Ensim® Pro 4.0.1 for Windows® 2003 Mail Server API Guide

February 25, 2005

Introduction

Ensim® Pro 4.0.1 for Windows® 2003 defines a fixed set of email-related tasks that you can perform using the control panel at each level of administration. Each task maps to one or more mail server backend API (application programming interface) calls. Ensim Pro 4.0.1 for Windows 2003 calls the API at appropriate times.

> Note
>
> For simplicity, we refer to Ensim Pro 4.0.1 for Windows 2003 as “Ensim Pro” throughout this document.

Ensim offers two types of API calls to help you integrate your mail server into the existing Ensim Pro setup: **Mail Server API** and **Registration API**.

- **Mail Server API.** This is a backend API used by Ensim Pro to carry out various actions on the underlying mail server. This API is called whenever you perform any action on the mail server through the Ensim Pro control panel.
- **Registration API.** This API is used to register your mail server add-on with Ensim Pro. As a part of the registration process, the mail server backend API that you provide gets plugged into Ensim Pro.

This document defines the semantics of all the API calls. Topics in this guide include:

Before you start

Before you start integrating the API calls, you may find it helpful to review the following list of rules.

- The mail server backend API calls use either an executable or a script written in any programming language that returns “0” on success and “1” on failure. If it is a script, you must specify the location of the executable that will interpret the script and the parameters it requires while registering the mail add-on. For instructions on registering the mail add-on, please refer to “Registration API calls” on page 24.
- The API commands included in this guide are executables and their syntax is:

  APIcommand.exe <*parameter1_value*> <*parameter2_value*> ...

  where
  APIcommand.exe is the name of the executable that implements the API call,
  <*parameter1_value*> is the value of the first parameter for the command,
  <*parameter2_value*> is the value of the second parameter for the command, and so on.

  Note

  Any additional capabilities provided by your mail server are independent and will not be reflected in the Ensim Pro control panel.

- Each backend API implementation performs an input parameter validation and fails the call if any invalid parameters are passed to it.
- **Optional:** If a certain functionality exposed by the Ensim Pro control panel is not supported by your mail server, Ensim Pro displays an error message that it cannot locate the corresponding executable or script to use the API. However, if you want to customize the error message, you can provide a placeholder implementation for the API and fail the API call by returning “1” and printing the appropriate error message.

Mail Server API calls

The Mail Server API commands provided in Ensim Pro 4.0.1 for Windows 2003 include some new commands, some updated commands, and some commands that have been deleted but were present in Ensim WEBppliance™ Pro 3.6 for Windows 2003.

Table 1. New API commands included in Ensim Pro

Type of command	Command name
New commands	EAddDistributionList EAddDistributionListMember EAddMailingList EAddMailingListMember EDeleteDistributionList EDeleteDistributionListMember EDeleteMailingList EDeleteMailingListMember EGetCatchallUser EGetDistributionListList EGetDistributionListMemberList EGetMailingListList EGetMailingListMemberList ESetCatchallUser ESuspendUser EResumeUser EPreExportDomain EPostExportDomain EPreImportDomain EPostImportDomain EPreExportUser EPostExportUser EPreImportUser EPostImportUser

Table 2. API commands updated in Ensim Pro

Name of the command	Change
EAddDomain	**Old Parameters:** DomainName, IpAddress, MailBoxDirectory, MaxMailBoxes, MailDomainOwner, MailDomainOwnerPassword **New parameters added:** SiteType, EnableMailingList, EnableDomainAliases, EnableDistributonList, EnableSpamFilters, EnableUserAliases, EnableForwards, EnableAutoResponders, EnableWebAccess

Table 2. API commands updated in Ensim Pro (continued)

