

DeskNow

Administration and Configuration

Version : 3.2
Date : 5 May 2009



Ventia Pty Limited A.C.N. 090 873 662
Web : <http://www.ventia.com>
Email : info@ventia.com

Overview

This manual describes the procedures involved in administering and configuring a DeskNow server.

DeskNow is a computing platform that provides a rich collaborative environment, accessible from everywhere and from different devices. The DeskNow platform is open to third-party developers and to integration with legacy systems.

A more detailed overview of DeskNow can be found at <http://www.desknow.com>.

In this document it is assumed that the Reader is already familiar with DeskNow features and concepts.

The Reader of this manual that wants to perform custom setup and advanced administration should also have a good knowledge of web server products, RDBMS systems and system administration (Windows/Unix) in general.

Legal

DeskNow is a registered trademark of Ventia Pty Limited.

Every other trademark present in this document belongs to its registered owner.

Several patent applications have been lodged by Ventia Pty Ltd to cover ideas and technologies that are illustrated in this document.

The concepts, names, or functionalities illustrated in this document may change without prior advice.

Table of contents

Overview.....	2
Legal.....	3
Table of contents.....	4
1 Introduction.....	14
2 Basic administration.....	15
2.1 Starting and Stopping DeskNow.....	15
2.1.1 Simple installation.....	15
2.1.2 Advanced installation.....	15
3 User administration.....	16
3.1.1 Domains.....	16
3.1.2 Communities.....	16
3.1.3 Users.....	17
3.1.4 User status.....	17
3.1.5 User groups.....	17
3.1.6 User classes.....	18
3.1.7 Planning disk quotas.....	18
3.1.8 Planning the virtual file system.....	19
3.1.9 Creating users, communities and domains.....	19
3.1.10 Bulk creation of user accounts.....	19
3.1.11 Custom authentication.....	20
3.1.12 Self registration.....	20
3.1.13 Domains and login.....	20
3.1.14 Domains and POP3, IMAP and SMTP authentication.....	21
3.1.15 Su-like login.....	21
3.1.16 Su-like login for community managers.....	22
3.1.17 Resetting the admin password.....	22
3.1.18 Account expiration.....	22
3.2 Importing accounts and emails from another mail server.....	23
3.2.1 Migrating from Exchange or other Active Directory or LDAP- based mail server.....	23
3.2.2 Migrating from a standalone mail server.....	24
4 Mail.....	25
4.1.1 DeskNow as MTA (Mail Transfer Agent).....	25
4.1.2 DeskNow as SMTP server.....	25
4.1.3 SMTP Forwarding.....	25
4.1.4 DeskNow as POP3 server.....	26
4.1.5 Mailing lists.....	26
4.1.6 Using DeskNow to retrieve all emails for a company from a single POP3 inbox.....	26
4.1.7 More configuration options.....	27
4.2 Spam filtering.....	27
4.2.1 DNSBL.....	27
4.2.2 SURBL.....	27
4.2.3 Bayesian analysis.....	27
4.2.4 Whitelists.....	28
4.2.5 Blacklists.....	28
4.2.6 Greylisting.....	28
4.2.7 SPF (Sender Policy Framework).....	29
4.2.8 Time limited email addresses.....	29
4.2.9 Spam filtering in email clients.....	29
4.3 DeskNow as default mail client.....	30
4.3.1 Default mail client in Windows.....	30
4.4 Importing mail messages from other applications.....	30
5 Server administration.....	31

5.1 Backups	31
5.1.1 Restore	31
5.1.2 Disaster recovery	33
5.2 Antivirus scanning	33
5.2.1 Configuration of Symantec/Norton Antivirus Corporate Edition	34
5.3 Logging	35
5.3.1 Log files	35
5.3.2 Plugins to post-process log messages	35
6 Audit logging	36
6.1.1 Where are the audit logs	36
6.1.2 Format of audit logs	36
6.1.3 Audit logs for Instant Messaging	36
6.1.4 Audit logs storage and maintenance	36
7 Using an external database	37
7.1 The Database Initialization Wizard	37
8 Using DeskNow with IIS or Apache	38
8.1 Simple solution: using different ports	38
8.2 Apache 1.3.x and 2.x	38
8.2.1 Using mod_proxy	38
8.2.2 Using mod_jk	39
8.3 Microsoft IIS on Windows Server	39
9 Using your own SSL certificate	41
9.1.1 To simply stop browser alerts without changing certificates	41
9.1.2 If you're are upgrading from a release before 2.7	41
9.1.3 To install a certificate signed by a Certificate Authority (Windows)	41
9.1.4 To install a self-signed certificate (Windows)	42
9.1.5 To install a certificate signed by a Certificate Authority (Unix/Linux/Mac OSX)	43
9.1.6 To install a self-signed certificate (Unix/Linux/ Mac OSX)	44
9.1.7 SSL for SMTP, POP3 and IMAP	44
9.1.8 SSL for virtual domains	45
10 External authentication	46
10.1 ActiveDirectory authentication	46
10.2 ActiveDirectory with Kerberos authentication	48
10.3 Kerberos authentication	49
10.4 LDAP authentication	49
10.5 Custom authentication	52
10.6 Switching back to internal authentication	52
11 Self registration	53
11.1 Types of self registration	53
11.2 Configuration	53
11.3 Personalization	53
11.3.1 Personalization of the self-registration pages	54
11.3.2 Personalization of the confirmation pages	54
12 Language and timezone	55
12.1 Default language and user language	55
12.2 Adding a language to DeskNow	55
12.2.1 Translation tips	56
12.2.2 Testing your progress	57
12.2.3 Share your work	57
12.2.4 What happens when a new version of DeskNow is released	57
12.3 Time zone	57
13 Clustering and load balancing	58
13.1 Advantages	58
13.2 Overview	58
13.3 Load balancing	58
13.3.1 Load balancing client requests	59

13.3.2	Load balancing batch jobs	59
13.3.3	Load balancing MX records	59
13.3.4	Load balancing instant messaging	59
13.4	Storage	59
13.4.1	Database	59
13.4.2	Files	59
13.5	Server to Server communication	60
13.5.1	Members of the cluster	60
13.5.2	Heartbeat	60
13.5.3	New server notice	60
13.5.4	Missing server alert	60
13.6	Caching	60
13.7	Cluster setup	61
13.7.1	Database setup	61
13.7.2	Installing the first server	61
13.7.3	Starting the first server	62
13.7.4	Adding other servers	62
13.8	Cluster administration	63
13.8.1	Monitoring	63
13.8.2	Configuration changes	63
13.8.3	Backup	63
13.9	Removing a server	63
14	Configuration reference	65
14.1	Configuration files	65
14.1.1	Simple installation	65
14.1.2	Advanced installation	65
14.1.3	Format of the configuration files	65
14.2	Main.cfg	66
14.2.1	ServerTimeZone	66
14.2.2	Network.LocalIP	66
14.3	DeskNowBase.cfg	67
14.3.1	%SET%DESKNOWDATA	67
14.4	Log.cfg	68
14.4.1	BaseFileName	68
14.4.2	AuditBaseFolder	68
14.4.3	FileANDConsole	68
14.4.4	CleanupDays	68
14.4.5	LogListener	68
14.4.6	Debug	68
14.4.7	InternalError	68
14.4.8	Security	68
14.4.9	Login	69
14.4.10	Admin	69
14.4.11	Database	69
14.4.12	Messenger	69
14.4.13	MessengerDebug	69
14.4.14	MessengerNIODebug	69
14.4.15	MessengerHTTPDebug	69
14.4.16	SMTPIn	69
14.4.17	SMTPInDebug	69
14.4.18	SMTPOut	70
14.4.19	SMTPOutDebug	70
14.4.20	POP3	70
14.4.21	POP3Debug	70
14.4.22	MailIngestion	70
14.4.23	MailIngestionDebug	70

