SCALIX™

Scalix Client Deployment Guide

Version 11.0.1

Client Deployment Guide

Published by Scalix Corporation
1400 Fashion Island Blvd., Suite 602
San Mateo, CA 94404-2061
USA

Product Version: 11.0.1

E:2.12.2007

Notices

The information contained in this document is subject to change without notice.

Scalix Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantibility and fitness for a particular purpose. Scalix Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Unix is used here as a generic term covering all versions of the UNIX operating system. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

Linux is a registered trademark of Linus Torvalds.

Red Hat, and Fedora are registered trademarks of Red Hat Software Inc. rpm is a trademark of Red Hat Software Inc.

SUSE is a registered trademark of Novell Inc.

Java is a registered trademark of Sun Microsystems Inc.

Microsoft, Windows XP, Windows 2000, Windows NT, Exchange, Outlook, Active Directory, and Internet Explorer are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

Restricted Rights Legend

Use, duplication, or disclosure is subject to restrictions as set forth in contract subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause 52.227-FAR14.

Contents

Introduction To This Guide

About This Guide

This guide outlines the preparation for, processes and procedures for deploying several different clients for use with the Scalix mail system.

Contents of this Guide

Included in this guide are the following topics:

How to Use This Guide

This guide uses the following typographical conventions:

Table 1: Typographical Conventions and their Explanations

Typographical Convention	Explanation
<Angle Brackets>	Values that you need to supply on your own are shown within angle brackets.
Numbered and alphabetized lists versus bullets	Numbered and alphabetized lists denote steps to be followed while bullets provide information.
Buttons	The larger boldface Verdana font indicates a button, a link, a field or any other UI element to click or press as well as a keyboard stroke. For example: Click **Finish**. Or In the **Username** field.

Table 1: Typographical Conventions and their Explanations

Typographical Convention	Explanation
Code	This smaller Lucida font indicates a piece of code to write or run. For example: `Launch scalix-installer.sh`
Document Names	References to other documents appear in italic font.
Italics	Indicates a directory path, a file or the name of a window or dialog box. For example: Go to */var/opt/scalix.* Or: You see the *Reply* screen.

Using the CLI

As with any procedure done on the command line, there may be more than one way to accomplish many of the tasks outlined in this manual. In many cases, these procedures are intended only as examples of how to complete a setup or configuration. If another method is more comfortable or more in keeping with your unique setup, it may be the best approach.

In addition, Scalix offers complete man pages for all commands. Please consult them whenever needed.

Identifying the Instance Home Directory

Throughout the various setup procedures, there are repeated references to the instance's home directory, known as "~". The location of this directory varies depending on how you ran your initial setup. For example, if you named the instance when you created it, the home directory becomes /var/opt/scalix/<instance>/s, where <instance> is a two-letter code created from the first and last letter of the instance name. If the instance is unnamed, the home directory becomes /var/opt/scalix/<nn>/s where <nn> is the first and last letter of the host name for that instance.

To determine the home directory for a particular instance, look in /etc/opt/scalix/instance.cfg for the appropriate value of OMDATADIR.

Related Documentation

Other Scalix product manuals include:

- Scalix Installation Guide
- Scalix Migration Guide
- Scalix Server Setup Guide
- Scalix Administration Guide
- Scalix API Guide
- Scalix Evaluation Guide

In addition, there are online help systems in:

- Scalix Management Console
- Scalix Web Access
- Outlook (if enabled for the Scalix connector)

Introduction To Scalix

This chapter introduces the Scalix system: Its different editions, access levels and licensing system.

Contents

This chapter includes the following information:

About the Scalix System

Capitalizing on a proven technology foundation and the openness of Linux, Scalix gives enterprise customers a simple to manage, highly reliable, and feature-rich Linux email and calendaring platform.This offers superior price and performance advantages with greater security, reliability, performance, openness and flexibility, when compared to other operating and messaging systems.

Based on open standards and a proven email server technology foundation, Scalix enables customers to create a robust and scalable environment that is flexible enough to adapt to their changing needs over time. The Scalix platform scales up to support organizations with hundreds of thousands of users and scales down for offices with fewer than one hundred users, making it a viable alternative for a broad range of organizations.

The Scalix architecture supports virtually any email client and device, without loss of functionality or data integrity. This means full-function support for popular clients like Microsoft Outlook and Novell Evolution, as well as the broad range of POP or IMAP clients available. Users can count on advanced features like enterprise calendaring and scheduling with real-time free/busy lookup, contact and task management, public folders, rich text formatting, offline folder synchronization, secure delegate access to calendar and email, email rules, resource booking and more.

About Scalix Product Editions

Scalix offers three editions of its powerful email and calendaring platform based on Linux and open systems: Scalix *Enterprise Edition, Small Business Edition* and Scalix *Community Edition*.

Scalix Enterprise Edition is the company's flagship product and is ideal for organizations that demand the full range of functionality in a commercial email and calendaring system. It includes multi-server support, unlimited number of *Standard* users, any number of *Premium* users, the full complement of Scalix advanced capabilities, and a wide variety of technical support options.

Scalix Small Business Edition targets organizations getting started with a commercial version of Scalix that do not have the higher end requirements of Enterprise Edition. It is functionally equivalent to Enterprise Edition except that it allows only single-server installations instead of multi-server, and does not include the capabilities for high availability and multi-instance support.

Scalix Community Edition is the free, single-server, unlimited-use version of the Scalix product and is great for cost-conscious organizations that desire a modern email and calendaring system but do not require advanced groupware and collaboration functionality for their entire user population. It includes unlimited Standard users, twenty-five free Premium users, a subset of Scalix functionality, and fee-based, incident-based technical support.

The following table compares the Scalix product editions in greater detail:

Table 1: Product Editions and their Features

Product Feature	Community Edition	Small Business Edition	Enterprise Edition
User Types			
Standard Users	Free, unlimited	Free, unlimited	Free, unlimited
Premium Users	Included: 25 Max: 25	Included: 50 Max: Unlimited	Min Purchase: 25 Max: Unlimited
Core Functionality			
Email & calendaring Server	Single-server	Single-server	Multi-server
Internal user directory	[X]	[X]	[X]
Choice of GUI-based or command line installation and administration	[X]	[X]	[X]
Unlimited POP/IMAP email client access	[X]	[X]	[X]
Native MS Outlook support (via MAPI)	Premium users only (max 25)	Premium users only	Premium users only
Fully functional AJAX web client (Scalix Web Access)	[X] (group scheduling in calendar for 25 premium users only)	[X] (group scheduling in calendar for all premium users)	[X] (group scheduling in calendar for all premium users)

Table 1: Product Editions and their Features

Native Novell Evolution support	[X] (group scheduling in calendar for 25 premium users only)	[X] (group scheduling in calendar for all premium users)	[X] (group scheduling in calendar for all premium users)
Public folders	Premium users only (max 25)	Premium users only	Premium users only
High availability	Not available	Not available	[X]
Multiple instances per server	Not available	Not available	[X]
Migration tools	Not available	[X]	[X]
Upgrade To Enterprise Edition	Via license key. Re-installation not required	Via license key. Re-installation not required	Not applicable
Mobile Access	[X]	[X]	[X]
Ecosystem Support			
Meta-directory support via LDAP	[X]	[X]	[X]
iCal support	[X]	[X]	[X]
Native Exchange Interoperability (via TNEF)	Not available	[X]	[X]
Active Directory integration with MMC plug-in	Not available	[X]	[X]
Anti-virus	Via flexible 3rd party interface	Via flexible 3rd party interface	Via flexible 3rd party interface
Anti-spam	Via flexible 3rd party interface	Via flexible 3rd party interface	Via flexible 3rd party interface
Archiving	Via flexible 3rd party interface	Via flexible 3rd party interface	Via flexible 3rd party interface
Wireless email & PIM	Email-only via POP/IMAP	Email & PIM via Notify	Email & PIM via Notify
Technical Support			
Community Forum	Free	Free	Free
Knowledgebase, Tech notes	Free	Free	Free
Incident-based Support	Fee-based	Fee-based	Fee-based
Software subscription	Not available	[X]	[X]
Premium 7x24 Support	Not available	[X]	[X]
Cost			
Licensing	Free, unlimited use	$995 for First 50 Premium Users	Per-user License; No Per-server Fees

About Scalix User Types

Scalix users can be defined as *Standard* or *Premium* users, as defined in the following:

Standard Users

Standard users gain access to a subset of Scalix functionality including email, personal calendar and contacts through Scalix Web Access and Novell Evolution as well as email access using POP/IMAP clients. The ability to deploy standard users is ideal for cost-conscious organizations with users who do not have high-end groupware and collaboration requirements. An unlimited number of standard users may be deployed with any Scalix edition for free.

Premium Users

Premium users have access to the full benefits and functionality of the Scalix email and calendaring system. The following Scalix product capabilities are available only to premium users:

- Native MS Outlook support (via MAPI)
- Group scheduling functionality including free/busy lookup in Outlook, Scalix Web Access and Evolution clients
- Access to public folders
- Wireless email and PIM

Any number of licensed premium users may be deployed with Scalix Enterprise Edition. Scalix Community Edition is limited to a maximum of twenty-five (25) free premium users, who enjoy many of the features available to Enterprise Edition premium users.

Flexible, Cost-Effective Email For Everyone

The distinction between standard and premium users provides organizations with the flexibility to cost-effectively provide email for all users. For example, manufacturers and retailers may desire headquarters staff to be designated as premium users as they require advanced groupware capabilities, while less demanding users, such as shop floor or store personnel, would be satisfied as standard users with only email and personal calendaring capabilities. Similarly, educational institutions may decide that faculty and staff are premium users that need advanced collaboration capabilities while students are standard users that just need email and personal calendaring. There is no cost for deploying standard users with either Scalix Community Edition or Scalix Enterprise Edition.

Required Licenses

Scalix *Community Edition, Small Business Edition* and *Enterprise Edition* use the same installer. The main difference is that Small Business Edition and Enterprise Edition require a license key while Community Edition does not. Additionally, if you are a Scalix Community Edition customer, you can only perform the “typical” installation, in which all the Scalix components are stored on a single host computer.

To activate your Scalix system as either a Small Business or Enterprise Edition system, you must enter a license key at a strategic point in the installation process. Please obtain your Scalix license key and have it ready for use before installing.

You may proceed with the installation without a license key, however, your system is treated as a Community Edition system and your users as Standard users until the correct license key is entered by means of the *Scalix Management Console*.

Additionally, you can install Scalix Enterprise Edition onto a single host, or distribute the primary components onto separate hosts—both of which are detailed fully in this guide.

Scalix Architecture

This chapter introduces the Scalix architecture: Its components, ecosystem, clients and more.

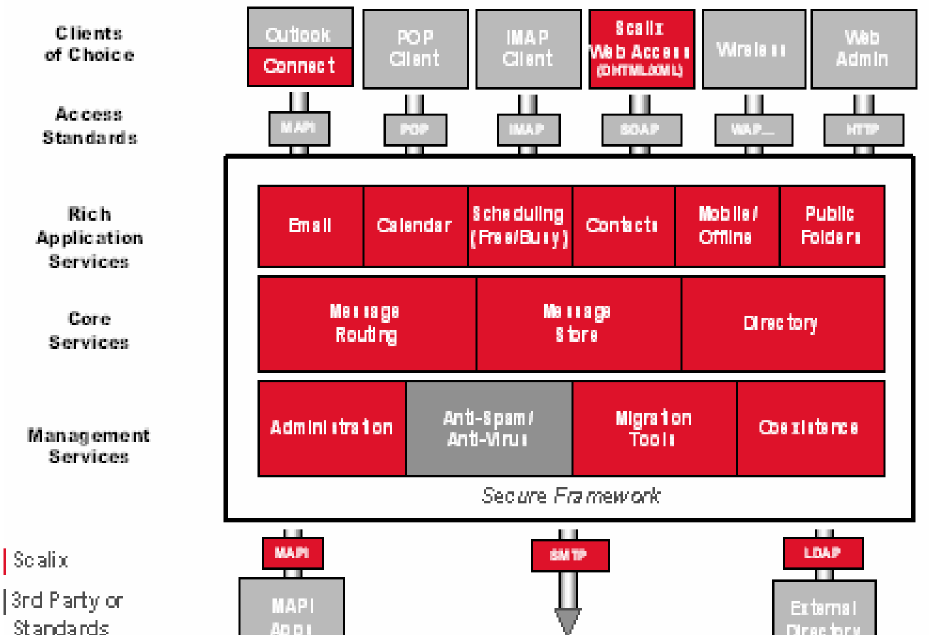
Contents

This chapter includes the following information:

About Scalix Architecture

The Scalix mail system is a client-server architecture based upon international standards and an open architecture that allows the flexibility to use many different client and third-party applications to send and receive messages between multiple Scalix servers, either inside or outside a company’s network.

A Linux operating system environment establishes the base for the actual Scalix platform.



Scalix Components

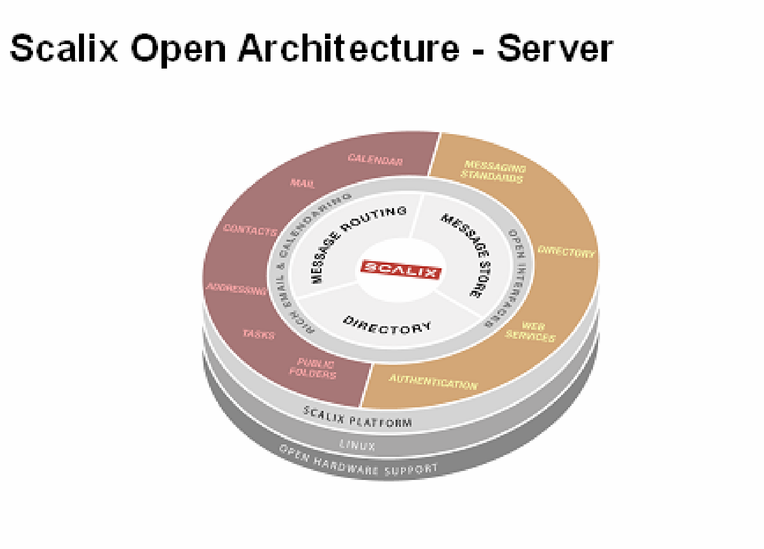
Scalix has two main components, the server and its clients:

The Scalix Server(s)

The Scalix server provides message storage, collection, dispatch, routing and delivery. It not only contains eMail messages, but also PIM/Groupware information such as calendaring data, contacts and task lists. In addition, it manages message delivery and provides or integrates with add-on services such as virus scanning, anti-spam or content-type conversion.