Name of the command	Change
EEditDomain	**Old Parameters:** DomainName, MaxMailBoxes **New parameters added:** SiteType, EnableMailingList, EnableDomainAliases, EnableDistributonList, EnableSpamFilters, EnableUserAliases, EnableForwards, EnableAutoResponders, EnableWebAccess
EAddUser	**Old Parameters:** DomainName, UserName, Password, FirstName, MiddleName, LastName **New parameters added:** EnableForwards, EnableAutoResponders, EnableWebAccess
EEditUser	**Old Parameters:** DomainName, UserName, Password, FirstName, MiddleName, LastName **New parameters added:** EnableForwards, EnableAutoResponders, EnableWebAccess
EdeleteDomainAlias	**Old Parameters:** DomainAlias **New parameters added:** DomainName
EgetSpamFilterList	On executing this command, it displays only those email addresses that have a spam filter record against them. It does not display the type of spam filter or the allow/block flag.
EAddSpamFilter	**Old Parameters:** DomainName bAllow SpamFilterType SpamFilter **New parameters:** DomainName, SpamFilterAddress, TypeOfSpamFilter
EdeleteSpamFilter	**Old Parameters:** DomainName, bAllow, SpamFilterType, SpamFilter **New parameters:** DomainName, SpamFilterAddress
EgetForwardList	On executing this command, it displays only the forward records of the specified username. It does not display the UserName.
EGetUserAliasList	On executing this command, it displays only the user aliases and not the UserName.

Table 3. API commands deleted from Ensim Pro

Type of command	Command name
Deleted commands	EGetDomainList EGetUserList EGetUserInfo EAuthenticateUser EGetStatus EConfigureMailServer EBackupMailServer ESetMailUrl EDeleteMailURL EIsInstalled

The APIs in this section include:

Add mail domain

Command name

EAddDomain.exe

Parameters

DomainName - *Name of the mail domain.*

IpAddress - *IP address of the mail domain.*

MailBoxDirectory - *Full path to the directory where the mails for this mail domain should be stored.*

MaxMailBoxes - *Maximum number of mailboxes allowed for this mail domain.*

MailDomainOwner - *Owner of this mail domain. This user is same as the site administrator.*

MailDomainOwnerPassword - *Plain text password for the owner of the mail domain specified by MailDomainOwner parameter.*

ipsite - *Value is equal to "1" for an IP-based site and "0" for a name-based site.*

EnableMailingLists - *Value is equal to "1" for enabling and "0" for disabling.*

EnableDomainAliases - *Value is equal to "1" for enabling and "0" for disabling.*

EnableDistributonLists - *Value is equal to "1" for enabling and "0" for disabling.*

EnableSpamFilters - *Value is equal to "1" for enabling and "0" for disabling.*

EnableUserAliases - *Value is equal to "1" for enabling and "0" for disabling.*

EnableForwards - *Value is equal to "1" for enabling and "0" for disabling.*

EnableAutoResponders - *Value is equal to "1" for enabling and "0" for disabling.*

EnableWebAccess - *Value is equal to "1" for enabling and "0" for disabling.*

Result

The API adds the mail domain and the user specified by MailDomainOwner on the mail domain, and sets the catchrest alias for the mail domain to point to this user. It also sets the properties of the mail domain according to the remaining parameters passed to it.

Note

If you want to set the number of mailboxes to unlimited, the value of the MaxMailBoxes parameter should be “-1”. This value is also valid in the backward compatibility mode.
For more information on the backward compatibility mode, please refer to the section “Appendix A: Backward compatibility for existing mail add-ons” on page 25.

Edit mail domain

Command name

EEditDomain.exe

Parameters

DomainName - *Name of the mail domain.*

MaxMailBoxes - *Maximum number of mailboxes allowed for this mail domain.*

EnableMailingLists - *Value is equal to “1” for enabling and “0” for disabling.*

EnableDomainAliases - *Value is equal to “1” for enabling and “0” for disabling.*

EnableDistributonLists - *Value is equal to “1” for enabling and “0” for disabling.*

EnableSpamFilters - *Value is equal to “1” for enabling and “0” for disabling.*

EnableUserAliases - *Value is equal to “1” for enabling and “0” for disabling.*

EnableForwards - *Value is equal to “1” for enabling and “0” for disabling.*

EnableAutoResponders - *Value is equal to “1” for enabling and “0” for disabling.*

EnableWebAccess - *Value is equal to “1” for enabling and “0” for disabling.*

Result

The API verifies whether the existing number of users on the mail domain is greater than the value of MaxMailBoxes. In such cases, the API fails the call with the appropriate error message. It also sets the properties of the mail domain according to the remaining parameters passed to it.

Suspend mail domain

Command name

ESuspendDomain.exe

Parameters

DomainName - *Name of the mail domain.*

Result

If your mail server does not support suspending a mail domain, the API provides a placeholder implementation. The placeholder implementation simply returns a success.

