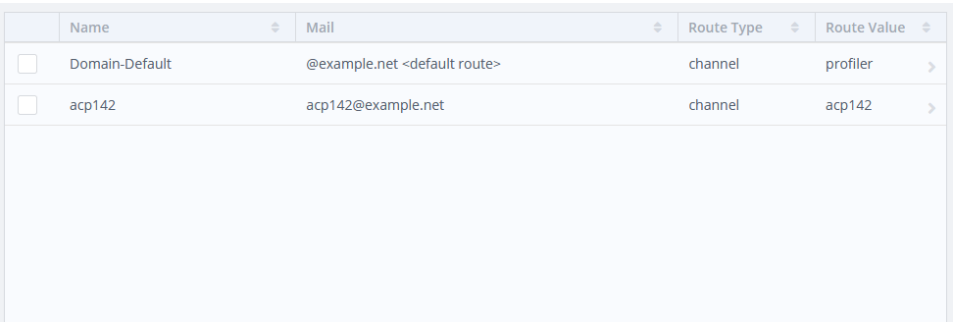


3.8 Managing Routed User Agents

A Routed User Agent (see Section 1.2.5, “Military Messaging Support”) can be created and managed by selecting the **Routed User Agents** item from the left sidebar (Figure 3.2, “LHS Sidebar”).

The figure below displays a list of **Routed User Agents** for a domain.