14.4.24 Internationalization	70
14.4.25 IMAP	70
14.4.26 IMAPDebug	70
14.4.27 IMAPProtocol	70
14.4.28 Audit	71
14.4.29 AuditIM	71
14.5 Database.cfg	71
14.5.1 JdbcDriver	71
14.5.2 Url	71
14.5.3 Username	71
14.5.4 Password	71
14.5.5 NumConnections	71
14.5.6 ConnectionPoolTimeout	72
14.5.7 IDFactory.DBAllocatedPoolSize	72
14.5.8 IDFactory.ChunckedPoolSize	72
14.5.9 IDFactory.NumProducers	72
14.6 Zroots.cfg	73
14.7 Vroots.cfg	74
14.8 Registration.cfg	75
14.8.1 MinCommunityNameLength	75
14.8.2 MinUserNameLength	75
14.8.3 MinPasswordLength	75
14.8.4 MinPasswordNonLetters	75
14.8.5 MinPasswordLetters	75
14.8.6 InitialFiles section	75
14.8.7 ReservedNames section	75
14.9 VisibleFolders.cfg	76
14.10 Security.cfg	77
14.10.1 ResetAdminPassword	77
14.10.2 AllowSuLoginFromCommunityManager	77
14.10.3 MaxFileUploadSize	77
14.10.4 ShowServerFiles	77
14.10.5 AllowCrossDomainSharing	77
14.10.6 TrackLastLoginTimeFromExternalConnections	77
14.10.7 EnableShareForcingAcrossDomains	77
14.10.8 ExternalAuthentication.Enable	78
14.10.9 ExternalAuthentication.CacheValidityTime	78
14.10.10 ExternalAuthentication.AutomaticAccountCreation.Enable	78
14.10.11 ExternalAuthentication.AutomaticAccountCreation.ImportADMailAliases	78
14.10.12 ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Enable	78
14.10.13 ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Host	78
14.10.14 ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Port	78
14.10.15 ExternalAuthentication.AutomaticAccountCreation.Community	79
14.10.16 ExternalAuthentication.AutomaticAccountCreation.UserClass	79
14.10.17 ExternalAuthentication.AutomaticAccountCreation.DefaultDiskQuota	79
14.10.18 ExternalAuthentication.AuthenticationProvider	79
14.10.19 ExternalAuthentication.CustomProperties.ServerAddress	79
14.10.20 ExternalAuthentication.CustomProperties.ServerPort	79
14.10.21 ExternalAuthentication.CustomProperties.BrowserUsername	79
14.10.22 ExternalAuthentication.CustomProperties.BrowserPassword	79
14.10.23 ExternalAuthentication.CustomProperties.SearchSubtrees	80
14.10.24 ExternalAuthentication.CustomProperties.AuthenticationMechanisms	80
14.10.25 ExternalAuthentication.CustomProperties.CustomQuery.x	80
14.10.26 ExternalAuthentication.CustomProperties.UIDAttribute	80
14.10.27 ExternalAuthentication.CustomProperties.UseCNAttribute	80
14.10.28 ExternalAuthentication.CustomProperties.GNAttribute	80

14.10.29 ExternalAuthentication.CustomProperties.SNAttribute	81
14.10.30 IMAPAutoImport.Enable	81
14.10.31 IMAPAutoImport.ExternalIMAPServer.Host	81
14.10.32 IMAPAutoImport.ExternalIMAPServer.Port	81
14.10.33 IMAPAutoImport.ExternalIMAPServer.SSL	81
14.10.34 FailedLoginAttemptTracker.enable	81
14.10.35 FailedLoginAttemptTracker.CheckPeriod	81
14.10.36 FailedLoginAttemptTracker.LockoutPeriod	81
14.10.37 FailedLoginAttemptTracker.TriggerAttempts	81
14.10.38 RemoveSharesPointingToNonExistingFolders	81
14.11 Mail.cfg	83
14.11.1 Domain	83
14.11.2 Administrator	83
14.11.3 AddressSeparators	83
14.11.4 ExternalPOP3	83
14.11.5 ExternalPOP3.delay	83
14.11.6 ExternalPOP3.fetchThreads	84
14.11.7 ExternalPOP3.parseThreads	84
14.11.8 ExternalPOP3.rootPath	84
14.11.9 ExternalPOP3.retryDelay	84
14.11.10 ExternalPOP3.retryMaxCycles	84
14.11.11 ExternalPOP3.CopyFetchedMailToPOP3Inbox	84
14.11.12 ExternalPOP3.Timeout	84
14.11.13 ExternalPOP3.quarantinePath.SizeLimit	84
14.11.14 ExternalPOP3.quarantinePath.NumberLimit	84
14.11.15 ExternalPOP3.errorPath.SizeLimit	85
14.11.16 ExternalPOP3.errorPath.NumberLimit	85
14.11.17 DirectIn	85
14.11.18 DirectIn.use	85
14.11.19 DirectIn.delay	85
14.11.20 DirectIn.parseThreads	85
14.11.21 DirectIn.rootPath	85
14.11.22 DirectIn.retryDelay	86
14.11.23 DirectIn.retryMaxCycles	86
14.11.24 DirectIn.RecipientPrefix	86
14.11.25 DirectIn.SendBounceMessagesOnError	86
14.11.26 DirectIn.quarantinePath.SizeLimit	86
14.11.27 DirectIn.quarantinePath.NumberLimit	86
14.11.28 DirectIn.errorPath.SizeLimit	86
14.11.29 DirectIn.errorPath.NumberLimit	87
14.11.30 Webmail.allowExternalDelivery	87
14.11.31 Webmail.CheckInternalRecipientExistence	87
14.11.32 Webmail.MaxRecipients	87
14.11.33 EmbeddedServer.ServerName	87
14.11.34 EmbeddedServer.BannerSoftwareName	87
14.11.35 EmbeddedServer.useSMTPIn	87
14.11.36 EmbeddedServer.useSMTPPlain	87
14.11.37 EmbeddedServer.SMTPDaemonPort	87
14.11.38 EmbeddedServer.enableSecondarySMTPDaemonPort	88
14.11.39 EmbeddedServer.secondarySMTPDaemonPort	88
14.11.40 EmbeddedServer.SMTPBindInterface	88
14.11.41 EmbeddedServer.useSMTPSSL	88
14.11.42 EmbeddedServer.SMTPSSLDaemonPort	88
14.11.43 EmbeddedServer.SMTPSSLBindInterface	88
14.11.44 EmbeddedServer.SMTPOutInterface	88
14.11.45 EmbeddedServer.routingThreads	88

14.11.46	EmbeddedServer.remoteDeliveryThreads.....	89
14.11.47	EmbeddedServer.remoteDeliveryRetryDelay	89
14.11.48	EmbeddedServer.remoteDeliveryTimeout.....	89
14.11.49	EmbeddedServer.remoteDeliveryRetryMaxCycles.....	89
14.11.50	EmbeddedServer.SMTPThreads	89
14.11.51	EmbeddedServer.SMTPInConnectionTimeout.....	89
14.11.52	EmbeddedServer.SMTPMaxMessageSize.....	89
14.11.53	EmbeddedServer.SMTPInMaxRecipients.....	89
14.11.54	EmbeddedServer.SMTPInMaxEmailsSentPerUserPerDay.....	90
14.11.55	EmbeddedServer.SMTPInMaxEmailsSentPerUserPerHour	90
14.11.56	EmbeddedServer.rootPath.....	90
14.11.57	EmbeddedServer.usePOP3.....	90
14.11.58	EmbeddedServer.usePOP3Plain.....	90
14.11.59	EmbeddedServer.POP3DaemonPort	90
14.11.60	EmbeddedServer.POP3BindInterface	90
14.11.61	EmbeddedServer.usePOP3SSL	90
14.11.62	EmbeddedServer.POP3SSLDaemonPort.....	91
14.11.63	EmbeddedServer.POP3SSLBindInterface	91
14.11.64	EmbeddedServer.POP3Threads	91
14.11.65	EmbeddedServer.POP3ConnectionTimeout	91
14.11.66	EmbeddedServer.ForceInboxStore.....	91
14.11.67	EmbeddedServer.authorizeRelay	91
14.11.68	EmbeddedServer.AuthorizeRelayFromThisServer.....	92
14.11.69	EmbeddedServer.excludeRelay.....	92
14.11.70	EmbeddedServer.SMTPIPBlackList	92
14.11.71	EmbeddedServer.acceptSMTPAuthentication.....	92
14.11.72	EmbeddedServer.AddReceivedHeaderForAuthorizedSenders.....	92
14.11.73	EmbeddedServer.verifySMTPIdentity	92
14.11.74	EmbeddedServer.SMTPForwardHost.....	93
14.11.75	EmbeddedServer.SMTPForwardPort	93
14.11.76	EmbeddedServer.SMTPForwardUsername	93
14.11.77	EmbeddedServer.SMTPForwardPassword	93
14.11.78	EmbeddedServer.useRemoteDeliveryForExistingLocalAccounts	93
14.11.79	EmbeddedServer.useRemoteDeliveryForNonExistingLocalAccounts	93
14.11.80	EmbeddedServer.SynchronizeWebmailWithPOP3Read.....	93
14.11.81	EmbeddedServer.SynchronizeWebmailWithPOP3Delete	94
14.11.82	EmbeddedServer.DetectTimeLimitedEmailAddresses	94
14.11.83	EmbeddedServer.DNSServer.x	94
14.11.84	EmbeddedServer.EnableReverseDNS.....	94
14.11.85	EmbeddedServer.EnableBounceMessages	95
14.11.86	EmbeddedServer.EnableFirstBounceMessage	95
14.11.87	EmbeddedServer.POP3SSLKeyStore	95
14.11.88	EmbeddedServer.POP3SSLKeyStorePassword	95
14.11.89	EmbeddedServer.SMTPSSLKeyStore.....	95
14.11.90	EmbeddedServer.SMTPSSLKeyStorePassword.....	95
14.11.91	EmbeddedServer.IMAP.Enable	95
14.11.92	EmbeddedServer.IMAP.MaxConnections.....	95
14.11.93	EmbeddedServer.IMAP.Plain.Enable	95
14.11.94	EmbeddedServer.IMAP.Plain.BindInterface	96
14.11.95	EmbeddedServer.IMAP.Plain.Port.....	96
14.11.96	EmbeddedServer.IMAP.SSL.Enable	96
14.11.97	EmbeddedServer.IMAP.SSL.BindInterface	96
14.11.98	EmbeddedServer.IMAP.SSL.Port	96
14.11.99	EmbeddedServer.IMAP.AuthenticatedStateTimeout.....	96
14.11.100	EmbeddedServer.IMAP.NonAuthenticatedStateTimeout	96
14.11.101	EmbeddedServer.IMAPSSLKeyStore.....	96