Server management is done in two places:

- Through the Scalix Management Console (aka SAC or Scalix Management Console), an easy-to-use GUI for frequently-undertaken, day-to-day tasks such as creating users, managing public distribution lists, assigning permissions and more.
- On the command line for more advanced configurations such as backups, integration of anti-virus and anti-spam applications, setting up authentication, etc.



Clients

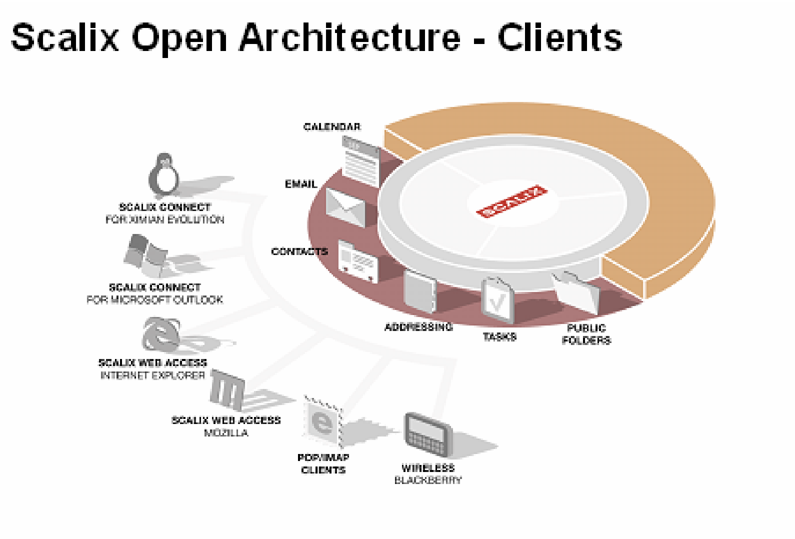
The clients are applications that allow users to create, view and manipulate messages, notify users when new mail arrives, access address directories, track the progress of message delivery, configure auto actions and more. They use the IMAP, POP and UAL (User Access Layer) protocols to connect into the Scalix server, where they access the message store, directory and personal folders. They are handled by remote client interfaces.

Scalix operates seamlessly and transparently with many different clients, including:

- Microsoft Outlook
- Novell Evolution
- IMAP and POP clients such as Mozilla Thunderbird, Outlook Express and Eudora
- Its own native client, Scalix Web Access (SWA).

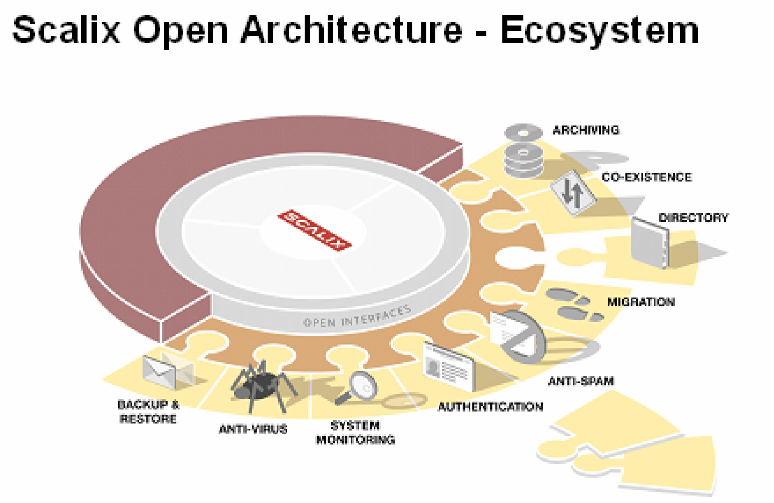
Client management is done in five places:

- The Management Console to set access levels, global server properties and more
- CLI to set access levels and more
- Configuration files to set properties, logging customizations and more
- Scalix Connectors to enable the use of the Scalix server with clients such as Microsoft Outlook and Novell Evolution
- 3rd Party Administration Interfaces


Scalix Open Architecture - Clients
CALENDAR
EMAIL
CONTACTS
SCALIX
SCALIX CONNECT FOR XIMIAN EVOLUTION
SCALIX CONNECT FOR MICROSOFT OUTLOOK
SCALIX WEB ACCESS INTERNET EXPLORER
SCALIX WEB ACCESS MOZILLA
POP/IMAP CLIENTS
WIRELESS BLACKBERRY
ADDRESSING
TASKS
PUBLIC FOLDERS

The Ecosystem

The ecosystem surrounding the Scalix server places a strong emphasis on open interfaces. This provides flexibility for integrating with a variety of best-of-breed solutions in important areas such as anti-virus protection, authentication, backup and recovery tools. The system broadly complies with messaging standards ranging from RFC 822 and continues to include MAPI, POP3, IMAP4, MIME, SMTP, and LDAP.


Scalix Open Architecture - Ecosystem

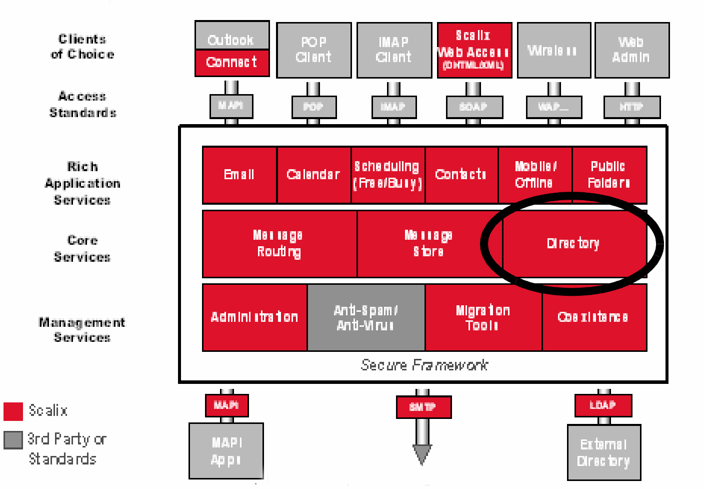
Directories

Scalix also holds a directory of known users that enables auto-finishing features for addressing of email messages.

The address directories are databases that clients use to look up names and addresses. The Scalix address directories can contain Scalix and non-Scalix users, other administrator-configurable information such as job titles and phone numbers, and can be shared with other Scalix directories or synchronized with MS Exchange servers.

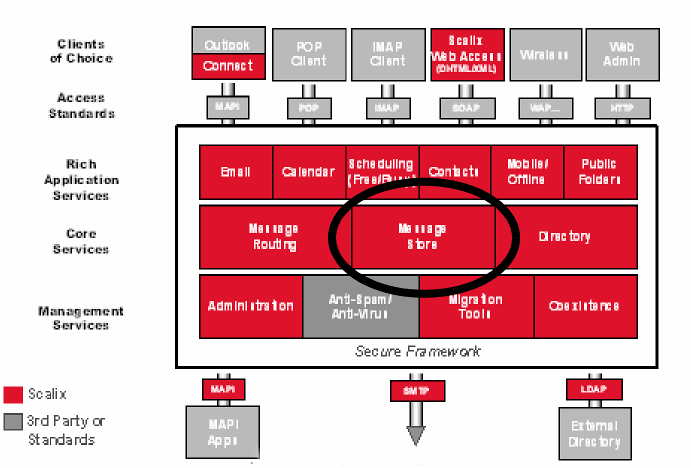
Directories are searchable by any number of attributes. They contain many standard attributes and also some that are rarely used.

Note	Scalix is a system that has grown up over time and it's good to note that it used to be (and still is to some respect) based on X.400. Addresses are still based on X.400 OR names and the X.400 nomenclature.



The Message Store

The message store is not a database. It's a collection of flat Linux files, held in file system directories on the Scalix server. It holds new messages received as well as messages in transit For clients that use the message store (server-based clients), it also holds old messages that are files for reference in folders, copies of outgoing messages, draft messages, in preparation, private distribution lists, personal information such as calendaring, tasks and journaling information and Bulletin Boards, or public folders, which are accessible to multiple users.



Routing and Local Delivery

The Service Router is the process on the Scalix Server that decides (or routes) where a message is supposed to go.

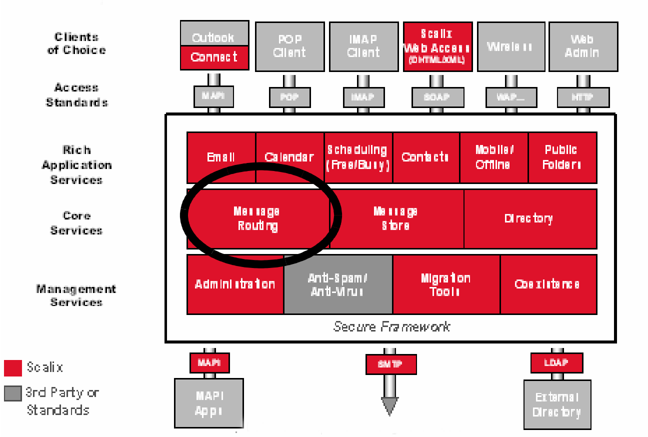
The Local Delivery process is the process on the Scalix Server that determines where a message ends up in the message store on the local machine for a local user.

Scalix's routing services check the recipients in a message, and then send it on to be either delivered locally to another Scalix system, or to leave Scalix entirely via a gateway. These routing services also create NDNs, or Non Delivery Notifications, when a message cannot be delivered due to an addressing fault. These NDNs go to the originator of the message as well as to the configured error manager.

Once the message has arrived at its destination, the local delivery process places it in the recipient's in tray.

Local Delivery and Service Router, together, also handle Public Distribution List Expansion and address resolution, up to the point where they can try to correct misspelled email addresses by phonetic matching.

As all messages in the system must pass through the Service Router, this also becomes the preferred point of integration for virus scanners, filtering rules and message archiving.

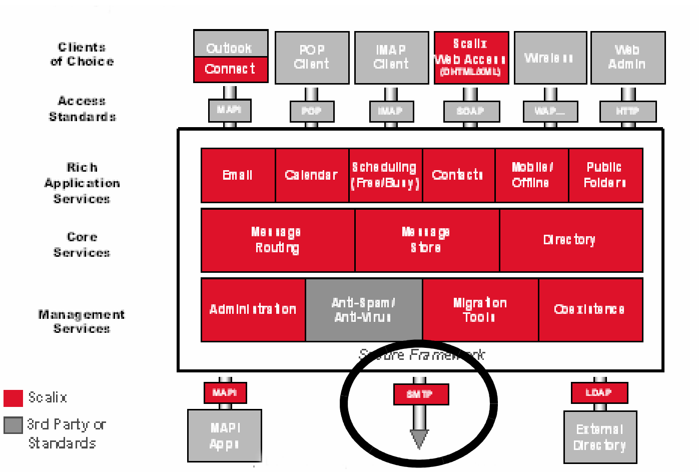


Gateways

Gateways are a way of passing messages out of the Scalix network to different mail environments. The gateway must convert the outgoing message from a Scalix format to one that an external service can send, and then convert the addresses into a format that the target environment can handle, such as an SMTP address.

Scalix comes with a standard SMTP gateway that converts Scalix formatted messages to SMTP formatted messages and vice-versa. This gateway is called the “Unix Mail Gateway” or “Internet Mail Gateway” on Scalix, but, because SMTP is the most important standard in messaging interoperability, it also connects to almost any other messaging system.

Other gateways can be written for connection to other mailing systems.



Transports

The Transport Service on the Scalix server is called the “Sendmail Interface”.

Transports are services that Scalix uses to pass Scalix-formatted messages to other Scalix services. Scalix uses Sendmail and SMTP format to send messages between servers in the Scalix network, but other connections can be written.

Search and Index Service

The Scalix Search and Index Service provides realtime indexing of all private and public folder messages. Built on the open-source Lucene technology, it enables sub-second, mailbox-wide message retrieval. It is localizable, and its Web services interface is available.

Messaging Service

Scalix Messaging Services are server-based REST APIs for email and calendaring application integration. They enable integration of Linux messaging with critical applications such as content management, mobile solutions, customer relationship management (CRM) software, or enterprise resource planning (ERP) packages. Calendaring functions and data can be integrated directly into other applications, or the data from other applications can be directly integrated into email and calendaring.

Deploying Scalix Connect for Microsoft Outlook

This chapter covers deployment of the Microsoft Outlook connector, a plug-in that enables the Outlook client for use with the Scalix Server.

Contents

This chapter includes the following information:

Features of Scalix Connect for Outlook

With Scalix, users continue to enjoy the functionality and productivity they are accustomed to with Microsoft Outlook.

Scalix Connect for Microsoft Outlook provides transparent, full function support for the Outlook email client on the Linux platform. It enables connectivity between Outlook and the Scalix Server, and supports the MAPI protocol for compatibility with the full range of functionality offered by Outlook.