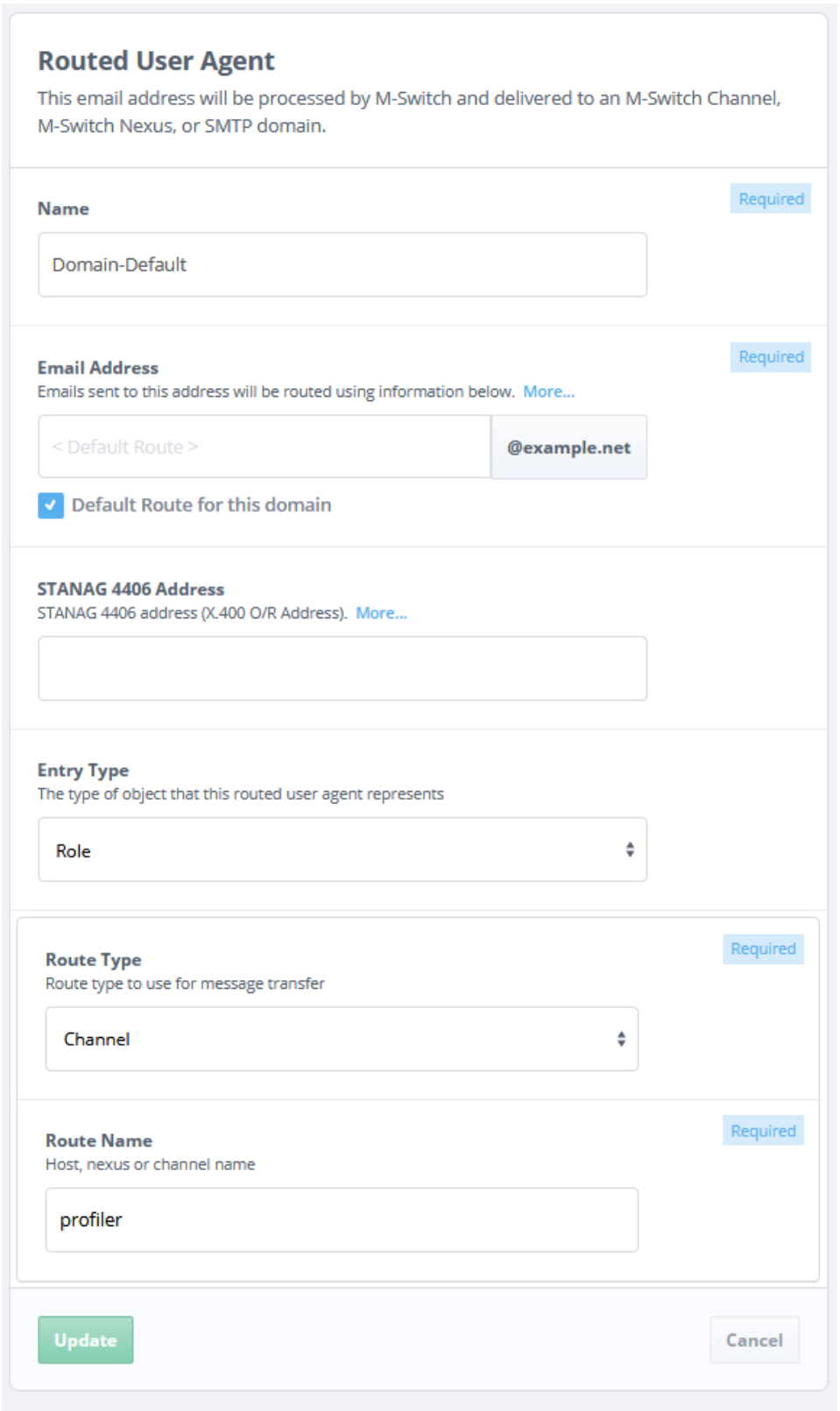
Figure 3.19. Routed User Agents



In order to create a new Routed User Agent an address and a route is required. Specify an email address or select **Default Route for this domain** and set a route value. A route value can be set by selecting a route type and setting its value.

The figure below displays the form for a **Routed User Agent**.

Figure 3.20. Routed User Agent Form



Appendix A Schema used by Cobalt

This appendix sets out the directory schema used by Cobalt. This schema is fully supported by M-Vault R18.0 and will be supported by future M-Vault releases. Cobalt can fully manage data in an LDAP directory that supports this schema. This schema is provided to facilitate configuration of an LDAP directory to use it.

A.1 Object Classes

A.1.1 Cobalt domain

```
Name: isodeCobaltDomain
SUP: top
Syntax: 1.3.6.1.4.1.453.16.8.2
Kind: Structural
MUST: (cobaltDomain, cobaltDsaAddress,
       cobaltDomainNamingContext)
MAY: (cobaltDomainRoleTypes, cobaltDomainUid,
      cobaltDomainDSAType, cobaltDomainSettings)
```

A.1.2 Cobalt role

```
Name: isodeCobaltRole
SUP: top
Syntax: 1.3.6.1.4.1.453.16.8.1
Kind: Structural
MUST: (cn, cobaltDomain)
MAY: (cobaltDomainRoleTypes, cobaltDomainUid,
      cobaltDomainDSAType, cobaltDomainSettings)
```

A.1.3 Cobalt entity type

```
Name: isodeCobaltObjectTypeOC
SUP: top
Syntax: 1.3.6.1.4.1.453.16.8.4
Kind: Auxiliary
MAY: (cobaltObjectType)
```

A.2 Attributes

A.2.1 cobaltRoleids

This attribute describes the user IDs that can occupy a Cobalt or domain administrator role. Each ID is one value of this multi-valued attribute.

```
Name: cobaltRoleids
Syntax: 1.3.6.1.4.1.453.16.9.2.1
Type: CaseIgnoreIA5String
Multi-value
```

Examples: *alice@example.net*

A.2.2 cobaltAccess

This attribute describes the Cobalt specific access controls defining read/write access for various resources.

```
Name: cobaltAccess
Syntax: 1.3.6.1.4.1.453.16.9.2.2
Type: BitString
Single-value
```

Example: *'00001000000010000000000000000000'B*

A.2.3 cobaltDomain

This attribute contains the value of domain.

```
Name: cobaltDomain
Syntax: 1.3.6.1.4.1.453.16.9.1.1
Type: CaseIgnoreIA5String
Single-value
```

Example: *example.net*

A.2.4 cobaltDsaAddress

This attribute contains the value of an LDAP address.

```
Name: cobaltDsaAddress
Syntax: 1.3.6.1.4.1.453.16.9.1.2
Type: CaseIgnoreString
Multi-value
```

Example: *ldap://gbwin2012.net:389*

A.2.5 cobaltDomainUid

This attribute describes an attribute value to be used for user id. Cobalt will use the value of this attribute to search in order to get the DN of the user for a given user ID.

```
Name: cobaltDomainUid
Syntax: 1.3.6.1.4.1.453.16.9.1.3
Type: CaseIgnoreIA5String
Single-value
```

Example: *mail*

A.2.6 cobaltDomainNamingContext

This attribute contains the value of naming context that holds Cobalt specific information.

```
Name: cobaltDomainNamingContext
Syntax: 1.3.6.1.4.1.453.16.9.1.4
Type: CaseIgnoreIA5String
Single-value
```