14.11.102	EmbeddedServer.IMAPSSLKeyStorePassword	96
14.11.103	EmbeddedServer.ExternalSMTPFilter.enable	97
14.11.104	EmbeddedServer.ExternalSMTPFilter.host	97
14.11.105	EmbeddedServer.ExternalSMTPFilter.port	97
14.11.106	EmbeddedServer.ExternalSMTPFilter.IPmatch	97
14.11.107	EmbeddedServer.BlockLocalMailFromSpoofing	97
14.11.108	EnableBackgroundClean	97
14.11.109	OutputCharset	97
14.11.110	InputCharset	98
14.11.111	GuessInputCharset	99
14.11.112	AntiSpam.Bayes.use	99
14.11.113	AntiSpam.Bayes.EnableLearning	99
14.11.114	AntiSpam.Bayes.DataFile	99
14.11.115	AntiSpam.Bayes.MaxTokens	99
14.11.116	AntiSpam.Bayes.DefaultProbabilityThreshold	99
14.11.117	AntiSpam.DNSBL.Service.x	99
14.11.118	AntiSpam.DNSBL.Reject	99
14.11.119	AntiSpam.DNSBL.Reject.ExceptAllowedToRelay	100
14.11.120	AntiSpam.CopySpamToPOP3	100
14.11.121	AntiSpam.SendDailySpamSummary	100
14.11.122	AntiSpam.EnableSpamSummary	100
14.11.123	AntiSpam.DailySpamSummary.Hour	100
14.11.124	AntiSpam.DailySpamSummary.Minutes	100
14.11.125	AntiSpam.POP3SubjectSpamFlag	100
14.11.126	AntiSpam.Greylisting.Enable	101
14.11.127	AntiSpam.Greylisting.MinimumDelay	101
14.11.128	AntiSpam.Greylisting.Validity	101
14.11.129	AntiSpam.Greylisting.ExceptionIPs	101
14.11.130	AntiSpam.Greylisting.ExplanationMessage	101
14.11.131	AntiSpam.Greylisting.SkipIfSPFPassed	101
14.11.132	AntiSpam.Greylisting.IncludeClassCSubnet	101
14.11.133	AntiSpam.Filters.NoSubjectHeader	101
14.11.134	AntiSpam.Filters.NoFromHeader	102
14.11.135	AntiSpam.Filters.InlineImagesBiggerThan	102
14.11.136	AntiSpam.WhiteList.LocalDomains	102
14.11.137	AntiSpam.BlockFromHeaderSpoofing	102
14.11.138	ExternalIMAP.DisableAuthPlain	102
14.11.139	CheckReverseDNSPTR.HasIt	102
14.11.140	CheckReverseDNSPTR.HELO	102
14.11.141	CheckReverseDNSPTR.MAILFROM	102
14.12	Messenger.cfg	103
14.12.1	use	103
14.12.2	PlainConnector.use	103
14.12.3	PlainConnector.port	103
14.12.4	PlainConnector.NIO	103
14.12.5	SSLConnector.use	103
14.12.6	SSLConnector.port	103
14.12.7	WebConnector.use	103
14.12.8	RosterMode	104
14.12.9	DefaultPresenceAlert	104
14.13	Web.cfg	105
14.13.1	Layouts section	105
14.13.2	Custom section	105
14.13.3	Custom.original.DefaultLongDatePattern	105
14.13.4	Custom.original.DefaultShortDatePattern	105
14.13.5	Custom.original.DefaultTimePattern	106

14.13.6 Custom.original.DefaultFontSize.....	106
14.13.7 Custom.LeftMenuWidth.....	106
14.13.8 Branding.ApplicationName.....	106
14.13.9 Branding.PocketPCTitle	106
14.13.10 Branding.LinkMain.....	106
14.13.11 Branding.LinkSupport.....	107
14.13.12 LoginPage.OpenInNewWindowByDefault	107
14.13.13 EnableTips	107
14.13.14 EnableAutoRefresh	107
14.13.15 ProxiedIPHeader	107
14.13.16 NumFileUploadFields.....	107
14.14 International.cfg	108
14.14.1 DefaultLanguage	108
14.14.2 Spelling.DefaultLexicon.....	108
14.14.3 Spelling.UseTechLexicon.....	108
14.15 Backup.cfg	108
14.15.1 Automatic	108
14.15.2 Hour.....	109
14.15.3 Minute.....	109
14.15.4 Day.x	109
14.15.5 BasePath.....	109
14.15.6 EmbeddedDatabase.....	109
14.15.7 UserFolders.....	109
14.15.8 MailQueue	109
14.15.9 CfgFiles	110
14.15.10 KeepCopies.....	110
14.16 SelfRegistration.cfg.....	110
14.16.1 AllowUserSelfRegistration.....	110
14.16.2 AllowCommunitySelfRegistration	110
14.16.3 AllowDomainSelfRegistration	110
14.16.4 DefaultUserDiskQuota.....	110
14.16.5 DefaultCommunityDiskQuota.....	110
14.16.6 DefaultDomainDiskQuota.....	110
14.16.7 DefaultDomainMaxUsers	111
14.16.8 DefaultDomainMaxSyncMLUsers	111
14.16.9 DefaultUserClass	111
14.16.10 DefaultDomain.....	111
14.16.11 DefaultCommunity.....	111
14.16.12 Trial.EnableExpiration	111
14.16.13 Trial.ExpireAfterDays	111
14.16.14 Trial.DeleteAfterDays	111
14.16.15 Trial.NotifyBeforeExpirationDays	112
14.16.16 Trial.UserNotificationSubject.....	112
14.16.17 Trial.UserNotificationMessage	112
14.16.18 Trial.CommunityNotificationSubject	112
14.16.19 Trial.CommunityNotificationMessage.....	112
14.16.20 Trial.DomainNotificationSubject	112
14.16.21 Trial.DomainNotificationMessage	112
14.17 Clustering.cfg	112
14.17.1 Enable	112
14.17.2 ConfigurationReloadCheckPeriod.....	113
14.17.3 Heartbeat.MulticastGroup	113
14.17.4 Heartbeat.MulticastPort.....	113
14.17.5 Heartbeat.NetworkInterface	113
14.17.6 Heartbeat.Interval.....	113
14.17.7 Heartbeat.Validity	113

14.18 SyncML.cfg	113
14.18.1 ShowAllEvents.StoreName	113
14.18.2 ShowAllTasks.StoreName	113
14.18.3 ShowAllCalendars.StoreName	113
14.18.4 ShowAllContacts.StoreName	114
14.18.5 AddTimeZoneInfoForPDAs	114
14.18.6 UseZuluTimeForPDAs	114
14.18.7 DefaultPDASyncWithMyCalendar	114
14.18.8 DefaultPDASyncWithContacts	114
14.18.9 AutoExpiryOfDeleteRecordsDays	114
14.18.10 LogSession.Enable	114
14.18.11 LogSession.BaseFolder	114
14.18.12 LogSession.SessionValidity	114
15 Administration API	116
15.1 Overview	116
15.1.1 Invocation result	116
15.2 Create user	117
15.3 Create community	117
15.4 Create domain	117
15.5 Delete domain	118
15.6 Delete community	118
15.7 Delete user	118
15.8 Update user	118
15.9 Create user class	119
15.10 Add user to user group	119
15.11 Authenticate user	120
15.12 Create calendar	120
15.13 Empty calendar	120
15.14 Create file folder	121
15.15 Create mail folder	121
15.16 Delete old emails from mail folder	121
15.17 Create mail filter	121
15.18 Delete mail filter	122
15.19 Add address to a mailing list	122
15.20 Remove address from a mailing list	122
15.21 Add action to a mail filter	122
15.22 Add match to a mail filter	123
15.23 Recalculate mail folder statistics	123
15.24 Validate email address	124
15.25 Import email message	124
15.26 Import mbox mail folder	124
15.27 Import mail from a remote IMAP account	125
15.28 Create external POP3 account	126
15.29 Create local mail alias	126
15.30 Create global mail alias	127
15.31 Post event to a calendar	127
15.32 Post task to a calendar	127
15.33 Post to message board	128
15.34 Create contact folder	128
15.35 Delete contact folder	128
15.36 Add contact	129
15.37 Delete contact	130
15.38 Send IM alert	130
15.39 Send IM broadcast	131
15.40 Re-create user directory	131
15.41 Clear user's SyncML data	131

15.42 Recalculate user's disk usage	131
15.43 Recalculate disk usage for all users of a domain	132
15.44 Recalculate disk usage for all users in the system	132
15.45 Set user preference	132
15.46 Set domain preference.....	132
15.47 Enable autoresponder.....	133
15.48 Disable autoresponder.....	133
15.49 Reload I18N files.....	134
15.50 Clear the system's cache.....	134
15.51 Emulate the expiration of the trial period	134
15.52 View the content of the Bayesian dictionary.....	134
16 Licensing DeskNow	135
17 Support	136

1 Introduction

This manual assumes that you have already successfully installed DeskNow, either using the self-installing executable in Windows, or performing a custom setup. The procedures are detailed respectively in the "Quick Start" page and "DeskNow – Advanced setup" document.