Scalix Connect for Microsoft Outlook is a MAPI Service Provider that allows users to connect to the Scalix Server using the Outlook client. Scalix Connect for Microsoft Outlook uses native MAPI Properties and a MAPI Default Message Store to unlock the group productivity functionality of Outlook operating in "Corporate or Workgroup" mode, such as rich electronic messaging, shared directories, shared calendars, free/busy lookups, public folders, and more.

Scalix Connect also enables advanced modes of operation such as offline folder synchronization and delegate access to mailboxes, calendars, and personal folders.

There are, however, some differences in the way Scalix handles Outlook features.

Note	In some cases where calendar items are created and accepted on Exchange 5.5 versus Scalix servers, you may see dual entries for certain calendar items. If so, make sure your Exchange GAL (Global Address List) and Scalix system directory are in sync.

Unsupported Features

Scalix Connect does not support:

- Journaling
- Forms
- Group by View
- Message recall
- MSN messenger integration

Additional Services in the Same Profile

Scalix Connect for Microsoft Outlook supports simultaneous use of additional services such as PST and LDAP address lookup, but not Internet Mail Service or the Microsoft Exchange Service in the same profile. Scalix Connect for Microsoft Outlook specifically does not support the features for Hotmail integration and the ability to send using a specific email account selection.

Scalix Connect for Microsoft Outlook does not allow direct delivery of incoming messages into a PST-based inbox.

System Requirements

There are some minimum system requirements for the use of Scalix Connect for Microsoft Outlook:

Components	Requirements
Client Software	Outlook 2000, XP, and 2003 * A legacy version of Scalix Connect forOutlook 98 and 2000 on Windows NT 4.0 is available to enterprise customers through Scalix support.
Operating System	Fedora Core 4 (or later) RedHat Enterprise Linux 4 (or later) *You can download and build the connector on another operating system, but it would be unsupported.
Disk Space	The installation of Scalix Connect for Microsoft Outlook requires approximately 13 MB of free disk space. 2 MB of this space is for the Microsoft Data Access Objects that might be present on user PCs already.
Access/Permissions	You must be a Power User, or have administration permissions to install Scalix Connect for Microsoft Outlook.
Ports	Ports 5729 and 5757 must be available

Additional Information

Detailed system requirements for Outlook is located at:

- Outlook 2000: http://support.microsoft.com/default.aspx pr=out2K&SD=OFFN
- Outlook 2002: http://support.microsoft.com/default.aspx?pr=ol2002&SD=OFFN
- Outlook 2003: http://support.microsoft.com/default.aspx?scid=fh;en-us;out2003

Before You Begin

Some application deployment solutions such as Microsoft's Systems Management Server require a Microsoft Windows Installer (.msi) file in order to deploy an application across multiple desktops without user intervention. To accomodate this, you can generate an .msi file for Scalix Connect. This .msi file contains all of the necessary information to accomplish a complete installation of Scalix Connect for Outlook and it can be used with large-scale application deployment solutions.

To generate an .msi file:

1. On the command line, execute the following command.

```
setup.exe /e
```

2. A file called "Scalix Connect for Microsoft Outlook.msi" creates in the same directory as the setup.exe application.

3 The .msi files for both English and German languages are located in the /deployment directory of the CD or tarball.

Installing Scalix Connect on an Outlook Client

Scalix Connect can only be installed on a client computer by a user with “Administrator” permissions on the computer.

Alert	All client PCs must be able to resolve the Scalix server by hostname and fully-qualified domain name. You cannot use IP addresses to specify the Scalix server when configuring Scalix Connect for Outlook.

Any upgrades to Scalix Connect can be made by the client system’s user, provided you have some means of distributing a copy of the updater to them.

You can also set up an automated update/upgrade process for Outlook users, as detailed in “Performing Automatic Upgrades” on page 42.

To install Scalix Connect on a client computer:

1 Log in to the user’s computer with an administrator account.

2 Navigate to the directory with the Scalix installation files and double-click the **setup.exe** icon.

3 The installation wizard launches, followed by the Welcome window. Click **Next** to proceed.

4 The wizard begins installing files to the Outlook system MAPI directory. A progress bar tracks the status.

5 When installation is complete, click **Finish** to exit the installer.

Enabling SmartCache

The Scalix SmartCache feature installs a copy of the user’s mailbox on the local PC for faster response time and reduced bandwidth usage. This feature must be enabled during profile creation and once enabled, cannot be reversed. You MUST create a new profile to enable SmartCache.

To enable SmartCache:

1 From the **Start** menu, go into the **Control Panel** and double click **Mail**.

2 At the Mail Setup window, click **Show Profiles.**

3 You get the Mail window. Click **Add,** give the new profile a name, then click **OK**.

4 This launches the Microsoft Email Accounts wizard. Select the radio button by **Add a new email account**.

5 At the Email Accounts window, select **Additional Server Types**, then click **Next.**

6 In the next screen, the field pre-populates with the Scalix server. Select that and click **Next**.

7 At the next Email Accounts screen, make sure the **User name** field has the Scalix username, and type in the Scalix user's password**.** If desired, you can click the advanced button to verify the mail server. When finished, click **OK**.

8 Two additional check boxes appear. Check the box next to **Use SmartCache** and **Optimize for Mobile Use** if desired. The **Optimize for Mobile Use** button sets the cache to populate with header information only, so that minimal data about each email is synchronized.

9 At the Local Storage screen, select the desired location of your cache, if different from the default, then click **OK**.

10 Back at the Mail window, select which profile you want to use (click the radio dial by **Always use this profile**), or whether you want to select each time you sign on (**Prompt for a profile to be used**).

11 Continue profile installation as usual.

Microsoft Outlook Security Model

The Outlook E-mail Security Model provides protection against viruses that come in to a user's Inbox as attachments.

To use the security model, create one or more security files on the Scalix Server that contain the configuration parameters described in "Security File Configuration Options" on page 24. You can create one file for all users, or many files for individual users or groups of users.

Using the Outlook Security Model for System-wide Use

To use the Outlook security model for general (system-wide) use:

1 On the Scalix Server, open a new document/file with any Linux text editor application.

2 Enter the security settings described in "Security File Configuration Options" on page 24. To view an example of a security file, see "Security File Example" on page 28.

3 Save the file. You can use any name to identify the security file.

4 Add the following entry in the file ~/sys/general.cfg:

```
UAL_OUTLOOK_SECURITY_FILE=~/sys/security file
```

This setting instructs Outlook to use the security parameters in the file you created in the steps above for all users on the server.

Using the Outlook Security Model for an Individual User

To use the Outlook security model for an individual user:

1 On the Scalix Server, go to the directory ~/sys/ and enter the following command using the common name of the user to which you want to apply security settings:

```
omshowu -n "/CN=common name" -G
```

This command lists (among other items) the Internal User ID of a user.

2 Use the Internal User ID value to identify the configuration file you must edit to apply the security settings. All user configuration files are in the directory `~/sys/`. The file names are derived from the Internal User ID values of the users to which the files respectively belong.

   If the user to which you want to apply the security settings does not have a configuration file:

   - Obtain the Internal User ID of the user by entering the following command:

     ```
     omshowu -n “/CN=common name” -G
     ```

   - Open a new document/file with any Linux text editor application.
   - Save the file to the directory `~/sys/` using the Internal User ID of the user to which you want to apply the security settings.

3 Open a new document/file with any Linux text editor application.

4 Enter the security settings described in “Security File Configuration Options” on page 24. To see an example of a general security file, see “Security File Example” on page 28.

5 Save the file. You can use any name to identify the security file.

6 Add the following entry to the appropriate user configuration file in the directory ~/sys/.

   ```
   UAL_OUTLOOK_SECURITY_FILE=~/sys/security file
   ```

   This setting instructs Outlook to use the security parameters you created in the steps above for this user only.

   Or you can enter the following in the user configuration file:

   ```
   UAL_OUTLOOK_SECURITY_FILE=~/sys/outlook.sec
   ```

If you created a system-wide security configuration file (as described in “Using the Outlook Security Model for System-wide Use” on page 23), this setting instructs Outlook to use the security settings in outlook.sec for this user, while the security settings for all other users is controlled by the system-wide security configuration file.

If you do not specify a security file in the file general.cfg or in the user configuration file(s), Outlook uses the security settings in the file ~/sys/outlook.sec. If the file outlook.sec does not exist, Outlook uses its default security settings.

Note	You can create groups of users with the same security settings by specifying the same security file in the user configuration files of the users to which you want to apply the security settings.

Security File Configuration Options

The security file you create contains configuration parameters that allow you to determine how the security model operates. The security file can contain the following sections:

- [Level 1 File Extensions]
- [Level 2 File Extensions]
- [Miscellaneous Attachment Settings]

- [Miscellaneous Custom Form Settings]
- [Programmatic Settings]

Make sure the section headings in the security file display as they appear in the list above, including the square brackets ([ ]). You do not have to include a section heading in the security file if you do not need to configure the parameters for that section.

See "Security File Example" on page 28 to view an example of a security file.

Level 1 File Extensions

The following table lists the parameters for the [Level 1 File Extensions] section.

Table 1: Configuration Parameters and their Descriptions

Parameter	Description
Add=	Enables you to add file types to the Level 1 file list. Incoming messages with attachments that match the file type(s) for this parameter are removed from the message. To view a list of file-type extensions that are generally considered "unsafe", go to: http://support.microsoft.com/support/kb/articles/Q262/6/31.asp
Remove=	Enables you to remove file types from the Level 1 list. Incoming messages with attachments that match the file type(s for this parameter are retained in the message. When you remove a file type from the Level 1 file list, the file type remains on the Level 2 file list. To remove a file type from both the Level 1 and Level 2 file lists, include Remove=value in both the [Level 1 File Extensions] and [Level 2 File Extensions] sections.

The values you enter for these parameters are file-type extensions without the period. For example, enter Add=bat and not Add=.bat for Batch files. Also if you want to enter multiple file type extensions for a parameter, separate the extensions with a semi-colon. For example Add=bat;doc;rtf;xls.

Level 2 File Extensions

The following table lists the parameters for the [Level 2 File Extensions] section.

Table 2: Configuration Parameters and their Descriptions

Parameter	Description
Add=	Enables you to add file types to the Level 2 file list. Incoming messages with attachments that match the file type(s) you specify for this parameter are removed from the message..
Remove=	Enables you to remove file types from the Level 2 list. Incoming messages with attachments that match the file type(s) you specify for this parameter are retained in the message.

The values you enter for these parameters are file-type extensions without the period. For example, enter Add=bat and not Add=.bat for Batch files. Also if you want to enter multiple file type extensions for a parameter, separate the extensions with a semi-colon. For example Add=bat;doc;rtf;xls.

Miscellaneous Attachment Settings

The following table lists the parameters for the [Miscellaneous Attachment Settings] section.

Table 3: Configuration Parameters and their Descriptions

Parameter	Description
ShowLevel1Attachments=	When you set this parameter to TRUE, users can access all attachments in the [Level 1 File Extensions] section. When you set this parameter to FALSE, users can only access attachments specified in the Add= parameter of the [Level 1 File Extensions] section.
DoNotPromptLevel1Attachments OnSend=	When you set this parameter to TRUE, users do not receive a warning when they send an item containing a Level 1 attachment. After the item is sent, users cannot view or access the attachment. When you set this parameter to FALSE, users receive a warning message.
DoNotPromptLevel1Attachments OnClose=	When you set this parameter to TRUE, users do not receive a warning when they close an item containing a Level 1 attachment. When the item is closed, users cannot view or access the attachment. If you set this parameter to FALSE, users receive a warning message.
AllowActivationOfOLEObjects=	When you set this parameter to TRUE, users can double-click on an embedded attachment, such as Microsoft Excel spreadsheet, and open it in the program. Set this parameter to FALSE to disable this capability. If users are using Microsoft Word as their e-mail editor, users can still open embedded objects even if you set this parameter to FALSE.
ShowOLEPackageObjects=	When you set this parameter to TRUE, users can view all packaged OLE objects (a package is an icon that represents an embedded or linked OLE object). Set this parameter to FALSE to disable this capability. When users open the package icon, the application that created the OLE object executes the OLE object.

Miscellaneous Custom Form Settings

The following table lists the parameters for the [Miscellaneous Custom Form Settings] section.

Table 4: Configuration Parameters and their Descriptions

Parameter	Description
EnableScriptsInForm=	When you set this parameter to TRUE, users can run scripts using Outlook forms. Outlook allows scripts to run when the script and the layout are contained in the message itself.

Table 4: Configuration Parameters and their Descriptions

Parameter	Description
ExecutingCustomActionViaOOM=	Specify the action Outlook takes when a program attempts to run a custom action using the Outlook object model. A custom action can be created to reply to a message and circumvent the programmatic send protections. The actions are: `Prompt`: Users receive a message enabling them to allow or deny the operation. `Approve`: Users do not receive a prompt and the operation is allowed. `Deny`: Users do not receive a prompt and the operation is denied.
AccessingItemPropOfaControl OnForm=	Specify the action Outlook takes when a user adds a control to a custom Outlook form and then binds that control directly to any of the Address Information fields. By doing this, code can be used to indirectly retrieve the value of the Address Information field by obtaining the Value property of the control. `Prompt`: Users receive a message enabling them to allow or deny the operation. `Approve`: Users do not receive a prompt and the operation is allowed. `Deny`: Users do not receive a prompt and the operation is denied.

Programmatic Settings

The following table lists the parameters for the [Miscellaneous Custom Form Settings] section. The parameters in this section require that you enter one of the following values:

- Prompt: Users receive a message enabling them to allow or deny the operation.
- Approve: Users do not receive a prompt and the operation is allowed.
- Deny: Users do not receive a prompt and the operation is denied.