Example: *o=Cobalt*

A.2.7 cobaltDomainRoleTypes

This attribute contains the value of role types supported for a Cobalt domain.

```
Name: cobaltDomainRoleTypes
Syntax: 1.3.6.1.4.1.453.16.9.1.5
Type: CaseIgnoreIA5String
Single-value
```

Examples: *routedua,dlist,profiledua,mlist,xmpp-users,roleua,redi* and *xmpp-users*

A.2.8 cobaltDomainDSAType

This attribute describes the type of directory server that holds domain information. It can have one of 2 values:

- 0 - M-Vault (default)
- 1 - Active Directory (default)

```
Name: cobaltDomainDSAType
Syntax: 1.3.6.1.4.1.453.16.9.1.6
Type: Integer
Single-value
```

A.2.9 cobaltDomainSettings

This attribute contains the values of domain specific settings.

```
Name: cobaltDomainSettings
Syntax: 1.3.6.1.4.1.453.16.9.1.7
Type: CaseIgnoreString
Multi-value
```

Example: *acp127=true*

A.2.10 cobaltObjectType

This attribute describes the type of entity that an entry represents. It can have one of 3 values.

- 0 - User
- 1 - Role
- 2 - Organization

```
Name: cobaltObjectType
Syntax: 1.3.6.1.4.1.453.16.9.4.1
Type: Integer
Single-value
```

Appendix B Glossary

This appendix provides a glossary of terms.

Technical Terms used

Active Directory (AD)
A *Directory service* developed by Microsoft for the *Windows* networks. AD is a key component of *Windows Integrated Single Sign-On* solution. AD can act as *LDAP server*.

Authentication
The process of determining the identity of a communications partner.
See Also Authorization.

Authorization
A security service aimed at preventing unauthorized access to a service or capability. Once an identity has been established (see *Authentication*), authorization determines what services, data, and operations may be accessed by that identity.

Cobalt Server
Isode's Cobalt Server for provisioning users and roles in an *LDAP Server*.

Certificate Authority (CA)
An issuer of *certificates*. Also typically a publisher of certificate revocation information, commonly in the form of *CRL*, for the certificates it have issued. See [X.509] and and [RFC5280]. Sodium CA is a GUI tool for performing CA functions.
See Also Root Certificate Authority (Root CA), Root Certificate Authority (Root CA).

Certificate
A data object providing identity information for a subject entity (e.g., a person or computer system) securely bound to a public key by the certificate issuer, a *certificate authority*. See [X.509] and [RFC5280].

Certificate Chain
A certificate chain is a bundle of *certificates* which consists of an entity's certificate and, if the certificate is not *self-signed*, a sequence of certificates, each the issuer of the previous one, usually finishing at a *root*.

Certificate Revocation List (CRL)
A list of certificates which a *certificate authority* has revoked. See [X.509] and [RFC5280].

Certificate Signing Request (CSR)
A data object representing an entity's request for a *certificate authority* to issue a *certificate*. See [X.509] and [RFC2986]. Isode provides a number of tools to produce CSRs, such as Sodium

Directory
When referred to as *the Directory*, it is a distributed database built to *X.500* standards [X.509] and, in the context of Cobalt, accessed using *LDAP*
Alternatively, a container which holds files and other containers in a filesystem. Also referred to as a folder.

Directory Entry
A unit in *the Directory* representing one object and identified by its *Distinguished Name*. See [RFC4512].

Directory Service
The service provided by *the Directory* to its users.

Directory System Agent (DSA)
A server process which maintains and provides access to *the Directory*. In the context of Isode Cobalt, an *LDAP Server*.

Distinguished Name (DN)
The name for a *directory entry*. Cobalt uses the LDAP DN string format to represent DNs. See [RFC4514].

Domain Name
A name within the *Domain Name System*. See [RFC1035].

Domain Name System (DNS)
A service for providing a mapping between *domain names* (for example, `example.com`) and *IP addresses*. See [RFC1035].

IP address
An address which identifies a host machine on an Internet network. For IPv4, it is 32-bit number commonly written in dotted number notation of the form `192.0.1.100`. For IPv6, it is a 128-bit number commonly written in a notation of the form `2001:db8::100`.

Kerberos
An *authentication* protocol which relies on a *trusted third party* to issue *tickets* used to mutually authenticate clients and servers. See [RFC4120].