2 Basic administration

2.1 Starting and Stopping DeskNow

The way you start and stop DeskNow server depends on the installation that you performed.

2.1.1 Simple installation

From the Windows Start menu, select “Programs/DeskNow/Start DeskNow” and “Stop DeskNow” respectively. If you are running DeskNow on Windows NT, 2000 or XP, you have probably also installed DeskNow as a service. In this case, DeskNow will start automatically when the system starts.

2.1.2 Advanced installation

Modern application servers/servlet engines such as Tomcat version 4 can stop the web application from the administration page, using the “manager” web application.

Please note that in some Tomcat configurations, especially when it is connected to a web server, it is necessary to restart the web server as well.

3 User administration

The Administrator ('admin') is the only user account that DeskNow creates automatically the first time it is run. You will need to create accounts for all the other users of the system.

3.1.1 Domains

DeskNow users are organized in **domains**. A domain generally represents a separated organization. It usually has its own associated internet domain name (ex. company1.com). Domains enable ISPs to offer hosted DeskNow services to multiple customers using a single DeskNow server.

Each domain is completely separated from the others. It is as if each domain was running on its own DeskNow server. Users of a domain can use collaboration features (apart from some exceptions, like public sharing, and, of course, email) with users of another domain, but this must be explicitly enabled by the Administrator (in the Administration/Security section). This ensures security and privacy for each domain.

Each domain has its own domain administrator, communities (see below) and users. User accounts with the same name can exist in different domains without conflict. For example, there can be joe.smith in the domain company1.com and joe.smith in the domain company2.com . They are completely different users.

When run for the first time, DeskNow contains only one domain, called the *default* domain. If you are not an ISP, you can completely ignore the use of domains and create users within your default domain. Otherwise, you can create as many domains as you want. Additional domains are also known as **virtual domains**.

Each domain, including the default one, has a special user account: **admin**. The admin user is the domain administrator, and has total control over the domain. He/she can create and manage communities (see below), user accounts and assign disk quotas to them. He/she can create mail aliases for the domain, and also create public folders that are accessible to every user of the domain.

The admin user of the default domain (very likely you, the reader of this document) is called **super administrator**. This user is the most powerful user of DeskNow. He/she can create and manage other domains, and can create or delete users in any domain.

Important: DeskNow uses the name of domains to determine what to do with incoming mail messages (e.g. to decide which accounts they must be delivered to). It is therefore essential that the name of a DeskNow domain is identical to the internet domain name for which DeskNow is receiving emails (ex. company.com).

3.1.2 Communities

DeskNow offers you the possibility of organizing user accounts of a domain into *communities*. A community is a group of users within a domain that is administered by a *community manager*.

Community managers can administer (create, change, delete) accounts within their community, becoming a sort of 'admin' user for the specific community. In this way, the domain Administrator can offload part of his tasks and responsibilities, with better response times for everyone.

For small organizations, you probably don't need to group your users in communities. All the users can belong to the default community, managed by the Administrator.

In large organizations however, planning the use of communities is a good way of decentralizing the administration of accounts.

If your organization has two branches, for instance, you can create two different communities and appoint a person within each branch as manager of the respective community. In this way, the managers can create accounts for employees of their branch without having to contact the Administrator.

The domain Administrator keeps full control over all users of the domain, anyway.

3.1.3 Users

A user belongs to a specific domain and community. The disk usage of a user can be limited by the user's disk quota, the community disk quota, and the domain disk quota (if they are not set to unlimited).

There can not be two users with the same user name within a domain, but users with the same username can exist in different domains.

3.1.4 User status

A user account can be in one of the following statuses:

- Enabled: this account is normally operational
- Disabled: the user cannot log into this account, but incoming mail is still received and processed for this account
- Scheduled to be deleted: the user cannot log into this account. Incoming mail is still received and processed for this account. If you're using DeskNow with an ASP license, the account will be automatically deleted on the 'Trial deletion date' specified in the user page. If you're not running DeskNow with an ASP license, automatic trial account deletion is not available and the 'Trial deletion date' field is not shown. However the user will still receive a 'account scheduled to be deleted' error when trying to login, and know that the Administrator has the intention to manually delete the account in the near future.
- Must change password: the user must change password at the next login using the web interface.

For more information about trial accounts, please see 11.

3.1.5 User groups

User groups define sets of users generally for the purpose of granting permissions. For instance, it is possible to define a 'Managers' group to which all the manager-level users belong. It is then easy, for instance, to give access to document folders only to the 'Managers' group. When a new user is inserted in the group, he/she will inherit all the permissions granted to the group.

A user can belong to an unlimited number of groups. There can be an unlimited number of groups in a domain. A group can contain users from different communities, as an employee in the New York branch and employee in the Tokyo branch can both be managers.

Only the domain administrator can manage user groups, but users can share objects with these groups.

3.1.6 User classes

Each user belongs to a *user class*. A user class defines what features of DeskNow are enabled for users of that class.

For instance, you can enable or disable access to email, change of preferences, change of password, access to files, access to calendars, creation of personal folders, etc.

You can define as many user classes as you want, and assign different users to each class. The default user class, called 'Normal' has all permissions enabled by default (you can change that).

Example: you can create a user class called 'Guests', for guest users. For this class you may want to disable access to email, change of password or preferences, creation of personal folders...

User classes are defined for each domain. To define user classes, login as admin for the domain, then click on Administration/User and communities/Manage user classes.

IMPORTANT: User classes are a feature of DeskNow Professional. They are available in the free DeskNow Lite only during the trial period. At the end of the trial period, all users will automatically be assigned to the default user class.

3.1.7 Planning disk quotas

A disk quota is a limit to the disk space that a user, a community, or a domain can use. Setting disk quotas prevents users to abuse the system, and to ensure that the system will not run out of disk space unexpectedly.

DeskNow provides two different disk quotas:

- the soft disk quota is the amount of disk space (expressed in bytes) that a user can normally use. If a user exceeds his soft disk quota, he will receive a warning message in his account status, which appear at every login. However, he will still be able to write files to his account.
- the hard disk quota is the maximum amount of disk space that an user account can use. When an user account reaches its hard disk quota, no new files can be created in his file system, and existing files cannot be extended. Obviously, the hard disk quota should never be less than the soft disk quota (DeskNow does not enforce this, anyway). Usually DeskNow sets the hard disk quota to be 120% of the soft disk quota.

Disk quotas can be applied to single users, communities and domains. The disk usage of a community is the sum of the disk usages of its users. The disk usage of a domain is the sum of the disk usages of its communities.

Note that all the mail messages and attachments contribute to the disk occupation of an account. This means that the disk quota can be reached by just receiving mail messages. This also means that if the hard disk quota is reached, DeskNow will NOT download new mail for the account. Depending on your configuration, *this could cause the loss of emails*. In a standard configuration, however, DeskNow keeps trying storing emails to a user account for some time, at repeated intervals, before giving up and discarding the message. Whenever it encounters a disk quota error, it reports the problem in its log files.

In addition, files, messages or attachments uploaded by a user into the shared space of another user will concur to the disk usage of the user sharing the space, and not of the uploading user.

Users should always check their disk usage and remove unnecessary files, mails, messages from their account, or contact the administrator to increase their disk quota.

Anyway, you can decide to NOT apply disk quotas to users, communities or domains, by setting their disk quota to “Unlimited”.

It is also possible to exclude some sections of the file system from disk quota calculation. To do so, simply create a file named “x._DNFSNoQuota” (can be empty) in the directory that you want to exclude from disk quota calculation. Any operation (file delete, create, write) in this directory or a subdirectory will not affect the diskquota of users.

3.1.8 Planning the virtual file system

Before creating users, it is important to plan how the file system that they will be able to access is composed. In other words, what default branches will be displayed under the “Files” node on the client. Consult section 14.7 for more details.

3.1.9 Creating users, communities and domains

The Administrator can create users and communities, manage quotas and passwords, by clicking on the “Administration” node on DeskNow. With the same link, the super Administrator can also manage domains.

Important: a community must have a manager, and the manager must be a user of that community, or the domain administrator.

Administrators of a community can create, manage, add users in their community as well, by clicking on the node corresponding to their community.

DeskNow has the possibility to create user accounts in many other ways, including online self- registration for publicly accessible sites (see section 3.1.12), or with automated procedures (section 15).

3.1.10 Bulk creation of user accounts

DeskNow lets you create many user accounts in a single, fast operation, by importing the accounts from a spreadsheet. This is useful when setting up a new domain which must have many users.