Table 5: Configuration Parameters and their Descriptions

Parameter	Description
SendingItemsViaOOM=	Specify the action Outlook takes when a program attempts to send mail programmatically using the Outlook object model.
SendingItemsViaCDO=	Specify the action Outlook takes when a program attempts to send mail programmatically using CDO (Collaboration Data Objects).
SendingItemsViaSMAPI=	Specify the action Outlook takes when a program tries to send mail programmatically using Simple MAPI.
AccessingAddrBookViaOOM=	Specify the action Outlook takes when a program attempts to access an address book using the Outlook object model.
AccessingAddrBookViaCDO=	Specify the action Outlook takes when a program tries to access the address book using CDO.
ResolvingNamesViaSMAPI=	Specify the action Outlook takes when a program tries to access the address book using Simple MAPI.

Table 5: Configuration Parameters and their Descriptions

Parameter	Description
AccessingAddrInfoViaOOM=	Specify the action Outlook takes when a program attempts to access address information through Outlook object model (for example, access a recipient field such as “To”).
AccessingAddrInfoViaCDO=	Specify the action Outlook takes when a program attempts to address information through CDO (for example, access a recipient field such as “To”).
OpeningMessagesViaSMAPI=	Specify the action Outlook takes when a program attempts to access a recipient field (such as “To”) using Simple MAPI.
RespondingToMeetingsAndTasks ViaOOM=	Specify the action Outlook takes when a program attempts to send mail programmatically using the Respond method on task requests and meeting requests. This method is similar to the Send method on mail messages.
ExecutingSaveAsViaOOM=	Specify the action Outlook takes when a program attempts to use (programmatically) the Save As command in the File menu to save an item. When an item is saved, a malicious program can search the file for e-mail addresses.
AccessingFormulaPropOfObject InOOM=	Specify the action Outlook takes when users add a Combination or Formula custom field to a custom form, and bind it to an Address Information field. By doing this, code can be used to indirectly retrieve the value of the Address Information field by obtaining the Value property of the field.
AccessingAddrInfoViaUserPropsInOOM=	Specify the action Outlook takes when a program attempts to search mail folders for address information through “UserPropertiesFind” in the Outlook object model.

Security File Example

The following information provides an example of an Outlook security configuration file. The parameters in this example are described in “Security File Configuration Options” on page 24.

```
[Level 1 File Extensions]
Add=bat;vbs
Remove=doc;xls

[Level 2 File Extensions]
Add=zip

[Miscellaneous Attachment Settings]
ShowLevel1Attachments=FALSE
DoNotPromptLevel1AttachmentsOnSend=FALSE
DoNotPromptLevel1AttachmentsOnClose=FALSE
AllowActivationOfOLEObjects=FALSE
ShowOLEPackageObjects=FALSE

[Miscellaneous Custom Form Settings]
EnableScriptsInForm=FALSE
ExecutingCustomActionViaOOM=PROMPT
AccessingItemPropOfaControlOnForm=PROMPT
```

```
[Programmatic Settings]
SendingItemsViaOOM=PROMPT
SendingItemsViaCDO=PROMPT
SendingItemsViaSMAPI=PROMPT
AccessingAddrBookViaOOM=PROMPT
AccessingAddrBookViaCDO=PROMPT
ResolvingNamesViaSMAPI=PROMPT
AccessingAddrInfoViaOOM=PROMPT
AccessingAddrInfoViaCDO=PROMPT
OpeningMessagesViaSMAPI=PROMPT
RespondingToMeetingsAndTasksViaOOM=PROMPT
ExecutingSaveAsViaOOM=PROMPT
AccessingFormulaPropOfObjectInOOM=PROMPT
AccessingAddrInfoViaUserPropsInOOM=PROMPT
```

Enabling Password Authentication

If needed, you can enable password authentication in a single-sign on environment.

When running in an environment created for single sign-on, the Scalix Connect profile creation wizard tries to create a SSO profile first. If you prefer password-based authentication for some users, you can set that up from within your own Windows session.

To enable password-based authentication for a single user:

1. Before running the profile wizard, use the Windows registry editor on the respective client machine to locate the HKEY_CURRENT_USER\Software\Scalix\Scalix\MAPI\Profiles\ on the client machine.
2. Under this key, locate the ShowAdvButton attribute and change the value to non-zero.
3. This enables an "Advanced" button on the final profile wizard screen, which allows an override of the usual single sign-on profile setup.
4. Run the profile creation wizard as usual.

Automating Scalix Connect Installation

If your mail system potentially involves hundreds or even thousands of desktops that will require Scalix Connect, some installation-automation resources exist, to help reduce the amount of end-user interaction. The following three files can be used for an automated deployment of Scalix Connect, on Windows desktops or through Windows desktop deployment tools.

Setup.exe - This program actually installs Scalix Connect. When the Scalix release (most often distributed as a “zipped” tar file) is unpacked, it will contain a directory /software/scalix_connect_outlook/, where the setup.exe program will be located.

Install.ini - This configuration file helps you to customize the Outlook profile settings. A template install.ini file is included with the latest Scalix release, and can be found in the /software/scalix_connect_outlook/deployment directory.

sxpro.exe - This program can be used to create Outlook profiles. It is not included within the directories of the Scalix release, but it can be extracted from the setup.exe program by passing the /e switch.

```
c:\Connect\setup.exe /e
```

Note	See the release notes for the location to which the sxpro.exe is extracted.

Method 1 – End-User Driven Deployment

With this process, the installation of the Scalix Connect program (setup.exe) and creation of the Scalix profile occurs through end-user interaction. While the end-user interaction is simple and intuitive, it does require that they know the answer to certain prompts.

Scalix Connect 9.4.2 or later requires Administrator or Power User privileges on the workstation where it is installed – subsequent upgrades to later versions of Connect that utilize the auto-upgrade service only require Power User.

To set up end-user driven deployment:

1. Establish a shared networkable directory (samba share or any network file share) that all users can access.
2. In this directory (\\servername\sharename\Connect), you would place the following files, provided by Scalix.

   ```
   setup.exe

   install.ini
   ```

3. Customize the install.ini file, so that an Outlook profile can be created and automatically populated with some of the necessary Scalix mailbox data. (This provides a level of standardization across all desktops.)

   The following general guidelines should be used.

   - The value noted in the second line should point to a local drive and directory where the setup.exe can write data (hence the c:\windows\temp location specified above).
   - Directory paths must include a double backslash (\\).
   - Use “InstallMigrateExProfile=1” only if there is an existing default profile and you wish to migrate some data from that profile into the newly created Scalix profile.

- Never use an IP address for the InstallMailServerName – a valid FQDN (either an A-record or C-name record) that is registered in DNS is required. Using a C-name (that points to an A-record) will be beneficial if there is ever a need to move users to a new physical server.

The following is a valid example of a working install.ini file:

```
[Install Flags]

InstallMailServerName=scalix01.company.com

InstallLogfileUploadLocation=c:\\windows\\temp\\install.log

InstallDefaultProfileName=Scalix1
```

```
InstallWithServerStore=1

InstallAllowSavedPassword=0

InstallSetDefaultProfile=1

InstallLogFileLocation=c:\\windows\\temp\\log.txt

InstallMigrateExProfile=0
```

The install.ini contains parameters, which can ultimately help streamline the installation, and help in the auditing the process of who has installed the Scalix Connect software. The settings in the install.ini can help standardize all profile settings across a group of people or an organization.

InstallMailServerName=

This should be set to the name of the Scalix server, which users will connect to, or optionally point to a DNS alias record (C-Name)

InstallLogFileUploadLocation=

This should be set to a directory on either the desktop PC or a mapped network drive. The user must have appropriate permissions (read/write).

InstallDefaultProfileName=

This value will be used for the Outlook profile name that setup creates on each desktop.

InstallWithServerStore=

This should be set to 1.

InstallAllowSavedPassword

In most deployments, you should set this value to 0, to force users to enter their password when they log into their Scalix mailbox. With this value set to 1, users will not have to enter their password after their initial login to Scalix server.

InstallSetDefaultProfile=

By setting this value to 1, the new Outlook profile created by the setup (or sxpro) program, will be the profile that is opened the next time Outlook is opened.

InstallLogFileLocation=

This should be set to a directory on either the desktop PC or a mapped network drive. The user must have approriate permissions (read/write).

InstallMigrateExProfile=

With this value set to 1, it will migrate some information from the existing default profile to the newly created profile.

Additional Notes about the install.ini File

- The InstallDefaultProfileName value should be set to some standard that is reflective of a group of users or a Scalix server they access.

- The InstallMailServerName value is where the user's mailbox would exist, hence in a multiple server environment it may be required to configure different install.ini files in different sub-directories for groups of users.
- The InstallWithServerStore=1, InstallAllowSavedPassword=0 and InstallSetDefaultProfile=1 are typically not changed.
- The two parameters that specify locations of log files, can be configured to point to a network drive and directory, but again this must be statically mapped by the client.
- The InstallMigrateExProfile parameter is only necessary in cases where existing Outlook profiles are being used prior to the transition to Scalix (such as an Exchange environment).

Password/Security

Installing Scalix Connect across your user population ought to be undertaken in a secure manner, and should involve the following considerations:

- If random passwords were used during the generation of the Scalix mailboxes, then some distribution mechanism will obviously need to take place prior to user installation of Connect. One solution that you can activate during the mailbox provisioning process is to set each mailbox to the same password, but with a pre-expiration. During installation, all users would enter the same initial password with their username, then each would immediately be prompted to enter a new, replacement password.
- If the Scalix environment is configured to use an external authentication source (such as an LDAP directory or Active Directory), then the user would simply need to provide the existing password text to this external source.

Enhancing Method 1 Using Scripts

By scripting a batch file, you can add error checking and automation that will help streamline the end-user deployment process. A batch file takes advantage of the following:

- Running setup with the /s switch will not prompt the user for Next and Finish buttons, it enables a silent install.
- The install.ini can contain a username value (bolded in the example below), which would then be placed inside the Outlook profile for the user.

Here is a sample batch file that you can script for your use:

```
[Install Flags]

InstallMailServerName=scalix01.company.com

InstallLogfileUploadLocation=c:\\windows\\temp\\install.log

InstallDefaultProfileName=Scalix1

InstallWithServerStore=1

InstallAllowSavedPassword=0

InstallSetDefaultProfile=1

InstallLogFileLocation=c:\\windows\\temp\\log.txt
```

```
        InstallMigrateExProfile=0

        InstallUsername=John Smith
```

The example shows how to utilize the following logic:

- A required variable is passed to the script (%1), which equates to a preset Username value.
- Verification that the appropriate static drive mapping exists, and if not automatic mapping of such drives and directories.
- In addition, if local directories are used, a check and subsequent creation, if needed, of the local directories.
- Check of permissions to the local drive (validating the Administrative user or equivalent requirement).
- Dynamic creation of the install.ini, whereby the value passed to the batch file is echoed into the last line (InstallUsername=).
- Moving to the correct working directory or providing full paths.
- Error handling should any of the operations fail.
- Messages echoed back to the user upon exit, based on success or failure.

Method 2 – Automation through Network/Domain Login

You also can use network/domain login scripts or systems management tools (often triggered through network/domain login) that automatically deploy Scalix Connect and facilitate creation of the Scalix profile—with no user interaction. The network/domain login or systems management tools need to provide the ability to execute scripts and programs, and to trap the login name— potentially in an environment variable.

The Scalix tools include the following options:

- Running setup with the /s switch bypasses prompting the user with Next and Finish buttons, and instead, enables a silent install.
- Running “setup.exe /s” on a machine that has an existing Scalix Connect installation will simply not install anything, with no errors.
- The install.ini can contain a username value
- The ability to install Scalix Connect in advance on desktops without creation of an Outlook profile is possible, because Scalix Connect can exist on a Windows machine (observed under Add/Remove Programs) but be both dormant and unobtrusive.
- The installation permits you to create a Outlook profile separately, using the sxpro.exe executable.

The network/domain login scripts and/or the systems management tools offer varying capabilities, but in essence there are two basic options:

- Execute both Scalix Connect install and Outlook Profile generation together
- Execute the installation of Scalix Connect in advance, and then at a later date, execute the installation of the Outlook profile.

In the case where an organization wants to execute both Scalix Connect installation and Outlook profile creation within the same login session then the script logic would basically be the same as described in the previous section. The difference: a user simply logs into the domain, and the value required for the InstallUsername= parameter of the install.ini is extracted from a variable that stored the user's login name.

Deploying Scalix Connect without Outlook Profile Creation

In the case where an organization wants to deploy Scalix Connect first, and then create Outlook profiles for the users at a later time, the steps are quite simple. Because Scalix Connect installation program (setup.exe) can be run against an existing Windows machine that already has it installed with no error, just enable the executable to run upon domain login over a period of time (days or weeks). Naturally there would certainly be requirements to check whether the user is qualified to install Scalix Connect (e.g., is their system running Windows 2000 or a later OS).

The most important point: if no install.ini file exists in the current working directory where setup.exe is executed, then no Outlook profile will be created and no errors will occur. In this scenario the /s switch (silent install) should also be used.

Scalix Profile Creation

At some point, after each user's computer has had Scalix Connect installed, each computer will require creation of a Scalix profile. The utility sxpro.exe can be used to create the needed Scalix profile and it can be called in either of your network/domain login scripts or systems management tools.

The sxpro.exe utility must read an install.ini in order to create the profile, but considering the install.ini is plain text, it could again easily be created within the script facility that is launched upon domain login. Similar to what has been explained in prior sections, the InstallUsername= parameter would potentially be filled with the login name value that has been trapped in an environment variable at the beginning of the login session.

The necessary format for sxpro is:

```
sxpro –i c:\connect
```

In this example, c:\connect is the path to where the install.ini has been stored. Note that you do not need to specify the install.ini filename; it simply needs to exist in that directory.