LDAP Client
A program which accesses *Directory* using *LDAP*. Examples: *Sodium*, *Cobalt*.

LDAP Server
A server process which provides *LDAP* access to *Directory*. Example: *M-Vault Server*.

Lightweight Directory Access Protocol (LDAP)
An Internet protocol used to provide access to the *Directory*. See [RFC4510].
See Also X.500.

M-Vault Server
Isode's Directory System Agent, an *LDAP server*.

M-Switch Server
M-Switch is Isode's Message Transfer Agent (MTA) that serves as the main component in a messaging system and supports Internet, X.400 and ACP127 messaging.

M-Box Server
The Isode IMAP (Internet Message Access Protocol) and POP (Post Office Protocol) server.

Public Key Infrastructure (PKI)
A collection of systems which support provisioning and use of *certificates*.

PEM
A format for representing *certificates*, keys, and other cryptographic objects. PEM stands for Privacy Enhanced Mail, a defunct standard for securing email. See [RFC1422].
See Also PKCS#12.

PKCS#12
An archive file format for bundling together a set of *certificates*, keys, and other cryptographic objects. See [RFC7292].
See Also PEM.

Root Certificate Authority (Root CA)
A *certificate authority* which utilizes a *self-signed* CA certificate when issuing *certificates*.

Self-Signed Certificate
A *certificate* which is signed by same entity which the certificate provide identity for.

Single sign-on (SSO)
Describes an access control system which allows a user, by authenticating to a system, to access multiple independent systems and/or services.
See Also Windows Integrated Single Sign-On (Windows SSO).

Sodium
Isode's directory data administration tool, an *LDAP client*. Though always written as "Sodium", Sodium is acronym standing for Secure Open Data, Identity and User Manager. Sodium is used for provisioning of users in *M-Vault Server*. deployments.

Transport Layer Security (TLS)
A protocol used by application protocols, such as HTTP, to provide communications security. It is formally known as Secure Socket Layer (SSL). See [RFC8446].

Trust Anchor (TA)
A certificate of a certificate authority trusted to issue (directly or indirectly) certificates for entities a party wishes to authenticate.

Trusted Third Party
An entity trusted by two parties, such as a client and a server, to facility *authentication* of one of the parties to the other or both parties to each other. In *public key infrastructures*, *certificate authorities*, when trusted, are trusted third parties.

Unix
Any operating system which complies with the *Single UNIX Specification*, such as the Linux and Solaris operating systems.
See Also Windows.

Windows
A family of operating system produced by Microsoft known as Microsoft Windows or simply Windows.
See Also Unix.

Windows Integrated Single Sign-On (Windows SSO)
Microsoft's *Kerberos* based *single sign-on* solution.

X.500
A set of standards devised for *the Directory*, developed jointly by the ITU-T and ISO/IEC. See [X.500].
See Also Lightweight Directory Access Protocol (LDAP).

Extensible Messaging and Presence Protocol (XMPP)
A collection of open standards for real-time communication, including those for instant messaging, presence, and multi-user chat. See [RFC6120].

Appendix C References

The documents listed in this appendix provide references to the appropriate standards and other sources of information.

If documents can be obtained electronically, the location is stated as part of the reference.

C.1 RFCs

RFC 4510
Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map [https://tools.ietf.org/html/rfc4510].

RFC 4512
Lightweight Directory Access Protocol (LDAP): Directory Information Models [https://tools.ietf.org/html/rfc4512]. K. Zeilenga. June 2006.

RFC 4514
Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names [https://tools.ietf.org/html/rfc4514].

RFC 5280
Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile [https://tools.ietf.org/html/rfc5280].

RFC 2986
Certification Request Syntax Specification [https://tools.ietf.org/html/rfc2986].

RFC 7292
Personal Information Exchange Syntax v1.1 [https://tools.ietf.org/html/rfc7292].

RFC 1422
Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management [https://tools.ietf.org/html/rfc1422].

RFC 8446
The Transport Layer Security (TLS) Protocol Version 1.3 [https://tools.ietf.org/html/rfc8446].

RFC 8259
The JavaScript Object Notation (JSON) Data Interchange Format [https://tools.ietf.org/html/rfc8259]. T. Bray, December 2017

RFC 6120
Extensible Messaging and Presence Protocol (XMPP): Core [https://tools.ietf.org/html/rfc6120].

RFC 1035
Domain names - implementation and specification [https://tools.ietf.org/html/rfc1035].