All you need to do is to create a spreadsheet: each row represents an account, and the spreadsheet must have 5 columns:

- community name
- username
- password
- disk quota (in Mb, use –1 for unlimited)
- user class: must be one of the user classes defined for the domain. NB the user class is case sensitive
- first name
- middle name (can be empty)
- last name

Example:

	A	B	C	D	E	F	G	H
1	tech	richard	mypasswo	100	Restricted	Richard		Smith
2	tech	malcom	password1	-1	Normal	Malcom	G	Fraser
3	sales	joe	aa1234	-1	Normal	.Joe		Anderson
4	sales	frances	abcde12	50	Guest	Frances	Darla	Bryson
5	tech	sarah	password2	-1	Normal	Sarah		Arai

Once the spreadsheet is created, save it *using the CSV (comma separated values) format*. You can upload the file into DeskNow following the apposite link in the domain page (under Administration).

When you upload the file DeskNow will create all the accounts listed. Communities that do not exist will be automatically created. 'admin' will be the default manager, and they will have unlimited disk quota. User accounts that already exist will NOT be modified in any way.

3.1.11 Custom authentication

Custom authentication is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See section 16 for information on how to purchase a DeskNow license.

It is possible to implement a custom authentication mechanism for DeskNow accounts, for instance if your organization has already a centralized database or directory of employees, like Active Directory or LDAP, or an existing application.

DeskNow can also create DeskNow user accounts on demand, if an account exists in the external directory and a corresponding user account is not found in DeskNow.

Custom plugins can be written in Java, and interface with virtually every system and authentication logic. For more information, see docs/control/api/index.html in the DeskNow distribution.

Several plugins for integration with Active Directory and LDAP are provided by default.

See section 14.10.8 for more details on how to configure external authentication.

3.1.12 Self registration

Self registration is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See section 16 for information on how to purchase a DeskNow license.

It is possible to enable user self registration into DeskNow, i.e. to give users the possibility to subscribe and register their own account, community or domain. See section 10 for more details.

3.1.13 Domains and login

How does DeskNow distinguish between users of different domains when they are logging in?

Users of the default domain can login by simply using their username and password.

Users of other domains have different options:

1. they can login by typing *username@domainname* in the username field of the login page (ex. joe@company1.com)
2. if (as it should be if you want email to work properly for a domain) the login page can be reached by using the domain name in the internet address, users can simply use their username and DeskNow will automatically match the domain from the URL.

Example: your DeskNow login page is accessible at www.isp.com **and** www.company1.com (or abc.company1.com, it doesn't matter) because of DNS settings. If user 'joe' accesses the login page at www.isp.com and types 'joe' in the username field, DeskNow will verify his credentials against the user account 'joe' in the domain www.isp.com. If user 'joe' accesses the login page at www.company1.com (the page is the same, on the same DeskNow server: only the URL is different!) and types 'joe' in the username field, DeskNow will match his credentials against the user account 'joe' in the domain www.company1.com.

Note that DeskNow always finds the best match between the URL and the registered domains, i.e. it can distinguish between xxx.abc.org, yyy.abc.org and abc.org.

3.1.14 Domains and POP3, IMAP and SMTP authentication

Unfortunately the POP3, IMAP and SMTP protocols do not provide a standard way to handle per-domain authentication.

Users of the default domain can authenticate themselves simply using their username and password.

Users of other domains must authenticate themselves by using *username@domainname* as their POP3/SMTP username.

Ex.: joe@company1.com

3.1.15 Su-like login

The su-like login is a mean for the Administrator to login as any user, without knowing the appropriate password. This is equivalent to the 'su' command in Unix systems, hence the name.

This feature is very useful to perform particular tasks on an account, without involving the user directly and without changing passwords.

To login as user 'joe' for instance, the Administrator can simply use the following login:

username: %admin%joe

password: the Administrator's password

Administrators of virtual domains can control their own users by logging in with:

%admin%joe@company1.com

or (extended form):

%admin@company1.com%joe@company1.com

The super administrator (administrator of the default domain) can login as any user, of any domain, using:

%admin@defaultdomain.com%joe@company1.com

Where defaultdomain.com is the default domain.

3.1.16 Su-like login for community managers

It is possible to enable the su-like login also for community managers, so that they can login as users of their community (they cannot login as a user of a different community, however).

To enable this feature, set the parameter AllowSuLoginFromCommunityManager to TRUE in \$(DESKNOW_WEBAPP)/WEB-INF/cfg/Security.cfg.

If 'manager1' is manager of 'community1' and 'joe' is a member of 'community1', manager1 can login as joe by using the following login:

username: %manager1%joe

password: password for 'manager1'

3.1.17 Resetting the admin password

The admin user can reset the password of any other user in the system, including administrators of virtual domains.

To reset the forgotten password of the admin user, add the following line to the Security.cfg file:

```
ResetAdminPassword newpassword
```

And restart DeskNow. DeskNow will change the password for admin to 'newpassword', and automatically remove the above line from Security.cfg, for security.

3.1.18 Account expiration

Account expiration is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. A DeskNow ASP license is then required (not DeskNow Professional). See chapter 16 for information on how to purchase a DeskNow license.

It is possible to configure user accounts, communities and entire domains to expire and / or be deleted automatically at a certain date. This is most commonly associated with the Self-Registration (see chapter 11) mechanism, for trial accounts.

An account (but the same concept applies to communities and domains) can be set to expire automatically at a certain date. This can be set manually by editing the account's properties in the Administration web-interface, or is set automatically by the self-registration process for accounts that are created via self-registration.

When an account is expired, the user can no longer log in. However, the account will still receive mail normally. Typically after a certain period, at the date set for automatic deletion, the account will be deleted. This staged expiration approach ensures that the user has a last opportunity to fully subscribe to the service (ex. by paying a subscription fee) before the account is actually deleted. In addition, a notification message is sent to accounts a set number of days (configurable) before they expire.

When a community or a domain expires or is automatically deleted, all the enclosed accounts expire or are deleted.

To reenablen an expired account/community/domain, simply edit its properties in the Administration console and set its status to 'enabled'.

To disable automatic expiration/deletion of an account/community/domain, simply edit its properties and set to 'Not specified' its expiry/deletion date.

3.2 Importing accounts and emails from another mail server

It is possible to migrate user accounts and all their emails and mail folders to another IMAP mail server to DeskNow.

There are two different ways to do this.

The first method is to do the import in conjunction with external authentication. This is useful if your environment uses Active Directory or LDAP for authentication, and your old mail server relies on this as well. A typical example is Microsoft Exchange,

The second method is your environment does not use external authentication, but a standalone mail server.

3.2.1 Migrating from Exchange or other Active Directory or LDAP-based mail server

In this scenario, the reference source of information for accounts and passwords is the external directory, and the old mail server is queried only to import mail messages.

Simply configure external authentication with automatic account creation (see section 10). In addition, add/uncomment the following lines in Security.cfg:

```
ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Enable TRUE
ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Host
oldmailserver.domain.com
ExternalAuthentication.AutomaticAccountCreation.ImportIMAPMail.Port 143
```

The mail import will occur only at the first time the user successfully logs in (because only then DeskNow will know the password to use to login to the old mail server). The mail import process could take some time, so all mail will not be immediately visible to the user. In addition, to avoid swamping the old server with connections, mail import jobs are queued.

If using Active Directory, you can also import Exchange's mail aliases for every account. They are stored in Active Directory as attributes with this form:

```
proxyAddresses: SMTP:joe.smith@domain.com
```

To enable auto import of mail aliases, add/uncomment this line in the Security.cfg configuration file:

```
ExternalAuthentication.AutomaticAccountCreation.ImportADMailAliases TRUE
```

Important: before importing accounts, consider optimizing aliases by creating global mail aliases in the administration interface. If, for instance, your main domain is domain.com and every Exchange user has also an alias joe.smith@otherdomain.com, it's easier to just create a global mail alias like *@otherdomain.com -> *@domain.com in the administration interface. In this way you won't end up with many individual aliases.

DeskNow will not auto create an alias if the system already has an alias (including wildcards) or username that matches the address.

3.2.2 Migrating from a standalone mail server

In this scenario, there is no external directory to query for account authentication. The old mail server is the authoritative source for all the information. The import will be done automatically as the users log for the first time in DeskNow.

Here is how the process works:

1. leave the old mail server running alongside DeskNow (typically on different computers to avoid port conflict)
2. edit Security.cfg and add/uncomment these lines:

```
IMAPAutoImport.Enable TRUE
IMAPAutoImport.ExternalIMAPServer.Host old.mail.server
IMAPAutoImport.ExternalIMAPServer.Port 143
IMAPAutoImport.ExternalIMAPServer.SSL FALSE
IMAPAutoImport.DefaultDiskQuota 100
```

(replace old.mail.server with the address of your old mail server. Adjust port and use of SSL as required. Set the disk quota for newly created accounts in Mb, use -1 for unlimited. Make sure the quota is at least the disk quota used on the accounts on the mail server, otherwise not all mail messages will be able to fit in the new account)

3. restart DeskNow
4. point the MX record of your domain to DeskNow
5. simply tell your users to login to the DeskNow web interface, using the same username and password that they used with the old server
6. whenever a user logs in to DeskNow, if the account does not exist in DeskNow, DeskNow will connect to the old mail server and check that the username/password credentials are valid. If they are, DeskNow will automatically create the account and start importing mail from the old server.
7. once the user is logged in, the mail will not appear in his/her DeskNow account yet, as it is being imported from the mail server. Inform your users of this.
8. you can monitor the import progress by looking at the log.Admin log file
9. while the IMAP autoimport feature is enabled, whenever an email arrives for an account, if the account does not exist (yet) DeskNow will reply to the sending mail server to retry later. This ensures that no incoming mail is lost during the migration. Sending servers will not retry forever, however. It is recommended that all users login to their web mail during the morning, so that all accounts are quickly created
10. once all accounts are created and mail is imported, you can stop DeskNow and set IMAPAutoImport.Enable FALSE in the Security.cfg file. Then you can restart DeskNow and stop the old mail server for good

Note: it is also possible to import mail from an external mail server for accounts that already exist in DeskNow. See section 15.27 .