Resume mail domain

Command name

EResumeDomain.exe

Parameters

DomainName - *Name of the mail domain.*

Result

If your mail server does not support suspending and resuming a mail domain, the API provides a placeholder implementation. The placeholder implementation simply returns a success.

Delete mail domain

Command name

EDeleteDomain.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API deletes all the information associated with the mail domain including its users, domain aliases, user forwards, and aliases.

Note

Ensim Pro does not make any additional backend calls to clean the "delete" process.

Add mail domain alias

Command name

EAddDomainAlias.exe

Parameters

DomainName - *Name of the mail domain.*

DomainAlias - *Name of the domain alias.*

Result

The API adds a new mail domain alias.

Note

Ensim Pro does not expect the API to add the necessary DNS records. You can choose to add the necessary DNS records on your own.

Delete mail domain alias

Command name

EDeleteDomainAlias.exe

Parameters

DomainName - *Name of the mail domain.*

DomainAlias - *Name of the domain alias.*

Result

The API deletes the specified mail domain alias.

Note

Ensim Pro does not expect the API to remove the necessary DNS records. You can choose to remove the necessary DNS records on your own.

Get mail domain alias list

Command name

EGetDomainAliasList.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API prints all the domain aliases for a mail domain specified by DomainName on standard output. Each domain alias name is printed on a separate line.

Add mail domain user

Command name

EAddUser.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Password - *Plain text password for the mail domain user.*

FirstName - *First name of the user.*

MiddleName - *Middle name of the user.*

LastName - *Last name of the user.*

UserEnableForwards - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableAutoResponders - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableWebAccess - *Value is equal to "1" for enabling and "0" for disabling.*

Result

The API adds all the information associated with the mail domain user.

Edit mail domain user

Command name

EEditUser.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Password - *Plain text password for the mail domain user.*

FirstName - *First name of the user.*

MiddleName - *Middle name of the user.*

LastName - *Last name of the user.*

UserEnableForwards - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableAutoResponders - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableWebAccess - *Value is equal to "1" for enabling and "0" for disabling.*

Result

The API modifies all the information associated with the mail domain user.

Delete mail domain user

Command name

EDeleteUser.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Result

The API deletes all the information associated with the mail domain user including aliases, forwards, and autoresponders. It also cleans up the mail storage used by this user.

Note

Ensim Pro does not make any additional backend calls to clean up the user's mail storage.

Add mail domain user alias

Command name

EAddUserAlias.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

UserAlias - *User alias for UserName.*

Result

The API adds the specified alias and ensures that there are no users hosted on DomainName with the same alias (as specified by UserAlias).

Delete mail domain user alias

Command name

EDeleteUserAlias.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

UserAlias - *User alias for the mail domain user.*

Result

The API deletes the specified alias.

Get mail domain user alias list

Command name

EGetUserAliasList.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Result

The API displays the user aliases for the specified UserName, one on each line.

Example

If “u1” is a user on a mail domain “xyz.com” and has two aliases defined, “al1” and “al2”, then the output of “EGetUserAliasList.exe xyz.com u1” will print the following on the standard output

```
al1

al2
```

Add mail domain user forward

Command name

EAddForward.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

UserForward - *Forward email address for UserName.*

Result

The API adds the forward address for the specified user.

Delete mail domain user forward

Command name

EDeleteForward.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

UserForward - *Forward email address for UserName.*

Result

The API deletes the specified forward.

Get mail domain user forward list

Command name

EGetForwardList.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Result

The API displays the forwards for the specified username, one on each line.

Example

If "u1" is a user on a mail domain "foo.com" and has two forward records defined: "u1@alternatemail.com" and "u1@mymail.com", then the output of "EGetForwardList.exe foo.com u1" should print the following on the standard output

```
u1@alternatemail.com

u1@mymail.com
```

Add a distribution list

Command name

EAddDistributionList.exe

Parameters

DomainName - *Name of the mail domain.*

DistributionListName - *Name of the distribution list.*

Result

The API creates the distribution list in the specified mail domain.