Example: Batch File

The following batch file provides a good template that can easily be modified for any of the scenarios described previously for automating the deployment of Scalix Connect and creation of the Outlook Profile. In this example note that the creation of the Outlook profile occurs along with the Scalix Connect installation.

```
@ECHO Off

REM Modify the names of the Shares in the following two lines
REM Assume the username variable and the COMPUTERNAME variable
either come from the login
REM session, or are passed on the command line.
SET INSTALL_DIR=\\postoffice\Scalix
SET LOG_DIR=\\postoffice\ScalixLogs

IF NOT EXIST %INSTALL_DIR% GOTO CantFindInstallDir
IF NOT EXIST %LOG_DIR% GOTO CantFindLogDir
```

```
SET STARTTIME=%DATE% - %TIME%

IF NOT DEFINED TEMP GOTO NoTempDir
IF NOT EXIST %TEMP% GOTO CantFindTempDir
GOTO TempDirOK

:NoTempDir
SET NO_TEMP_VAR=TRUE
IF NOT EXIST \TEMP GOTO CantFindTempDir
SET TEMP=\TEMP
GOTO TempDirOK

:TempDirOK
SET CLIENT_LOG_FILE=%TEMP%\%COMPUTERNAME%_ScalixConnect_clientlog.txt
SET INSTALL_LOG_FILE=%TEMP%\%COMPUTERNAME%_ScalixConnect_InstallLog.txt
SET INI_FILE=%TEMP%\Install.ini

ECHO [Install Flags] > %INI_FILE%
ECHO InstallWithServerStore=1 >> %INI_FILE%
ECHO InstallAllowSavedPassword=0 >> %INI_FILE%
ECHO InstallSetDefaultProfile=1 >> %INI_FILE%
ECHO InstallMigrateExProfile=0 >> %INI_FILE%
ECHO InstallDefaultProfileName=Scalix1 >> %INI_FILE%
ECHO InstallMailServerName=sx1.company.com >> %INI_FILE%
ECHO InstallUsername=%username% >> %INI_FILE%
ECHO InstallLogfileUpLoadLocation=%CLIENT_LOG_FILE% >> %INI_FILE%
ECHO InstallLogFileLocation=%INSTALL_LOG_FILE% >> %INI_FILE%

CD %TEMP%
COPY %INSTALL_DIR%\setup.exe %TEMP%


%TEMP%\setup.exe /s

ERASE /q %TEMP%\setup.exe

ERASE /q %TEMP%\install.ini

ECHO  >> %CLIENT_LOG_FILE%
ECHO  >> %INSTALL_LOG_FILE%
ECHO Install Started at %STARTTIME% >> %CLIENT_LOG_FILE%
ECHO Install Started at %STARTTIME% >> %INSTALL_LOG_FILE%
SET ENDTIME=%DATE% - %TIME%
ECHO Install Ended at %ENDTIME% >> %CLIENT_LOG_FILE%
ECHO Install Ended at %ENDTIME% >> %INSTALL_LOG_FILE%

COPY /y %CLIENT_LOG_FILE% %LOG_DIR%
COPY /y %INSTALL_LOG_FILE% %LOG_DIR%
ERASE /q %CLIENT_LOG_FILE%
ERASE /q %INSTALL_LOG_FILE%

SET CLIENT_LOG_FILE=
SET INSTALL_LOG_FILE=
SET INI_FILE=
SET ENDTIME=

GOTO Exit


:CantFindInstallDir
:CantFindLogDir
:CantFindTempDir
GOTO Exit


:Exit
IF NOT DEFINED NO_TEMP_VAR GOTO ContinueExit
SET NO_TEMP_VAR=
SET TEMP=
GOTO ContinueExit

\:ContinueExit
SET INSTALL_DIR=
SET LOG_DIR=
SET STARTTIME=
```

Remotely Modifying a User Profile

You can modify your mail user's Outlook client profiles to add a Scalix profile. This can be done remotely, when they first log into the Scalix server. The process involves edits to a couple of files that are incorporated in your Scalix server.

Note that when a user first logs into Scalix, you can set up a dual-purpose process to install Connect in their Outlook, and create the needed Scalix profile.

For more information, review the Knowledgebase at Scalix support web site.

Setting Up Automatic Upgrades

You have the option to set up a server-side upgrade process that will automatically upgrade Scalix Connect plugins on client systems, when the clients log into your mail system. This process automatically detects log-ins, determines the version of Scalix Connect, then pauses the mail connection for the duration of the upgrade. The process results in a short interruption for the user, but the connection is reestablished afterwards and they can proceed to get their mail.

To set up such an automatic upgrade, follow this general process:

1. Modify the install.ini according to your preferences.
2. Create a client-accessible directory on the Scalix Server that includes the setup.exe file.

Note	All users must have read/execute rights to this folder.

3. Create a template message for omsndmsg.
4. Generate the user password list using ommigu –show and copy the output file to the directory containing omsndmsg.
5. Run omsndmsg.

All of these tasks and options are detailed in this section.

Overview

You have the option to upgrade Scalix Connect on more than one client systems at the same time, and with only a minimum of client user interaction. Several options exist, including extended capabilities and tools included with Scalix Connect that you can use when installing Scalix Connect on user systems, such as:

- New mailbox password distribution
- Outlook profile standardization
- Outlook profile migration
- Logging

The Scalix Connect setup.exe file initiates the installation of Scalix client software on user systems. Depending on the options you specify in the install.ini file, setup.exe can also launch the Outlook profile creation wizard.

The first time you install Scalix Connect on client systems, make sure the install.ini file is in the same shared network directory as the setup.exe file. The install.ini file contains parameters which can:

- Streamline the installation process
- Generate installation audits to determine the users systems have been successfully upgraded
- Standardize all profile settings across a group of users or an organization

Customizing Directory and File Permissions

Make sure that all Scalix Connect users have the required access to the shared network directory that contains the following Scalix Connect installation files:

- setup.exe
- install.ini

"Administrator" permissions are required on user systems for the initial installation of Scalix Connect. After the initial installation, you can upgrade Scalix Connect using any system permissions. See "Performing Automatic Upgrades" on page 42 for more information.

Configuring the install.ini File Parameters

The Scalix Connect install.ini file includes parameters you can configure to customize the automatic installation of Scalix Connect.

Parameters	Description
InstallMailServerName=	The name of the Scalix Server to which users connect.
InstallLogfileUpLoadLocation=	Enter the path and file name for the file that progressively accumulates all installation information on a user system.
	The file stores data generated from the file specified in InstallLogFileLocation= parameter.
InstallDefaultProfileName=	The name of the Outlook Profile that you want to create for all users.
	Leave this parameter empty to disable Profile creation.
InstallWithServerStore=	This parameter must be set to 1.
InstallAllowSavedPassword=	Set this parameter to 0 to prompt users to enter their password when they log in to their Scalix Server mailbox.
	Set this parameter to 1 to disable the password prompt.
InstallSetDefaultProfile=	Set this parameter to 1 to configure the Profile created by InstallDefault-ProfileName as the default Outlook Profile on user systems.
InstallLogFileLocation=	Enter the path and file name for the file that stores installation information on a user system.

Sample install.ini File

```
[Install Flags]

InstallMailServerName=server.domain

InstallLogfileUpLoadLocation=c:\\temp\\all_installs.log

InstallDefaultProfileName=Scalix Connect XYZ Corp

InstallWithServerStore=1

InstallAllowSavedPassword=1

InstallSetDefaultProfile=1
```

Customizing Password Capabilities in the ommigu Command

The ommigu Linux shell script is the mailbox provisioning and password management tool for Scalix Server. ommigu is in the /opt/scalix/bin directory. It provides a range of password-specific actions that you may want to employ.

Activating Random Password Generation

Add the –g option to ommigu, to generate a random password for an individual mailbox or for a group of mailboxes.

Note	You must notify each user about the new password for their mailbox, which can be done with the sxsendmsg command, detailed later in “Communicating with Scalix Connect Users” on page 40

Recording Specific Passwords for Mailboxes

Use the ommigu script with the –p option to specify a specific password for an individual mailbox or for a group of mailboxes.

Activating One-Time, Pre-Expired Passwords

You can enter one-time passwords that pre-expire at the first user login. The ommigu script –expire option directs Scalix Connect to prompt users to change their password the first time they log into their mailbox with their regular password.

Listing Mailbox Passwords in Use

The ommigu command provides a –show option that collects status information about individual user mailboxes and the associated passwords.

```
ommigu -show
```

The output from the –show option includes (separated by semi-colons):

- common name
- password
- SMTP address
- distinguished name
- state (status)
- date

Here is a sample of output (a single user) from ommigu -show:

```
Hugh Packard;IsiPAi55;imap.mail.org;<???>;active;07/13/2005
```

To get only the mailbox password of a user, enter the -n option, followed by the user’s common name wrapped in quotes (as shown below):

```
ommigu –n “Hugh Packard” –show
```

To save the output to a standalone file, add the “>*filename*” option:

```
ommigu –show >filename
```

The -userfile option t generates the common names of the users from which you want to generate data. For example,

```
ommigu --userfile /tmp/group1.txt --show>/tmp/list1.txt
```

You can then use the list1.txt output file to manage the dissemination of new Scalix Server mailbox passwords for the users in your organization.

Communicating with Scalix Connect Users

The omsndmsg application sends a message to user mailboxes without using mail transports. The message itself can contain text that streamlines the Scalix Connect automatic installation process by instructing users what (if any) action they need to take to ensure that the installation (or upgrade) process operates correctly.

You can add the following to an omsndmsg message:

- notification about a migration
- information about using the setup.exe file
- support contact information

omsndmsg can also read ommigu output files, which enables efficient and secure distribution of new mailbox passwords to users.

Here is an example on a omsndmsg message:

```
From: IT Team

To: John Stevens

Cc:

Subject: Important Notice - Your New Scalix Mailbox is Available

!!! Please Read !!!

You have now been migrated from Exchange to Scalix. Your previous Exchange mailbox is still available but you will not be able to send email from that account.

Please log into Scalix and use this password when requested: 4k$jU76

Use the following link to complete the installation of Scalix Connect:

“\\server\share\ScalixMAPI\install.html\”

Please contact the IT Support Team at extension 9-8179 if you have any problems or questions.

Thanks,

Information Technology Department
```

The omsndmsg command includes the following extensions:

Option	Enter the following:	Description
-a	<Exchange Administrator Profile Name>	The system on which omsndmsg operates must be a 32-bit version of the Windows operating system (Windows NT, 2000, XP) and must contain an Outlook Profile for an Administrator account. This level of access is required to access all mailboxes.
-p	<Administrator password>	The password for the Administrator Outlook Profile.

Option	Enter the following:	Description
-l	<Logfile path/name>	A log file is generated which contains omsndmsg information. A typical entry for this option is a filename in the local temporary directory.
-u	<UserPasswordlist path/name>	This is the path/file generated by ommigu.
		This text file must reside in a directory accessible by the Windows system running omsndmsg.
-t	<Message template path/name>	The message template included with Scalix must exist in a directory that is accessible by the Windows system running omsndmsg.
-e	<Administrators Return email address>	The address you enter for this option is included in the From: line of the message which is delivered to each user.
		Entering an invalid address for this option results in undeliverable messages (if the user replies to the message). If you do not enter an address, no address displays in From: line.

Typically, you will install Scalix Connect on groups of user systems (instead of individually or all systems at the same time). Therefore, omsndmsg will likely be used several times. You can create a batch file to simplify to use of this application.

The following is an example of a batch file that you could use if it is executed from the same directory that contains the files referenced in the command line (omsndmsg.exe, group1.txt, template.txt).

```
omsndmsg -a "adminprof" -p "pass" -l "log1.txt" -u "group1.txt" -t
"template.txt" –e “IT Team”
```

A Sample Message Template

The following illustration displays the contents of a customizable template text file included with Scalix for use in creating notification messages for users. Use the -t <Message template path/name> option to use the template.

Note the HTML delimiters in the various parts of the message.

```
<subject>Subject: Important Notice - Your New Scalix Mailbox is
Available</subject>

<body>

!!! Please Read !!!

You have now been migrated from Exchange to Scalix. Your previous
Exchange mailbox is still available but you will not be able to
send email from that account.

Please log into Scalix and use this password when requested: <pass-
word></password>

Use the following link to complete the installation of Scalix Con-
nect:
```

```
“\\server\share\ScalixMAPI\install.html\”

Please contact the IT Support Team at extension 9-8179 if you have
any problems or questions.

Thanks,

Information Technology Department

</body>
```

Performing Automatic Upgrades

Once Scalix Connect is installed on the PC’s of your “Premium” *Outlook* client users, periodic upgrades of Scalix Connect will be needed. These upgrades include new functionality and fixes that ensure the proper operation of Scalix Connect. You can configure your Scalix Server to perform automatic upgrades that work this way:

- The automatic upgrade process is configured and the resource files stored appropriately.
- A client “Premium” user connects to their Scalix mailbox.
- Scalix detects the connection, polls the version number of the Connect software. This involves a comparison of the version number of the mapi.cfg file on the server with the version number of the mapi.cfg file on the user system.
- If the version is not up-to-date, Scalix will pause the mailbox connection, and perform the upgrade (after confirming the user’s consent)
- Once the upgrade is complete, Scalix restores the mailbox connection and the user can check their mail.

The process involves a brief interruption, but is otherwise transparent to the user.

Creating the Required mapi.cfg File

When a Premium user logs into Scalix with Outlook for the first time, a mapi.cfg file is automatically copied to the client system—if one already exists on the server. This master server-side mapi.cfg file must be created manually using the values described in this section of the manual. (This file is not automatically created during installation.)