NB it is not possible to use this feature in conjunction with external authentication (Active Directory, LDAP, etc). When using IMAP autoimport, you must turn off external authentication. You can turn it on again once imap migration is completed and you have turned it off.

4 Mail

DeskNow is a full SMTP, IMAP4 and POP3 mail server.
DeskNow can receive and send emails on behalf of every domain registered with it.

Users have two ways of accessing and sending emails:

- using the web-based interface (webmail)
- using a traditional mail client (Outlook, Eudora, etc.)

DeskNow can also automatically retrieve emails on behalf of users from other POP3 servers.

DeskNow can be used as standalone mail server, or in conjunction with another existing mail server, leaving to it the SMTP functionalities and adding mail and webmail capabilities.

DeskNow can also be used as a closed system, i.e. for internal mail only, preventing users to send mail outside DeskNow.

4.1.1 DeskNow as MTA (Mail Transfer Agent)

In order for DeskNow to work as a full mail server that can receive and send mail from/to the Internet, the “embedded mail server” in DeskNow must be enabled (see the Configuration wizard or section 14.11.35).

You will also need to setup the MX records for every Internet domain that corresponds to a domain in DeskNow to point to your DeskNow server. Obtain help from your Internet Registrar on how to do this.

Finally, the TCP port 25 must be open between your DeskNow server and the Internet, in both directions. Check your firewalls to make sure this is the case.

4.1.2 DeskNow as SMTP server

DeskNow supports the standard SMTP Internet protocol to transfer emails. By default, DeskNow does not allow anyone (except from applications running on the computer where DeskNow is running, and users sending emails via the webmail interface) to send emails to the external world (“mail relay”). This is to prevent abuse from spammers.

You can authorize SMTP clients to send emails outside in two ways (they can co-exist):

- by setting a pool of “trusted” IP addresses (ex. your office LAN): every computer connecting to the DeskNow SMTP server from a trusted address will be able to send emails anywhere. This can be done using the Configuration Wizard, or by manually editing the server properties (see section 14.11.67)
- by using SMTP authentication: users must configure their mail client to send their DeskNow username and password before trying to send emails outside (see sections 14.11.69 and 14.11.73)

Note: users of virtual domains (i.e. not the default domain) must use *username@domainname* as their SMTP username. Ex.: joe@company1.com

4.1.3 SMTP Forwarding

DeskNow can be configured to forward all SMTP traffic to another SMTP server (ex. the server of your ISP) without attempting to deliver messages directly. This is useful in many

circumstances (ex. the DeskNow server is behind a firewall, you want to use additional functionalities provided by the external SMTP server such as filtering/logging, etc.).

See section 14.11.74 and following.

4.1.4 DeskNow as POP3 server

Besides webmail, users can retrieve emails from their DeskNow inbox using a traditional POP3 client.

DeskNow supports **POP3 synchronization with webmail**, in other words you can configure DeskNow so that when users download messages using POP3, the messages are automatically marked as “read” in the webmail and IMAP interface (this is the default for new installations). See section 14.11.80.

Note: users of virtual domains (i.e. not the default domain) must use *username@domainname* as their POP3 username. Ex.: joe@company1.com

4.1.5 Mailing lists

It is possible to define unlimited mailing list in every domain.

A mailing list is a list of email addresses (“members”, for instance joe@domain.com and paolo@hotmail.com), and there is a unique email address associated to the list itself (for instance partners@domain.com).

A list can contain an unlimited number of addresses. These can be both the addresses of DeskNow users, or email addresses in external domains.

When someone sends an email to the list’s address, the email is delivered to all the members of the list.

The domain administrator can create mailing lists, and define what addresses are members of the list. The administrator can also decide who is allowed to send emails to the list. It is possible to allow only members of the list to send to the list, or only some users, or both.

Individual users can also have personal mailing lists, i.e. groups of addresses that they can email to easily by picking a ‘contact folder’ from their contacts when composing an email. Personal mailing lists do not have an associated email address.

4.1.6 Using DeskNow to retrieve all emails for a company from a single POP3 inbox

It is possible to use DeskNow to retrieve all emails for a company from a single POP3 inbox, and distribute the emails to the various users. This is useful when emails for john@domain.com, sally@domain.com, etc all arrive in the single POP3 inbox of your ISP and you want to distribute them to the corresponding users john, sally, etc. in your DeskNow server.

The process can be summarized as follows:

- As admin user, create a setting for an External Account to retrieve emails from the external POP3 inbox
- As admin user, create one mail filter for each user in your organization. The filter should read like: “If To/Cc contains john@domain.com then forward copy to mail address john and delete the message”. To learn how to setup mail filters, see the User Help online

4.1.7 More configuration options

DeskNow is a very flexible mail server with many configuration parameters. See section 14.11 for more information.

4.2 Spam filtering

DeskNow offers several methods to fight spam:

4.2.1 DNSBL

DNSBL services (also known as RBL, although RBL is now a registered trademark) maintain a list of SMTP servers that are known to be used to send spam, or that are open relays. Many DNSBL services (free and commercial) are available on the net, often using different listing criteria.

When receiving a message (either through SMTP or POP3 from external accounts) DeskNow can query one or more DNSBL services to validate the IP address of the mail server that sent the message.

Messages coming from a blacklisted IP can be rejected immediately (if coming from SMTP), or classified as spam and stored in the user's 'Spam' folder, where the user can review them. Old messages are automatically removed from the Spam folder.

The use of DNSBL services can be configured by the Administrator via the Administration / Antispam configuration page. **Important:** make sure to understand how DNSBL works, and what listing criteria is used by the services you use. DeskNow by default uses DNSBL services provided by sorbs.net, but you can change this setting, or add other services.

Messages coming from a blacklisted IP address will contain this line in their headers:
X-DeskNow-Spam-Blacklisted-By: <address of DNSBL service that blacklisted the IP>

4.2.2 SURBL

SURBL is a free online service similar to DNSBL (see above). When this antispam method is enabled DeskNow scans the content of every incoming email. For every IP address or fully qualified host name found in the message, DeskNow will contact the SURBL service asking if it is a known web site used by spammers to promote their merchandise. The SURBL organization keeps a real-time database of such web sites.

This method is very effective, because it does not target the mail servers used to send spam, but the actual websites where people are directed to buy goods by spammers.

4.2.3 Bayesian analysis

DeskNow can analyse the subject and text of email messages and assign them an overall "spam probability", i.e. the likelihood that the message is spam. This probability is expressed as a number between 0 (not spam) and 1 (spam).

DeskNow will move messages whose spam probability is above a certain threshold level directly to the user's 'Spam' folder, where the user can review them. Old messages are automatically removed from the Spam folder.

Every user can decide what threshold level DeskNow should use for classifying messages directed to the account.

The computed spam probability of a message is added to the message headers, with this line:

X-DeskNow-Spam-Probability: <spam probability>

Important: statistical analysis is not an exact science (it provides a probability, not a certainty). The Bayesian filter in DeskNow can be **trained** to better detect spam. If a user receives a spam message in the Inbox (i.e. a message that was not detected as spam by the filter), he/she can press the 'This is spam' link to feed the message to the engine, which will analyse the message text and subject and adjust the engine parameters. Similarly, if a message is wrongly classified as spam, the user can press the 'This is not spam' link to teach the engine and force it to adjust its parameters.

The engine accuracy should improve with the training.

Note: the spam probability calculation is not performed if the message was already blacklisted via DNSBL, or cleared by the whitelist (see below).

4.2.4 Whitelists

The biggest problem in spam filtering technologies are *false positives*: messages that are classified as spam, but are not spam. To reduce this risk, DeskNow offers the possibility to create a list of people whose emails will never be classified as spam: a "whitelist".

Every user has his/her own whitelist (defined in the Preferences / Mail section). The whitelist can contain an email address or a domain address (in which case, all emails from that domain will be considered non-spam). A user can also add emails to his/her whitelist by simply selecting the 'This is not spam' link on a mail that is in the Spam folder.

All the email addresses stored in the user's Contacts (including subfolders) are automatically considered "whitelisted", so there is no need to manually add them to the whitelist. This also makes it easy for the Administrator to create a global whitelist, by simply creating a contact folder and sharing it with the users.

Messages that are considered "whitelisted" will contain the following line in their headers:

X-DeskNow-Sender-In-Whitelist: YES

4.2.5 Blacklists

Every user can define a blacklist using the mail filters. Note that blacklists are of little effect against most spam, since the email addresses used are often fake and continuously changed. Do not add thousands of email addresses to a blacklist, because the only likely effect will be to slow down the mail processing.