Delete a distribution list

Command name

EDeleteDistributionList.exe

Parameters

DomainName - *Name of the mail domain.*

DistributionListName - *Name of the distribution list.*

Result

The API deletes the distribution list from the specified mail domain.

Display the list of distribution lists

Command name

EGetDistributionListList.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API displays the list of all distribution lists belonging to the specified mail domain, one on each line.

Add a member to a distribution list

Command name

EAddDistributionListMember.exe

Parameters

DomainName - *Name of the mail domain.*

DistributionListName - *Name of the distribution list.*

DistributionListMemberEmailAddress - *Email address of the user to be added.*

Result

The API adds the user to the specified distribution list.

Delete a member from a distribution list

Command name

EDeleteDistributionListMember.exe

Parameters

DomainName - *Name of the mail domain*.

DistributionListName - *Name of the distribution list.*

DistributionListMemberEmailAddress - *Email address of the user to be deleted.*

Result

The API deletes the user from the specified distribution list.

Display the list of members belonging to a distribution list

Command name

EGetDistributionListMemberList.exe

Parameters

DomainName - *Name of the mail domain.*

DistributionListName - *Name of the distribution list.*

Result

The API displays the list of all members belonging to the specified distribution list, one on each line.

Add a mailing list

Command name

EAddMailingList.exe

Parameters

DomainName - *Name of the mail domain.*

MailingListName - *Name of the mailing list to be created.*

Result

The API creates the mailing list in the specified mail domain.

Delete a mailing list

Command name

EDeleteMailingList.exe

Parameters

DomainName - *Name of the mail domain.*

MailingListName - *Name of the mailing list to be deleted.*

Result

The API deletes the mailing list from the specified mail domain.

Display the list of mailing lists

Command name

EGetMailingListList.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API displays the list of all the mailing lists belonging to the specified mail domain, one on each line.

Add a member to a mailing list

Command name

EAddMailingListMember.exe

Parameters

DomainName - *Name of the mail domain.*

MailingListName - *Name of the mailing list.*

MailingListMemberName - *Name of the user to be added to the mailing list.*

MailingListMemberEmailAddress - *Email address of the user to be added to the mailing list.*

Result

The API adds the user to the specified mailing list.

Delete a member from a mailing list

Command name

EDeleteMailingListMember.exe

Parameters

DomainName - *Name of the mail domain.*

MailingListName - *Name of the mailing list.*

MailingListMemberEmailAddress - *Email address of the user to be deleted from the mailing list.*

Result

The API deletes the user from the specified mailing list.

Display the list of members belonging to a mailing list

Command name

EGetMailingListMemberList.exe

Parameters

DomainName - *Name of the mail domain.*

MailingListName - *Name of the mailing list.*

Result

The API displays the list of all members belonging to the specified mailing list, one on each line.

Suspend a user

Command name

ESuspendUser.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the user to be suspended.*

Result

The API suspends the specified user.

Resume a user

Command name

EResumeUser.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the user to be resumed.*

Result

The API resumes the specified user.

Set a catchall user for a mail domain

Command name

ESetCatchallUser.exe

Parameters

DomainName - *Name of the mail domain.*

CatchallUserEmailAddress - *Email address of the user who will receive all the email messages returned from that mail domain.*

Result

The API adds a user to the specified mail domain as a catchall user who will receive all the email messages returned from that mail domain.

Display the catchall users for a mail domain

Command name

EGetCatchallUser.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API displays the list of all the catchall users belonging to the specified mail domain.

Set up autoresponder for mail domain user

Command name

ESetAutoResponder.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Subject - *Subject text for auto-response email.*

Message - *Message text for auto-response email.*

Result

The API sets up an autoresponder with the specified text for the mail domain user.

Get autoresponder record for mail domain user

Command name

EGetAutoResponder.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Result

If autoresponder is not enabled for the specified user, the API does not display anything on the screen and returns “0” (successful). If autoresponder is enabled for the specified user, the API prints the Subject text and Message text on the standard output. The first line of output is the Subject text and rest is the Message text.

Delete autoresponder record for mail domain user

Command name

EDeleteAutoResponder.exe

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

Result

The API deletes all the autoresponder records for the specified mail domain user.