When a mapi.cfg file is copied to the client user’s computer, it is stored in this Scalix directory:

```
C:\Documents and Settings\user\Local Settings\Application
Data\Scalix\Scalix\MAPI\Profiles\profile_name\Scalix
```

A Sample mapi.cfg File Showing [Autoupgrade] Options

The following illustration shows the [AutoUpgrade] section of a typical server-side mapi.cfg file.

```
# MAPI Client configuration file

20

[AutoUpgrade]
```

```
SetupPath=\\server\\setuppath_location\\setup.exe (9.1 to 1.0x
upgrades)

HTTPSetupPath=http://address/directory/setup.exe (9.2 to 10.x
upgrades)

HTTPUpdateInstallMgr=1

HTTPUpgradeExemptList=username1;username2

MinimumScalixVersion=x.xx.xx.xx

ForwardInstallLogsTo=administrator@company.com

ForwardInstallLogsFrom=IT Team

ForwardInstallLogsSubject=Auto-upgrade Status

UseLocalTimeVSGMT=1

UpgradeIntervalTimeCheck=8
```

Automating Upgrades to Scalix Connect

To automatically upgrade all current Premium users to a new version of Scalix Connect:

1 Place the new versions of setup.exe and SXInstallMgr.exe on a shared network directory that is accessible to all client user computers that will require upgrades.

2 Write down the directory pathway for use in the mapi.cfg file.

See “Directory and File Permissions” on page 31 for more information about permissions.

3 Create the mapi.cfg file and store it in this directory:

```
~/nls/C/mapi.cfg
```

- Use the example file text shown previously as a template.

4 Make the following changes to the mapi.cfg file:

[version number]

(*Represented by “20” in the previous example.*) Be sure the version number noted in the file is greater than the version number of the mapi.cfg file on the user systems you want to upgrade.

For example, changing the value from **20** to **21** initiates the auto-upgrade process as all current client computers will have 20 in their mapi.cfg files.

SetupPath

Enter the pathway to the shared directory storing the setup.exe and SXInstallMgr.exe files. (Users must have read access to the files and their directories.)

MinimumScalixVersion

Enter the version of the sxmapi32.dll file that is part of the Scalix Connect release to which you want to upgrade. The Scalix Release Notes contain the version number.

UseLocalTimeVSGMT

Specify whether you want to use local time or Greenwich Mean Time (GMT) to auto-upgrade users. Enter **1** for local time, or **0** (zero) for GMT.

UpgradeIntervalTimeCheck

Enter a parameter to specify the (metric) time at which Scalix Server polls client systems to verify whether they are using the latest version of Scalix Connect. For example, enter 8 to poll for auto-upgrade status information at 8 am. Enter 22 to poll for information at 10 pm.

Entering value of 24 or greater causes Scalix Server to poll for auto-upgrade information in intervals (by seconds). For example, if you want client systems to poll clients systems every hour, enter 3600.

5 Save the changes to the mapi.cfg file.

Note: See the Scalix Administration Guide for more information about information to be included in a mapi.cfg file.

More Information about mapi.cfg [AutoUpgrade] Parameters

This section provides a reference guide (in table format) to all the Auto-upgrade options:

Table 6: MAPI Parameters and their Descriptions

Parameter	Description
n	The mapi.cfg version number that is used to determine whether auto-upgrades occur. This number is also used to determine whether mapi.cfg is downloaded to update other administrative settings. NOTE: If the version number of the mapi.cfg file on the user system is equal to or greater than the version number of the mapi.cfg file on the server, the system does not upgrade Scalix Connect on the user system with the latest version of the MAPI service provider and/or update the mapi.cfg file.
SetupPath	The path to the shared directory that contains the source Scalix Connect installation files. The SetupPath value must be a valid UNC path. Use this parameter if you are upgrading from 9.2 to 10.x
HTTPSetupPath	The path to the shared directory that contains the source Scalix Connect installation files. The HTTPSetupPath value must be a valid http:// address. Use this parameter if you are upgrading from 9.2 to 10.x.
HTTPUpdateInstallMgr	Set this parameter to 1 to install the SXInstallMgr.exe file on user systems. When set to 0, the SXInstallMgr.exe file is not installed on local systems.

Table 6: MAPI Parameters and their Descriptions

HTTPUpgradeExemptList	Allows you to specify individual users that you do not want to upgrade to the latest version of Scalix Connect.
MinimumScalixVersion	The version number of the Scalix Connect dynamic link libraries.
ForwardInstallLogsTo	Enter the administrator mailbox to which auto-upgrade results are sent. Enter off to disable error logging.
ForwardInstallLogsFrom	The text that displays in the From field of the Auto-install log message.
ForwardInstallLogsSubject	The Subject line of the e-mail that includes the auto-upgrade results.
UseLocalTimeVSGMT	Specify whether you want to use local time or Greenwich Mean Time (GMT) to auto-upgrade users. Enter 1 to use local time, or 0 to use GMT.
UpgradeIntervalTimeCheck	The (metric) time at which Scalix polls client systems to verify whether they are using the latest version of Scalix Connect. For example, enter 8 to poll for Auto-upgrade status information at 8 am. Enter 22 to poll for information at 10 pm.
	Note: Entering value of 24 or greater causes Scalix to poll for Auto-upgrade information in intervals (by seconds). For example, if you want to poll client systems every hour, enter 3600.

Activating the Auto-upgrade Logging

Auto-upgrade logging enables you to view a list of those users who have (or, by absence, have not) successfully upgraded their client systems to the latest version of Scalix Connect.

Status logs are saved to a temporary file location and then automatically sent from the user system to the mailbox previously specified in the mapi.cfg **ForwardInstallLogsTo** parameter.

You can also:

- Specify the text that displays in the From field of the auto-install log messages by entering a value in the ForwardInstallLogsFrom field
- Specify the Subject of the auto-install log messages by entering a text in the ForwardInstallLogsSubject field

See the *Scalix Administration Guide* for more information about configurable options of the mapi.cfg file.

Localizing Outlook

You can localize the MAPI Connector for any language that Outlook supports.

Tools

For this procedure, you need the following tools:

- Microsoft resource compiler Visual Studio 6.0, including the compiler RC.EXE and the linker LINK.EXE. For more on the compiler, see http://msdn.microsoft.com/library/default.asp?url=/library/en-us/tools/tools/resource_compiler.asp.

Procedures

To localize the MAPI Connector:

1. Use your preferred editing and translation tools to localize the English strings in the file SCLXRES\RESEN.RC and in all sub-files of the SCLXRES\EN subdirectory.

   Make sure you can save the file with the proper font files.

2. From the directory SCLXRES, run the resource compiler. The command is:

   ```
   rc /l<LCID Culture ID> /fo "<file name>\sclxres.res" /i
   "..\shared\include" /i "..\common" /d "THIS_LANG_<NAME OF LAN-
   GUAGE>" sclxres.rc
   ```

   Where:

   - **rc -** The compiler command
   - **<LCID Culture ID> -** Can be found at http://www.microsoft.co.ke/globaldev/nlsweb/default.mspx. This code number must be preceded by an “l” (a lowercase l, not an i) and must match the “THIS-LANG...” option
   - **l -** Specifies default language for compilation. For example, -l409 is equivalent to including the following statement at the top of the resource script file: LANGUAGE LANG_ENGLISH,SUBLANG_ENGLISH_US
   - **fo -** Renames the source file so that it comes out as a .res file
   - **i -** The "include" command so takes a directory as its variable
   - **d -** This variable changes according to the culture ID specified earlier

   Note that in this step, you are compiling the file SCLXRES.RC, which includes all subfiles and subfolders, and creates a new file named SCLXRES.RES to the same directory.

3. When the compiler completes, you see a new file named sclxres.res in the directory SCLXRES.

4. Using the resource compiler's LINK.EXE tool, link the .res file to produce a dll in the same directory that will be called SCLXRES.DLL. The command is:

   ```
   link /nologo /dll /pdb:none /machine:I386 /nodefaultlib /
   implib:"sclxres.lib" /NOENTRY sclxres.res
   ```

5. Copy the new file, SCLXRES.DLL to the MAPI Connector's installation directory (the default installation directory is C:\Program Files\Scalix\Connect).

Note	For more on the link command, see http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vccore98/html/_core_linker_reference.asp.

Troubleshooting Scalix Connect

Using the Scalix Properties dialog box and the system tray, you can troubleshoot problems with Scalix Connect. The Support tab of the Scalix Properties dialog box provides different logging settings to record events and their details. In addition, the system tray notifies you if there are connection problems with the server.

Changing Scalix Connect Logging Settings

> **Alert** This tab is for troubleshooting purposes only and all changes made here are applied globally.

To change logging levels for troubleshooting purposes:

1. In the Outlook menu, select **Tools** and then **Scalix Properties**.
2. On the Scalix Properties dialog box, select the **Support** tab.
3. Use the radio buttons to set logging to the desired level.
4. When finished, click **OK**.

Using the System Tray to Detect Problems with Scalix Connect

Scalix Connect displays the status of its connection and cache in the system tray with an icon that changes color and design. The icon has six states:

Table 7: System Tray Icons and Their Definitions

Icon	Description
✓	In a suspended state or offline. Normally encountered at startup
↘	SmartCache is starting up, synchronizing, creating an item, or moving an item
✓	SmartCache is in a stable and ready state
!	SmartCache is not synchronized or is offline
✕	SmartCache has encountered a minor synchronization error
✖	SmartCache has encountered a serious error

To troubleshoot cache problems via the system tray:

1 In the system tray, look for the SmartCache status icon.

2 Double click the icon to get a log for the cache status. That log shows what phase (Phase 1, 2, or 3) or folders are in process of caching. The phases are:

- Phase 1 is ... signing on
- Phase 2 is the mode and detail level of caching
- Phase 3 is the caching of folders

Uninstalling Scalix Connect from a User's PC

If you need to remove the Scalix Connect software from a client PC, you have two options:

- Use the Microsoft Windows *Add/Remove Programs* control panel
- Run the Scalix Installer wizard, which includes an “uninstallation” option

To use the Scalix Intaller to remove Scalix Connect:

1 Insert the Scalix Installation CD into the CD ROM drive, or go to the network directory that contains Scalix installation files.

2 Double-click the setup.exe icon.

3 When the Repair/Remove window displays, select **Remove** and click **Next**.

4 When the process is complete, click **Finish**.

Alert

Do not remove Scalix Connect by deleting the component files. Many of these files (especially .dll files) might be used by other applications.

Deploying Scalix Web Access

This chapter covers deployment of the Scalix Web Access client for use with the Scalix Server.

Contents

This chapter includes the following information:

Features of Scalix Web Access

Web-based email provides the advantage of anywhere, anytime access through a browser. But the current generation of web email clients are used infrequently because of slow performance and limited functionality that significantly affect productivity. Scalix Web Access is different. A high-performance web email and calendaring client, Scalix Web Access represents a significant leap forward for web-based email access.

Scalix Web Access brings popular and familiar desktop application features including a three-pane user interface, pull-down menus, tool tips and drag-and-drop capabilities to the browser for web email access. Users can drag-and-drop messages, meeting appointments and contacts in exactly the same way they can in their desktop mail client today. Users familiar with Microsoft Outlook will be productive with Scalix Web Access without training.

High performance is enabled by Scalix Web Access' unique caching architecture. By performing most client activities including composing, reading, deleting, filing and sorting messages from an in-memory cache, Scalix Web Access reduces expensive server-side processing and significant network traffic common with page-oriented web email clients. The result is a highly responsive web email and calendaring experience, even when using low bandwidth network connections.

Scalix Web Access offers rich interoperability with Microsoft Outlook and Novell (Ximian) Evolution, enabling users to work interchangeably between the two interfaces for a fully synchronized view of their mail, calendar and contact data. Scalix Web Access runs on a broad range of desktop platforms, including Windows, Linux, UNIX and Mac, giving customers the flexibility to use Internet Explorer or the Firebox browser to access email, calendar and contacts.

Scalix Web Access utilizes the latest standards-based technologies, including DHTML, XML, and SOAP, to achieve the performance, look-and-feel and functionality of a modern desktop application. No Java or browser plug-ins are required to provide snappy performance typically only associated with desktop applications.

System Requirements for SWA

Scalix is compatible with the following browsers and email clients. For more on the system requirements for each client, see the *Scalix Client Deployment Guide.*

Components	Requirements
Approved Web client software (Scalix Web Access client browsers)	• Internet Explorer 5.5, 6.0 with SP3* and 7.0 • Mozilla 1.7 or higher • Firefox 1.0.7 and later *We do not currently support IE 6.0 in combination with Windows 98. In addition, we do not support IE 6.0.28 (SP1) in combination with NT4.0 server (SP6).
Approved e-mail client software	• Microsoft Outlook versions 2000, XP, and 2003 and later (versions 9, 10, 11) • Novell Evolution, versions 2.4.2 and later
Approved Windows OS versions (for client workstations)	• Windows 2000 • Windows XP (All earlier versions of Windows are not supported, irrespective of which version of Outlook you have installed.)

Installing SWA

Scalix Web Access installs with the product. No additional installation procedures are required.

Configuring SWA

You can configure many different SWA settings by modifying the file, swa.properties. A few of the more commonly-used settings include:

- Enabling or disabling major SWA components such as the calendar, the rules wizard, out-of-office auto-replies and more
- Enable or disable smaller SWA features such as read-receipt acknowledgements, auto spell check, the mini calendar frame, auto refresh and more
- Change the default mapping program used from Google to Yahoo!, Mapquest or others
- Change the email domain, IMAP server and SMTP server
- Change the size limit on attachments or the maximum number of search results
- Change calendar properties such as the date fromat and default calendar view, the days of the work week, the start and end times of the work day, the range for upcoming appointments and more
- Change the sensitivity prompts or the level at which emails are blocked

To change settings in the SWA properties file:

1. Go to swa.properties, which is located in WEB-INF.

   ```
   cd  ~/tomcat/webapps/webmail/WEB-INF
   ```

2. Using your preferred text editor, change or comment out the property as desired.
3. Restart Tomcat.

   ```
   ~/tomcat/bin/shutdown.sh
   ```

   ```
   ~/tomcat/bin/startup.sh
   ```

Disabling the Change Password Option in the Rules Wizard

To disable the "Change Password" option in the Scalix Rules Wizard (a feature of Scalix Web Access), a modification of a Scalix server file must be made. First, you have to determine the language profile currently in use.