4.2.6 Greylisting

Greylisting is an anti-spam mechanism implemented at SMTP level. When a remote mail server connects to DeskNow SMTP to deliver an email to a local user, DeskNow will check the IP address of the remote server. If the IP address is not in DeskNow's "greylist", or has been there for less than a certain time, the delivery is rejected with a temporary error. If it was not in the greylist, the IP address is added to it.

The remote server will then try the delivery again, at a later time. At the second attempt, if the IP has been in the greylist for at least the 'minimum delay' time, the message is accepted. DeskNow will accept other messages from this server directly, for an amount of time called validity. After this validity, the IP address is removed again from the greylist.

In short, this mechanism forces unknown remote servers to retry the delivery after an interval. Many spammers do not bother retrying, so the mechanism is successful. Legitimate sender will only experience a short delay in delivering emails. DeskNow does not apply greylisting for mail coming from an address in the recipient's whitelist or contacts. Additionally, it is possible to configure DeskNow to not apply greylisting to certain IP addresses. Greylisting is disabled by default.

For more information on greylisting, see <http://www.greylisting.org> .

4.2.7 SPF (Sender Policy Framework)

DeskNow supports the checking of SPF records to determine if the sender address of an email has been forged (indicator that the message is probably spam). SPF is a protocol based on additional DNS records being created for every internet domain. For more information regarding SPF, visit <http://spf.pobox.com> .

4.2.8 Time limited email addresses

Professional spammers often obtain people's email addresses by using web-agents that browse the net and look for email addresses in web pages, message boards, newgroups, etc. The best way to avoid spam is then to avoid publishing one's email address. When it is necessary to do so, DeskNow offers *time-limited* addresses.

A time limited email address is an address in the form joe.040105@domain.com . If 'joe' is a valid account in the domain "domain.com", DeskNow will accept messages to that address only until January 5th, 2004 , delivering them to user 'joe'. In this way, even if the address ends up in a spammer's list, it will quickly become invalid. Time limited addresses do not require special configuration. The user can just input such addresses, whenever required, at will, simply deciding what the expiration date should be. DeskNow will automatically handle the expiration check and the forward to the actual address. Please note that time limited email addresses can only be used if DeskNow is the official MTA for your domain; it can't be used if the primary mail server for your domain is another server, because that server will likely not recognize the address. Remember the rule: username.YYMMDD@domain

4.2.9 Spam filtering in email clients

If you use external email clients (such as Outlook or Eudora) to access email, you can set up rules to filter messages that have been classified as spam by DeskNow.

Note that if DeskNow classifies a message as spam (either because of DNSBL or Bayesian analysis), it will add the following line to the headers:

X-Spam-Flag: YES

(Note that this is the same header used by SpamAssassin).

For instance, if you're using Microsoft Outlook, you can set up this rule using the Tools/Rules Wizard command:

"Apply this rule after the message arrives with X-Spam-Flag: YES in the message header delete it" . (You can also choose to move the message to another folder, etc).

Microsoft Outlook Express does not allow rules based on message headers. If your users use Outlook Express, you can configure DeskNow to add a flag to the message subject (see section 14.11.125).

The rule (set via the command Tools/Message Rules/Mail...) would then be:

“Apply this rule after the message arrives Where the Subject line contains '***SPAM***' Delete it” . (You can also choose to move the message to another folder, etc).

Note: it is also possible to setup DeskNow to not deliver messages classified as spam to POP3 clients. See section 14.11.120.

4.3 DeskNow as default mail client

On many systems it is possible to designate a particular application as default mail client. Whenever the user, in any application, clicks on an email address, the default mail client is usually launched to allow the composition of a new email.

DeskNow has a special login mode that, after verifying the user password, allows to compose an email addressed to a particular recipient. To invoke this special login/mail composition process, simply use the following URL as 'default mail application':
<http://yourdesknowserveraddress/desknow/index.html?CustomPage=MainMailTo.jsp&MailTo=%1>

4.3.1 Default mail client in Windows

Configuring Windows to use a particular mail client as default requires adding some keys to the Windows registry. A registry script for this purpose is included with DeskNow, and is available in the directory C:\Program Files\DeskNow\docs (/var/desknow/docs in Unix/Linux).

The file name is mailclient.reg. To use it, follow these steps:

- Open the file with a text editor (ex. Notepad)
- Edit the **two** lines where the URL of your DeskNow server is specified, and adjust according to the internet address/port of your server.
- Save the file
- Distribute the file to all the client PCs that you want to use DeskNow as default mail client.
- On every client PC, run the script by simply double-clicking on it. Make sure to be logged in as Windows Administrator.

Note: it is not possible to use DeskNow as mail client for the 'Send as attachment' command to attach files to an email directly from the desktop. Web browsers do not allow uploading files from the command line, for security reasons. Remember that DeskNow is not an application that runs on the client PC, and it cannot 'take' a file from the client PC unless the user manually uploads it.

4.4 Importing mail messages from other applications

It is possible to import mail messages from other applications (Microsoft Outlook, Microsoft Outlook Express, Eudora, Mozilla, and any 'mbox' file). Please consult the online user help for more information.

5 Server administration

5.1 Backups

DeskNow can perform automatic backups of all the relevant data, at scheduled times. DeskNow can also perform manual backups, on request of the Administrator. The backups are “online” i.e. they can happen during the normal operation of the server.

It is a good security practice to perform regular data backups, to prevent accidental data loss.

Currently DeskNow can backup the following data:

- User files: these are all the files stored in the **Files** section of the various users (except [DeskNow Logs] and [All server files]), and all the attachments
- Mail queues: these include all the messages being processed by the mail server (including messages quarantined, errors, and scheduled for delivery retry)
- User mail: the source files (.eml) of all the emails sent or received by every user (NB for emails sent via SMTP client, these are stored on the server only if the client uses IMAP and is configured to store sent emails on the server).
- Calendars: all the content of all calendars of every user (events and tasks)
- Contacts: all the contacts in every contact folder of every user
- Embedded database: the embedded McKoi database (for old versions of DeskNow), or the PostgreSQL database that is included in DeskNow for Windows. The database contains user settings, emails indexes, calendars, contacts, etc.

Note: If you use an external database instead of the embedded one, DeskNow cannot back it up for you. You can however read the instructions of your database to find out the best backup method, and schedule backups to happen roughly at the same time as the DeskNow backup.

Many backup options are configurable via the web administration page, or via the configuration files (see section 14.14.3). File and mail queues backup is NOT enabled by default, to avoid excessive and unexpected disk usage. You can enable the backup of these directories as well.

By default DeskNow creates the backup files inside a directory under the desknowdata\backup folder. We recommend to change this option to make DeskNow store the backups on a network drive running on a different server, to reduce the risk of data loss.

5.1.1 Restore

By backing up different elements in different places, DeskNow lets you to do a complete system restore (useful in case of system crash and total data loss), or to restore only individual elements (in case some data was accidentally deleted by a user).

DeskNow creates the following backup files:

- cfg.tar.gz (if enabled) contains the configuration files
- mail.tar.gz (if enabled) contains the mail delivery queues
- userfolders.tar.gz (if enabled) contains the user files and attachments
- usermail.tar.gz (if enabled) contains the user mail folders and emails

- o database.tar.gz (if enabled) contains the embedded PostgreSQL database

In case you need to restore a backup, refer to the following procedure:

- locate the backup folder corresponding to the date and time of the backup you want to restore (for instance:
C:\desknowdata\backup\DeskNowBackup_20031124_0925)
 - to restore configuration files:
 - o stop DeskNow
 - o uncompress cfg.tar.gz and replace the existing 'cfg' folder
 - to restore mail queues:
 - o stop DeskNow
 - o uncompress mail.tar.gz and replace the existing 'mail' folder
 - to restore all emails of all users:
 - o restore the database (see below)
 - o uncompress usermail.tar.gz and replace the existing 'usermail' folder
 - to restore user files:
 - o uncompress userfolders.tar.gz and replace the existing corresponding 'userfolders' folder
 - to restore calendars: to restore all calendars for all users, simply restore the database backup. calendars.tar.gz is useful only if you want to restore a particular calendar. In this case, simply uncompress the archive, locate the .ics file based on domainname\username\calendar name, and import it in DeskNow using the web interface: open the user account, the calendar, and select the 'Import' command from the folder menu.
 - to restore contacts: to restore all contacts for all users, simply restore the database backup. contacts.tar.gz is useful only if you want to restore a particular contact folder. In this case, simply uncompress the archive, locate the .csv file based on domainname\username\folder name, and import it in DeskNow using the web interface: open the user account, the folder, and select the 'Import' command from the folder menu.
 - to restore the embedded PostgreSQL database on Windows (default if you first installed DeskNow with version 3.1 or later):
 - o stop DeskNow
 - o uncompress the database.tar.gz archive and save the database.backup file to C:\temp (example)
 - o manually start the DeskNowDB service from the Control Panel
 - o open the Command Prompt
 - o type:

```
cd "C:\Program Files\DeskNow\pgsql\bin"
```



```
dropdb -U postgres -W desknow
```



```
createdb.exe -E UNICODE -O desknowserver -U postgres -W desknow
```



```
psql desknow desknowserver
```
 - o
 - o `vi C:/temp/database.backup`
- When prompted for a password, input 'pgpassword' without the quotes.
- o
 - o Note the forward slashes in the last command, instead of the usual back slashes.
- to restore an external database:

- stop DeskNow
- follow the restore instructions of your database

5.1.2 Disaster recovery

In case of total data loss, you can restore DeskNow to a fully working state using a nightly backup.