Add spam filter for mail domain

Command name

EAddSpamFilter.exe

Parameters

DomainName - *Name of the mail domain.*

SpamFilterAddress - *Spam filter address.*

TypeOfSpamFilter - *Type of spam filter to be applied.*

Result

The API adds a spam filter for the specified mail domain. Depending upon the type of spam filter, the spam filter address is interpreted as email, Domain, or IP address.

- If TypeOfSpamFilter=”EMAIL”, the API will add an email spam filter
- If TypeOfSpamFilter=”IP”, the API will add an IP spam filter
- If TypeOfSpamFilter=”DOMAIN”, the API will add a domain spam filter

Delete spam filter for mail domain

Command name

EDeleteSpamFilter.exe

Parameters

DomainName - *Name of the mail domain.*

SpamFilter - *Spam filter address.*

Result

The API removes the spam filter for the specified mail domain.

Get spam filter list

Command name

EGetSpamFilterList.exe

Parameters

DomainName - *Name of the mail domain.*

Result

The API displays all spam filters for the specified domain, one on each line.

Export a mail domain

Command name

EPreExportDomain.exe - *This command must be called at the beginning of the export process.*

EPostExportDomain.exe - *This command must be called at the end of the export process.*

Note

Both the commands use the same list of parameters given below.

Parameters

DomainName - *Name of the mail domain.*

MailBoxDirectory - *Location of the directory where the mails for this mail domain should be stored.*

ExportImportDataLocation - *Location of the directory where the data will be stored.*

MailDomainOwner - *Owner of the mail domain. This user is same as the site administrator.*

EnableMailingLists - *Value is equal to "1" for enabling and "0" for disabling.*

EnableDomainAliases - *Value is equal to "1" for enabling and "0" for disabling.*

EnableDistributonLists - *Value is equal to "1" for enabling and "0" for disabling.*

EnableSpamFilters - *Value is equal to "1" for enabling and "0" for disabling.*

EnableUserAliases - *Value is equal to "1" for enabling and "0" for disabling.*

EnableForwards - *Value is equal to "1" for enabling and "0" for disabling.*

EnableAutoResponders - *Value is equal to "1" for enabling and "0" for disabling.*

EnableWebAccess - *Value is equal to "1" for enabling and "0" for disabling.*

Result

The API calls provide the ability to export the mail domain to the specified directory.

Import a mail domain

Command name

EPreImportDomain.exe - *This command must be called at the beginning of the import process.*

EPostImportDomain.exe - *This command must be called at the end of the import process.*

Parameters

Both the commands use the same list of parameters as mentioned for exporting a mail domain.

Result

The API calls provide the ability to import the mail domain from the specified directory.

Export a mail domain user

Command name

EPreExportUser.exe - *This command must be called at the beginning of the export process.*

EPostExportUser.exe - *This command must be called at the end of the export process.*

Note

Both the commands use the same list of parameters given below.

Parameters

DomainName - *Name of the mail domain.*

UserName - *Name of the mail domain user.*

MailBoxDirectory - *Location of the directory where the mails for this mail domain should be stored.*

ExportImportDataLocation - *Location of the directory where the data will be stored.*

UserEnableForwards - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableAutoResponders - *Value is equal to "1" for enabling and "0" for disabling.*

UserEnableWebAccess - *Value is equal to "1" for enabling and "0" for disabling.*

Result

The API calls provide the ability to export the mail domain user to the specified directory.

Import a mail domain user

Command name

EPreImportUser.exe - *This command must be called at the beginning of the import process.*

EPostImportUser.exe - *This command must be called at the end of the import process.*

Parameters

Both the commands use the same list of parameters as mentioned for exporting a mail domain user.

Result

The API calls provide the ability to import the mail domain user from the specified directory.

Get Web interface URL

Command name

EGetMailURL.exe

Parameters

DomainName - *Name of the mail domain.*

Result

If your mail server supports a Web-based interface, the API prints the Web URL for the specified mail domain on standard output. Otherwise, the API fails with an exit code of "1".

Start mail server

Command name

EStartMailServer.exe

Parameters

None

Result

The API performs the tasks required to start all the mail services.

Stop mail server

Command name

EStopMailServer.exe

Parameters

None

Result

The API performs the tasks required to stop all the mail services.