To disable the password option:

1. Run the following command:

   ```
   omwebconf -ql
   ```

2. The resulting output is:

   ```
   "C", "AMERICAN" or "ENGLISH".
   ```

3. Open this file:

   ```
   ~/omhtml/{RW-profile}/page/rules-lffunc.html
   ```

4. Replace the "profile" placeholder in {RW-profile} with the language profile from omwebconf.
5. After opening this file, find and change this line:

   ```
   var fEnableChangePassword = true
   ```

6 The final edited version should match this example:

```
var fEnableChangePassword = false
```

Localizing SWA

Localization involves translating two xml files, putting them in the appropriate location, and then restarting Tomcat.

Note	Unless your localization file is part of a Scalix distribution, you MUST do a backup before the next upgrade. If you do not, your file will be lost.

To localize SWA for use in other countries:

1 Translate the files strings_en.xml and strings_en_US.xml to create the files strings_xx.xml and strings_xx_XX.xml. Those strings are located at:

```
~/tomcat/webapps/webmail/WEB-INF/data
```

For example: For Nederlands Dutch, create files named string_nl.xml and strings_nl_NL.xml.

2 Stop Tomcat by running the shutdown script found at:

```
 /etc/init.d/tomcat stop
```

3 If Tomcat has not already unpacked it, uncompress the war file.

4 Put the files strings_xx.xml and strings_xx_XX.xml in the WEB-INF/data directory.

5 Delete the Tomcat cache if needed by deleting the contents of the following directory.

```
~/tomcat/work/Catalina/ directory
```

6 Restart Tomcat.

```
~/tomcat/bin/shutdown.sh
```

```
~/tomcat/bin/startup.sh
```

Customizing the SWA Login Page

When Scalix Web Access users access their mailboxes by means of a web browser, it initially asks (via a "login" page) that the user log in with a user name and password. This function is supplied by scripting in Scalix, but you can replace the "login page" that appears by default with one of your own, using a different layout and corporate-image graphics, if you choose. Additionally, if you want to modify the status messages or instructions, you can do so. This package will help you to complete this task.

About the Login Process

When a user starts their browser and types the Scalix SWA URL, the following sequence of pages appears:

The first SWA login page appears, displaying a simple one-cell table with the Scalix logo, above a "Loading, please wait" message.

A second login page replaces the first, displaying the same table with a Scalix logo, a "Welcome" message and a set of login fields, including prompts and a "Login" button.

After user information is submitted to the server, a third page with the same table and Scalix logo appears, with the "Logging in, please wait" status message.

A fourth page appears when login is successful, which is quickly obscured by a new SWA window.

If the password is wrong, or out of date, the appropriate login page with relevant fields will be interleaved in this sequence.

As you will see, this series is all actually a single page running a sequence of scripts, some of which require user input.

What you Can Customize

You can customize:

- The Scalix logo can be replaced with a corporate logo or image of your choice.
- The status messages, instructional texts and other messages can be edited.

How to Create a Custom Login Page

Building and utilizing a custom login page requires that you create and place the login file (.html format) in a particular Scalix server directory. This file will be detected by Scalix, and will be used in place of the Java-originated login page.

To create a customized web page for user login, you can modify a template file.

To create the customized Web page:

1. Open either of the included HTML/XHTML files that were included in this package. (Links to the files available below.)
2. Save the file as “index.html” in this Scalix directory:

   `~/tomcat/webapps/webmail`

3. Do not edit or remove the relevant Scalix javascripts in the file header (including browser detection).
4. Verify the presence of an "onload" extension in the <body> tag.
5. Look first for the basic master one-celled table that "centers" the placeholder logo/graphic and the form elements underneath. Verify the presence of the "loadingMessage" code (including scripting) below the graphic image.
6. Insert the coding for the following "loginForm" elements, where you want these features to appear:

   `Username`

   `Password`

   `Submit button`

7 Review the coding for other login page functions, including "waitLogin", "expPwdForm", "loginSuccessful" -- all of which contain message texts that can be edited to your preference.

8 Store a copy of your corporate logo/image (in GIF or JPG format) in the same /webmail directory, then link the placeholder image in your file to this image.

9 In the file ~/tomcat/webapps/webmail/WEB-INF/web.xml, look for the section marked <welcome-file-list>.

10 To use index.html or index.htm as the default page, change it to read:

```
<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>index.jsp</welcome-file>

</welcome-file-list>
```

11 Restart Tomcat.

```
~/tomcat/bin/shutdown.sh

~/tomcat/bin/startup.sh
```

12 Start your browser and attempt to log into SWA, to test this page. Your work should appear in place of the standard Scalix login sequence, and you should be able to successfully log into your mailbox.

TIP 1

To view a sample XHTML file ("default.htm") containing the table frame, a graphics placeholder, texts and layout, that can be adapted to your uses, click LINK.

To view a sample XHTML file ("index.htm") that provides a shorter, more efficient login sequence, click LINK.

TIP 2

If you upgrade or re-install SWA for any reason, the modifications you make to index.html and web.xml will not be retained.

So, prior to an update of any sort, please take a copy of those files and make the changes again once the update has completed successfully.

Improving SWA Performance

You can improve SWA performance by putting a second SWA server outside the firewall in a DMZ. For more on this procedure, see the *Scalix Installation Guide.*

Upgrading SWA

For instructions on upgrading Scalix Web Access, see the *Scalix Installation Guide*.

Blocking Images in SWA

If desired, you can block images from displaying in incoming emails unless they are explicitly allowed. This is useful to prevent privacy and security issues with remote image requests. It can prevent users from unwittingly sending out image requests to spam servers.

To block remote images by default:

1. On the server on which the SWA module is installed, go to the file /etc/opt/scalix/webmail/swa.properties.

   ```
   cd /etc/opt/scalix/webmail/swa.properties
   ```

2. Look for the following option.

   ```
   swa.user.blockRemoteImages
   ```

3. Change the value on this option from FALSE to TRUE.
4. Restart Tomcat.

   ```
   ~/tomcat/bin/shutdown.sh
   ~/tomcat/bin/startup.sh
   ```

Some SWA Behaviors to Note

A few behaviors to note about SWA:

- When composing Scalix Web Access messages in Mozilla or Firefox browsers, URLs and email addresses do not get underlined. To correct this, you can mark text as hyperlinks by using Insert > Hyperlink from the Scalix Web Access menu, or by selecting the Create Hyperlink button.
- Popup-blocking software installed on your machine might prevent SWA from starting up. To correct this, you need to either disable pop-up blocking or add the Scalix Web Access URL to the list of allowed sites.
- You lose Rich Text (RTF) formatting if you edit the Notes field of an Appointment or Contact in Scalix Web Access. To correct this, If you need to edit this field and want to maintain RTF formatting, use Outlook to do the editing.

Integrating SWA with Portals

If you need to put SWA behind a portal such as a corporate Intranet, you can do so while still maintaining single sign on capability. That means users can provide their name and password through the portal login, then access SWA and their mailbox from inside the portal without having to sign in again.

There are two ways to do this:

- The mailbox link embedded in the portal calls a POST request
- The mailbox link embedded in the portal calls a URL that includes the user’s name and password information. This method is less desirable because it exposes the user’s password in the brower’s address bar.

Using a POST Request

In the POST request scenario, the link to the SWA mailbox embedded on the portal page calls a URL that passes along the user's login information via a POST request. The user logs in to Scalix via the URL in the link, but also via an HTML form using the POST method. In other words, a simple form such as the one below enables the user to log in to SWA when clicking the mailbox link:

```
<html>
   <head>
   </head>
   <body>
      <form method="post" action="http://<your_scalix_mailserver_FQDN>/webmail/index.jsp">
         username: <input type="text" name="username"><br>
         password: <input type="password" name="password"><br>
         <input type="submit" value="go to SWA" >
      </form>
   </body>
</html>
```

Where:

- The username is the name of the mailbox user
- The password is the password of the mailbox user
- The action attribute must point to "/webmail/index.jsp" or the POST request is dropped
- The method of the form must be a "post".

Embedding the Username and Password in the URL

In the second scenario, the link to the SWA mailbox that is embedded on the portal page calls a URL that passes along the user's login information. This method is less desirable because it briefly exposes the user's password in the browser address bar. There are two potential URLs:

- A regular login screen with the username already filled in and the focus in the password field: http://<your_scalix_mailserver_FQDN>/webmail?username=Boris
- The user sees only the loading screen and SWA opens in a new window. The password is clearly visible in the URL, but is protected by HTTPS: https://mailserver.com/webmail?username=Boris&password=scalix

Uninstalling SWA

For instructions on uninstalling Scalix Web Access, see the *Scalix Installation Guide*.

Deploying IMAP Clients

This chapter covers special deployment issues affecting IMAP clients such as Outlook (used in IMAP mode), Outlook Express, Thunderbird or Eudora. Because deployment of IMAP clients with the Scalix server is straightforward, we do not offer specific installation or deployment instructions.

Contents

This chapter includes the following information:

Localizing for Japanese Language Characters

When using Japanese characters with Scalix, you may want to change the preferred character standard for representing rich multi-byte text in messages from UTF-8 to ISO-2022-JP. This is because more Japanese Internet mail clients and systems understand ISO-2022-JP than understand UTF-8.

- For IMAP and Pop3 clients, the default values should be "US-ASCII,ISO-8859-1,UTF-8"
- For outgoing Internet gateway, the default values should be "US-ASCII,ISO-8859-1,UTF-8"

To change the default character standard:

1 Go to the file general.cfg, which is located in

```
~/sys/general.cfg.
```

2 Add the following lines (or modify them if they are already present):

- For IMAP and Pop3 clients:

  BRW_MIME_TEXTFILE_CHARSETS="US-ASCII,ISO-8859-1,ISO-2022-JP,UTF-8"

- For outgoing Internet gateway:

  UXO_MIME_TEXTFILE_CHARSETS="US-ASCII,ISO-8859-1,ISO-2022-JP,UTF-8"

Note: Different settings may be relevant for the domains of different recipient organizations. Therefore, you may want to add different values for UXO_MIME_TEXTFILE_CHARSETS in per-domain configuration files located in ~/

scalix/sys/domain.cfg/<domainname>. You may need to create these files if they do not exist already.

3 Restart the relevant Scalix services

```
omoff -d0 unix mime imap

omon unix mime imap
```

Note that once a message has been cached for use by IMAP, SWA and Pop3 clients, it is not typically re-constructed to reflect changes to BRW_MIME_TEXTFILE_CHARSETS unless the cache is manually refreshed.

Manually refreshing the cache requires two steps: 1) Changing the settings on MSL_SUBVERSION in general.cfg and 2) Running the command omtidyallu (or omtidyu on specific users).

To manually refresh the cache:

1 Go to the file general.cfg, which is located in

```
~/sys/general.cfg.
```

2 Locate the following setting and change it to a small number such as 1 so that when checking its cache, the mime construction routines can see that the version has changed, and so regenerate.

```
MSL_SUBVERSION=1
```

3 Run the command omtidyallu with the -M flag (mime generation) to traverse each user’s message store and prime the cache.

SSL with IMAP

Scalix does not have native Transport Layer Security, but Secure Sockets Layer (SSL) support is handled through the use of stunnel. The appropriate cofiguration entries for the file stunnel.conf are:

To configure an IMAP client for SSL and Scalix:

1 Go to the file stunnel.conf.

2 Change to following entries:

- [imaps]
  - accept = 993
  - connect = 143
- [ssmtp]
  - accept = 465
  - connect = 25

Deploying Scalix Connect for Novell Evolution

This chapter covers deployment of the Scalix connector for Novell Evolution, which enables use of the Evolution client with the Scalix Server.

Contents

This chapter includes the following information:

Overview

Scalix Connect for Novell Evolution provides transparent, full function support for email on Linux desktops. It enables connectivity between Evolution and the Scalix Server, and supports IMAP and iCal for compatibility with the full range of functionality offered by Evolution.

You can use the connector with Novell Evolution 2.4 (or later), an email client bundled with Gnome 2.12. We recommend you completely update Gnome to 2.12, which automatically installs Evolution 2.4 before beginning installation of the Scalix Connect RPM.

Alert	The Scalix Connect plugin for Evolution works only with Scalix Server 10.0 or higher. Contact your mail system administrator for confirmation if your Scalix Connect does not work.

.

Note	Support for the Scalix Connect for Evolution binaries is provided by the Scalix Community Forum as well as Scalix Technical Support (via incident support or premium support). In addition, you can download the latest source from the GNOME CVS.

.

Note	Group scheduling features and public folders in Scalix Connect for Evolution are available only to Scalix "premium" users, not "standard" users.

Features of Scalix Connect for Evolution

Table 1: Features of Scalix Connect for Evolution

Component	Features
Mail	• Standard IMAP functionality • Public folders • Outlook-style handling of deleted items • Search folders • Encryption and signing of messages (PGP) • Message receipts
Calendar	• Standard calendaring and scheduling functionality • Reading and writing of freebusy information • Create new calendar folders • Access to calendars in public folders
Contacts	• View, create, modify and delete contacts and PDLs • Lookup of system entries through LDAP (can be included in PDLs) • Create new contact folders • Access to contacts in public folders
Other	• Wizard-driven account setup • Offline mode for mail • Change password • Easy access to Scalix Rules Wizard • Localized for English and German

System Requirements

Any system already configured to work with Novell Evolution is assumed to meet the basic requirements for the Scalix connector. There are, however, some minimum system requirements for its use.