Change domain IP address

Command name

EChangeDomainIP.exe

Parameters

DomainName - *Name of the mail domain*

IPAddress - *New IP address for the mail domain*

Result

The API changes the IP address of the specified domain to the value specified by IPAddress.

Change network settings

Command name

EChangeNetworkSettings.exe

Parameters

OldServerIP - *Old primary IP address of the server.*

OldServerFQDN - *Old fully qualified domain name of the server.*

NewServerIP - *New primary IP address of the server.*

NewServerFQDN - *New fully qualified domain name of the server.*

Result

The API updates the mail server with new network settings.

Note

Ensim Pro calls this API when you change the primary IP address of the server and then uses the network configuration utility provided by Ensim Pro to configure it for new network settings. The API will completely take care of updating the mail server with new network settings. Ensim Pro will not touch any mail server configuration.

Registration API calls

Register mail add-on

Command name

RegisterMailAddon.exe

Parameters

VendorName - *Name of the vendor.*

MailServerName - *Name of the mail server.*

MailServerVersion - *Mail server version number.*

MailAPIExtension - *Extension of the API calls.*

MailAPIInterpreter - *Interpreter used by the API calls. If this parameter is equal to "", then a null value is passed.*

MailAPIInterpreterParams - *Parameters to be passed to the interpreter. If you are passing more than one parameter, then they must be separated by a space ( ) or a comma (,) or as required by the interpreter. If this parameter is equal to "", then a null value is passed.*

3.XCompliantAPI - *If you want to use the mail add-on written for Ensim WEBppliance Pro 3.X for Windows 2003, set this parameter to "1".*

Result

The API registers the specified mail add-on.

Note

The RegisterMailAddon command resides in **<*install_dir*>\cp\admin\bin\serviceComponents\mail**, where **<*install_dir*>** refers to the path where Ensim Pro is installed. For example, if Ensim Pro is installed in **C:\Program Files\Ensim\WEBppliance**, then this command resides in the **C:\Program Files\Ensim\WEBppliance\cp\admin\bin\serviceComponents\mail** directory.

If you are registering an add-on written for **Ensim WEBppliance Pro 3.X for Windows 2003**, and the API calls are executables, run the following command.

RegisterMailAddon.exe VendorName MailServerName MailServerVersion exe "" "" 1

If you are registering an add-on written for **Ensim Pro 4.0.1 for Windows 2003**, and the API calls are implemented as VBScripts, run the following command.

RegisterMailAddon.exe VendorName MailServerName MailServerVersion vbs cscript //nologo

Appendix A: Backward compatibility for existing mail add-ons

Ensim Pro supports mail add-ons that are based on the Mail API commands included in Ensim WEBppliance Pro 3.X for Windows 2003. This backward compatibility feature can be enabled using the 3.XCompliantAPI parameter while registering the mail add-on with the RegisterMailAddon command.

In backward compatibility mode, the existing mail add-ons are supposed to work as is with Ensim Pro. Exceptions to this include:

1. The new APIs will not work, since they are not implemented by existing mail add-ons. Ensim Pro will throw exceptions in such cases.
2. If the mail add-on assumes Ensim WEBppliance Pro 3.X environment, it may not work as is and might require some modifications.

Feedback and support

To take advantage of Ensim's support services or to find additional product documentation, visit the Ensim support site, http://support.ensim.com.

To log in to Ensim online support, go to https://onlinesupport.ensim.com.

To find answers to frequently asked questions or resolve product issues, visit the Knowledge Base section on Ensim's support site at http://onlinesupport.ensim.com/kb_search_sln.asp.

To provide feedback about Ensim products or documentation, please use the feedback form at http://www.ensim.com/about/feedback.asp.

Legal and copyright notice

This document contains information proprietary to Ensim Corporation and its receipt or possession does not convey any rights to reproduce, disclose, manufacture, or sell anything it might describe. Reproduction, disclosure, or use without Ensim's specific written authorization is strictly forbidden.

Ensim Corporation makes no representations or warranties with respect to the content or use of this document. It also reserves the right to revise this document at any time without the obligation to notify any person of such revision.

Ensim and the Ensim logo are registered trademarks of Ensim Corporation.

All other trademarks and copyrights are the property of their respective owners.

Ensim Corporation
1366 Borregas Avenue
Sunnyvale, CA 94089
(408) 745-3300
www.ensim.com