Scalix Connect for Novell Evolution requires:

Components	Requirements
Client Software	Novell Evolution 2.4 or later *Evolution is an email client included with Gnome 2.12. We recommend that you (1) completely update Gnome to 2.12, which automatically installs Evolution 2.4.
Operating System	Fedora Core 4 (or later) RedHat Enterprise Linux 4 (or later) *You can download and build the connector on another operating system, but it would be unsupported.

Approved Web Browsers and Email Clients

Components	Requirements
Approved web client software (Scalix Web Access client browsers)	• Internet Explorer 5.5 or 6.0 • Mozilla 1.7 or higher • Firefox 1.0 and later
Approved e-mail client software	• Microsoft Outlook versions 2000, XP, and 2003 and later (versions 9, 10, 11) • Novell Evolution, versions 2.4.2 and later
Approved Windows OS versions (for client workstations)	• Windows 2000 • Windows XP (All earlier versions of Windows are not supported, irrespective of which version of Outlook you have installed.)

Pre-Installation Steps

Before installation, update Gnome to 2.12, which automatically installs Evolution 2.4.

If your computer is running Fedora Core:

1. Completely update the Fedora installation with a yum update.
2. Download and install the NRPMS package for your Fedora version from this URL:

   `www.nrpms.net/Docs/Yum`

3. Complete a second yum update.
4. You can now install the connector.

Alert

The Scalix Connect plugin for Evolution works only with Scalix Server 10.0 or higher. Contact your mail system administrator for confirmation if your Scalix Connect does not work.

Installing Scalix Connect for Novell Evolution

Download and Install the RPM

These steps assume you already have the correct version of Evolution installed. If you do not, see the steps above for upgrade instructions.

To download and install:

1. Get the Scalix Connect for Evolution RPM.
2. As super-user or via sudo, install the RPM:

   rpm -ivh evolution-scalix-10.0.0.x-1.i386.rpm
3. If you have an older version of the connector installed, then upgrade as follows:

   rpm -Uvh evolution-scalix-10.0.0.x-1.i386.rpm

Setting up an Account

You need to create an account profile--unless you were using the connector before and have just upgraded it to a newer version. This wizard is useful if you have not previously configured mail accounts in Evolution. If you already have one or more accounts in Evolution, choose **Edit -> Preferences** and add a new account with the **Mail Accounts** section.

Uninstalling Scalix Connect for Evolution

Uninstalling Scalix Connect for Evolution is a two-step process.

To uninstall:

1. Remove the account profile by choosing **Edit > Preferences > Mail Accounts**
2. Remove the connector by opening a terminal window and running this command:

```
rpm -ev evolution-scalix
```

Using Scalix Connect for Evolution

Once Scalix Connect is installed and running, your Novell Evolution client works as it would with any other email server. For more information on how to use the Evolution client, see the Novell user documentation.

There are four special features added by the Scalix Connector, though. They include:

- Scalix Rules Wizard
- Change Password
- Deleting Messages
- About Scalix

Scalix Rules Wizard

The Scalix Rules Wizard (SRW) lets you set up and monitor any number of server-side rules for message filtering. A browser-based application, SRW can be accessed through the Edit menu in Evolution. When SRW opens in a new browser window, your username will already be filled in.

You can change the location of SRW under the Defaults tab in your account settings (**Edit->Preferences->Mail Accounts**) in case the default location is wrong.

Evolution must be in the mail module before this link is usable. If you have no account selected in the tree view, a list of Scalix profiles will be displayed, for you to choose from.

Change Password

To change your Scalix account password in Evolution:

1. Select the mail module/account.
2. In the dialog box that appears, change your account password.

Deleting Messages

The procedure for deleting a message from a Scalix mailbox with Evolution is different from other email clients. This process resembles that of Microsoft Outlook, in which a message is first moved to the “Deleted Items” folder, then erased by use of a Empty Deleted Items command.

You must be in the mail module to access this feature. If no account is selected in the tree view, you are presented with a list of Scalix profiles from which to choose.

About Scalix

To find information about Scalix Connect for Evolution:

1. In the toolbar, click **Help**
2. Click **About Scalix**.

Deploying Scalix Mobile Web Client

This chapter covers deloyment of the Scalix Mobile Web Client.

Contents

This chapter includes the following information:

Overview

Scalix Mobile Web Client is a lightweight web interface that runs in any browser and provides basic mailbox access from cell phones, PDAs and hotel set-top boxes. It lets mobile users perform simple operations such as reading, replying to, and forwarding messages, composing new messages, and searching their Inbox. Mobile Web Client is available for any Scalix user type on any of our product editions.

Using Scalix Mobile Web Client

To use the Scalix Mobile Web Client:

1 Point your PDA's browser to the following URL:

```
http://<your_scalix_mailserver_FQDN>/m
```

2 Use the same username and password as for the regular user account.

Glossary

Some terms and acronyms in this manual may be unfamiliar to users. Here are some terms and definitions that are specific to the Scalix product and the Linux platform.

Table 1: Terms and Their Definitions

Term	Definition
Address Directories	In Scalix terminology, the address directories are databases that clients use to look up names and addresses. Scalix directories can hold addresses of both Scalix and non-Scalix users, and other information that an administrator can configure such as job titles and phone numbers. Directories can be searched by any number of attributes.
Management Console or SAC	The Scalix Management Console (SAC) is a browser-based application that enables most day-to-day system administration tasks on a Scalix messaging system through an easy-to-use GUI. It is a separate component of Scalix that users can access with any approved browser on either Microsoft Windows or Linux workstations. SAC provides efficient access to a wide range of Scalix server options, including user account management, starting and stopping server services, administering queues, public distribution list or group management, and changing low-level server configuration settings. It also provides system monitoring to assess the status of processes and resources.
ADUC	(Active Directory Users and Computers) ...
Authentication Identifier	The Scalix system has several ways of identifying users for different purposes: Display names, personal names, authentication IDs and Internet addresses. The display name (also known as a “common name”) is used in Outlook and other clients as the “displayed” address. It can serve authentication purposes and determines the sort order in the Outlook address book. Authentication IDs support the concept of a separate login name and allow for integration with external authentication systems that may have their own naming rules. Internet addresses are SMTP addresses of the form name@domain. Personal names are used for internal addressing of email and are sometimes referred to as “X.400 addresses,” “OpenMail addresses” or “ORN (originator Recipient name.”
Bulletin Board	In Scalix terminology, a bulletin board is a set of public folders where members can share files, ideas, documents and more. They are a shared area in the Scalix message store.
Clam AV	An open source freeware program that protects against viruses.

Table 1: Terms and Their Definitions

Term	Definition
Community Edition	The free, single-server, unlimited-use version of the Scalix product. Does not include advanced groupware and collaboration functionality.
DDR	
Display Names vs User Names vs Personal Names vs authentication ID vs Internet address	The Scalix system has several ways of identifying users for different purposes: Display names, personal names, authentication IDs and Internet addresses. The display name (also known as a “common name”) is used in Outlook and other clients as the “displayed” address. It can be used for authentication purposes and determines the sort order in the Outlook address book. Authentication IDs support the concept of a separate login name and allow for integration with external authentication systems that may have their own naming rules. Internet addresses are SMTP addresses of the form name@domain. Personal names are used for internal addressing of email and are sometimes referred to as “X.400 addresses,” “OpenMail addresses” or “ORN (originator Recipient name.”
Enterprise Edition	The company’s flagship product, which includes multi-server support, unlimited number of Standard users, any number of Premium users, the full complement of Scalix advanced capabilities, and a wide variety of technical support options.
Gateway	Gateways are a way of passing messages out of the Scalix network to different mail environments. The gateway converts outgoing messages from a Scalix format to a format that external services can use to do send processes, and later to a format that target environments can receive such as an SMTP address. Scalix comes with a standard SMTP gateway that converts Scalix-formatted messages to SMTP and vice-versa. This SMTP gateway is called the Unix Mail Gateway or Internet Mail Gateway.
Groups and PDLs	In Scalix terminology, the terms “group” and “PDL” are used interchangeably to mean a group of people organized into a mailing list. PDLs can contain both local and remote users, and can contain nested PDLs.
IMAP	(Internet Message Access Protocol) A standard interface between an e-mail client program and the mail server. In Scalix, the iMAP4 server enables a client to: Access, list, read, and delete items from inboxes, filing cabinets and public folders; read parts of a message without downloading the entire thing, keep a record of which messages have been read, and update messages on the server from a client. IMAP extesnions also provide for calendaring and contact management.
Internet Domains vs mailnodes	Mailnodes have no direct relationship to Internet domains. However, you can set up rules so that when a user is created on a mailnode, Internet address generation kicks in and creates an Internet address for the user. You can map multiple mailnodes to the same Internet domain name.
LDAP	(Lightweight Directory Access Protocol) A protocol used to access a directory listing. In Scalix, the LDAP server is a daemon process based on a client/server model that provides an interface to enable LDAP clients to store and retreive data from a Scalix directory without any information about the operation of Scalix. It provides LDAP clients access to shared Scalix directories that do not have an associated password.
LVM	(Logical Volume Manager) Used for backing up Scalix directories.

Table 1: Terms and Their Definitions

Term	Definition
Mail Nodes	A logical structure used to organize users into administrative groupings. For example, some companies organize their email users by work group whereas others break their users down by employment status. Each Scalix server is associated with a single mail node created during installation. After installation, you can use the Management Console to create additional mail nodes on a server, including customizing any new mail nodes with a specific Internet address or domain name.
MAPI	(Mail API) A programming interface from Microsoft that enables a client application to send to and receive mail from Exchange Server or a Microsoft Mail (MS Mail) messaging system. Microsoft applications such as Outlook, the Exchange client and Microsoft Schedule use MAPI.
Message Store	The message store is a collection of flat Linux files held in file system directories on the Scalix server. It holds new messages received as well as messages in transit. For clients that use the message store (server-based clients), It also holds old messages that are files for reference in folders, copies of outgoing messages, draft messages, private distribution lists, personal information such as calendaring, tasks, bulletin boards, public folders and more.
Mx Records	Mail exchanger records inside DNS servers. These decide which server is responsible for dealing with mail or domain DNS actions.
OpenMail	The original technology, licensed from Hewlett Packard, upon which the Scalix system is based.
O/R or Originator/Recipient Address	An attribute list that distinguishes one user, or distribution list, from another and defines the user's point of access to the message handling system or the distribution list's location.
PAM	(Pluggable Authentication Modules). A standard library in Linux that connects applications that require authentication with shared library modules interfacing with authentication mechanisms.
PDL	In Scalix terminology, the terms “group” and “PDL” are used interchangeably to mean a group of people organized into a mailing list. PDLs can contain both local and remove users, and can contain nested PDLs.
Personal Name	The Scalix system has several ways of identifying users for different purposes: Display names, personal names, authentication IDs and Internet addresses. The display name (also known as a “common name”) is used in Outlook and other clients as the “displayed” address. It can be used for authentication purposes and determines the sort order in the Outlook address book. Authentication IDs support the concept of a separate login name and allow for integration with external authentication systems that may have their own naming rules. Internet addresses are SMTP addresses of the form name@domain. Personal names are used for internal addressing of email and are sometimes referred to as “X.400 addresses,” “OpenMail addresses” or “ORN (originator Recipient name.”
POP	(Post Office Protocol) A standard interface between an e-mail client program and the mail server. The Scalix POP3 server enables clients to list, read and delte items from the inbox area of the Scalix message store. The Scalix POP3 server does not provide access to any other areas of the message store such as public folders.

Table 1: Terms and Their Definitions

Term	Definition
Premium Users	Scalix has two levels of access and usage: Premium and Standard. Premium users have access to the full benefits and functionality of the Scalix email and calendaring system. Standard users gain access to a subset of Scalix functionality including email, personal calendar and contacts through Scalix Web Access and Novell Evolution as well as email access using POP/IMAP clients.
Realm	
SAC	The Scalix Management Console (SAC) is a browser-based application that enables most day-to-day system administration tasks on a Scalix messaging system through an easy-to-use GUI. It is a separate component of Scalix that users can access with any approved browser on either Microsoft Windows or Linux workstations. SAC provides efficient access to a wide range of Scalix server options, including user account management, starting and stopping server services, administering queues, public distribution list or group management, and changing low-level server configuration settings. It also provides system monitoring to assess the status of processes and resources.
Scalix Connect	A MAPI application that enables the use of the Outlook client interface and all of its functionality.
Sendmail	An SMTP-based message transfer agent (MTA) that runs under Unix and Linux. It is the mail transfer process used inside the Scalix system.
SSL	
Small Business Edition	A version of the Scalix system that targets organizations getting started with a commercial version of Scalix that do not have the higher end requirements of Enterprise Edition. It is functionally equivalent to Enterprise Edition except that it allows only single-server installations
SmartHost	
Spam Assasin	An open source freeware program that filters spam.
Standard Users	Scalix has two levels of access and usage: Premium and Standard. Premium users have access to the full benefits and functionality of the Scalix email and calendaring system. Standard users gain access to a subset of Scalix functionality including email, personal calendar and contacts through Scalix Web Access and Novell Evolution as well as email access using POP/IMAP clients.
SWA	Scalix Web Access, the browser-based email, calendar, contacts and public folders client that comes with any Scalix installation.
Transports	Transports are services that Scalix uses to pass Scalix format messages to other Scalix services. Scalix uses Sendmail and SMTP formatted messages to send messages between servers in the Scalix network, but other connections can be written. The transport service on the Scalix server is called the Sendmail Interface.
UAL	(User Access Layer) A proprietary Scalix protocol that enables communication between clients and the Scalix server.
WAP	(Wireless Application Protocol) A standard for providing cellular phones, pagers and other handheld devices with secure access to e-mail and text-based Web pages.