1. Install DeskNow (even a new version) on a stable system. Step through the configuration wizard using the appropriate values for your environment.
2. restore the database (see above)
3. restore the configuration files (see above)
4. restore the mail queues – if you have a backup (see above)
5. restore emails of all users (see above)
6. restore user files (see above)
7. ... calendars and contacts are already included in the database. You do not need to restore them individually

5.2 Antivirus scanning

Antivirus scanning is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See section 16 for information on how to purchase a DeskNow license.

DeskNow can integrate with most professional antivirus software to scan all incoming / outgoing emails for dangerous attachments. DeskNow can also request the virus scanning of all the files uploaded to the system, either via the web interface or via WebFolders.

To configure antivirus scanning, you need an antivirus that supports command line operation, i.e. it must provide a command that can be invoked to scan a single file only. Most commercial antivirus software allows this.

DeskNow has been tested with the following antivirus software:

- Norton Antivirus 2003/4
- Norton Antivirus Corporate Edition 7.x
- Symantec Antivirus Corporate Edition 8
- F-Secure Antivirus 2003/4
- McAfee Antivirus Command Line edition
- Grisoft Antivirus
- ClamAV antivirus (Linux)

Many other antivirus applications can be used without problems, as long as they support command line invocation. Also make sure to search our public forums at <http://www.desknow.com> , since many users have posted instructions for other antivirus applications.

To configure antivirus scanning:

1. login as admin user
2. click on Administration and then on Antivirus
3. enable antivirus scanning by checking the appropriate check box
4. select one of the predefined antivirus commands, and adjust the path of the command depending on where you installed the antivirus
5. if your antivirus software is not listed in the predefined commands, find out if it can be invoked by command line (most are), and what is the required syntax.

Typically the command line will be like "C:\Program Files\MyAntivirus\scan.exe"

followed by some options and the full path of the file to scan. Use %FILE% to specify that parameter.

For example, the command line for Norton Antivirus 2003 is the following:
"C:\Program Files\Norton AntiVirus\navw32.exe %FILE% /NORESULTS"

Every antivirus return a result code to indicate if the file was infected. The result code for a clean file is typically 0, but it is a good idea to double check with the documentation. You need to specify the result code for success in the apposite field.

Please inform us of the suitable command line for other antivirus products, so that we can add it to future releases to help other users.

6. When you press OK, DeskNow will try to invoke the antivirus to scan a clean file, and will expect to receive the success result code. It will display a message if an error occurred. In this case, check that the command line specified corresponds to your antivirus (it may have been installed in a different location) and retry.

Important: antivirus integration is available in DeskNow Lite only for a trial period. After this period is expired, files and attachments will not be scanned, and messages in both the Security and InternalError logs will be created, to warn you of the insecurity. You can then decide to purchase a DeskNow license, or disable antivirus scanning to avoid the warnings.

5.2.1 Configuration of Symantec/Norton Antivirus Corporate Edition

This popular antivirus does not include in the default installation a command line scanner. However a command line scanner utility is present in the installation CD.

Follow these steps to configure the antivirus to work with DeskNow:

1. Copy all the content of the directory navcorp\rollout\avserver\clients\dos on the CD 2 of NAVCE to the installation directory of your NAVCE (ex: c:\ProgramFiles\NAV). Do not replace existing files.
2. copy the file C:\Program Files\DeskNow\docs\dscan.bat to C:\ .
3. using Notepad, edit dscan.bat and if necessary adjust the path of the vscand.exe command (by default it is C:\Program Files\NAV), and the location of your desknowdata folder (by default it is c:\desknowdata)
4. open the Norton Antivirus Corporate Edition management interface, select Configure / File System Realtime Protection . Check 'Exclude selected files and folders' then click on 'Exclusions' and select the folder c:\desknowdata (adjust according to your installation) and C:\Program Files\NAV . Save the changes.
5. test a scan by opening a command prompt window, change directory to C:\ and type **dscan "C:\dscan.bat"**
6. configure DeskNow to use the pre-set settings for Norton/Symantec AV Corporate Edition

Important: the command line interface of NAVCE (vscand.exe) is a 16 bit application. It is slow to start because it loads a compatibility module to run in 32 bit Windows. The dscan.bat file is required to rename the file to be scanned (vscand.exe does not work with long file names). This solution is not recommended for sites with medium to high traffic, where many scan operations are needed. Symantec proposes the use of its Antivirus for SMTP Gateways in these circumstances. We also recommend evaluating other antivirus solutions from one of the other vendors supported.

Also note that vscand.exe scans files only of known dangerous extensions (.exe, .com, .doc, etc. and zip archives). If testing the setup using the EICAR virus test file, use the .com version rather than the .txt version.

5.3 Logging

DeskNow provides extensive logging information, with advanced features to filter and post-process log messages.

DeskNow generates log messages following system events. Each log message belongs to a category, like InternalError, SMTPIn, Debug, Security, and so on. It is possible to configure DeskNow to generate log messages only for some categories, to reduce the log verbosity. This is done through the Administration interface or the Lo.cfg configuration file. Note that some log categories, if enabled, can generate a *lot* of messages and affect the system's performance.

5.3.1 Log files

Log messages are typically output to stdout (the console), and to log files.

DeskNow generates one log file for each category, using names like log.Security.2005-02-15.txt and so on.

Log files are rotated daily, and by default removed if older than seven days. Log files are normally generated in \desknowdata\log, but the location can be changed in the Log.cfg configuration file. The log rotation period can be changed in the Administration interface.

5.3.2 Plugins to post-process log messages

Logging to external plugins is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See section 16 for information on how to purchase a DeskNow license.

DeskNow provides a plugin architecture to allow post processing of log messages. It is possible to write a Java class that will receive all log messages generated by DeskNow. This class can then use the information to store it on a different medium, collect statistics, and so on. For more information, see docs/control/api/index.html in the DeskNow distribution.

6 Audit logging

DeskNow can generate a detailed log of all the operations performed by users that change the data in the system. This is particularly useful in regulated environments, where a track of every operation performed is necessary for legal reasons.

Audit logging is disabled by default. It can be enabled via the Administration interface.

6.1.1 Where are the audit logs

Audit logs are separated from normal server logs, and kept in a separate folder, typically `desknowdata\logaudit`. Their position is configured in the `Log.cfg` configuration file.

Audit logs are separated per domain, year, month and day, to facilitate forensic analysis.

6.1.2 Format of audit logs

Every entry in an Audit log will take the format:

`[date and time][thread][username][ip address][channel]Message`

where:

- **date and time** is the date and time of the operation, in the server's timezone
- **thread** is the name of the internal server thread in which the operation was performed
- **username** is the full username ([user@domain.com](#)) of the user that performed the operation. If the account was accessed via a su-like login, the entry will read [user@domain.com](#) (su-login by [admin@domain.com](#)).
- **ip address** is the ip address from which the username connected to the server
- **channel** is the channel via which the operation was performed. Possible values are: web, smtp, pop3, imap, syncml, webdav, rss, ical

6.1.3 Audit logs for Instant Messaging

It is possible to enable audit logging of all the instant messaging activity on the server. Instant messaging audit logs will record every instant message sent by one user to another via the DeskNow server, whether using the integrated Messenger client or an external one.

6.1.4 Audit logs storage and maintenance

DeskNow does not delete audit log files, ever.

It is essential that the Administrator sets up appropriate policies to periodically backup audit logs and remove them from the DeskNow server, according to existing regulations, in order to avoid filling up all the disk space available on the server.

7 Using an external database

DeskNow can use an external SQL database to store user data. DeskNow for Windows comes with an embedded PostgreSQL database, which is perfect for most scenarios. Other databases (ex. MySQL, SQL Server) can be used. On Unix/Linux, it is necessary to use an external database.

DeskNow currently supports these popular RDBMS:

- Microsoft SQL Server 2000/2005 (including Express)
- MySQL version 4.0.12 or later
- PostgreSQL version 7.3 or later

7.1 The Database Initialization Wizard

The Database Initialization Wizard is a web-based wizard that will guide you in the process of configuring DeskNow to use an external database. It is normally not used in DeskNow for Windows as this version already includes a pre-configured database.

If DeskNow is not yet configured to use a database, the configuration wizard will automatically appear after you start DeskNow and access it with a web browser (ex. using the address <http://localhost/desknow/index.html>).

To use or re-run the initialization wizard, simply edit the configuration file Database.cfg, delete the lines for the following properties: JdbcDriver, Url, Username, Password and then restart DeskNow. Then access the wizard at <http://localhost/desknow/index.html> .

8 Using DeskNow with IIS or Apache

It is possible to configure DeskNow so that it can coexist with the most popular web servers on the same server.

8.1 Simple solution: using different ports

The easiest way to make DeskNow coexist with your web server is to configure it to use different port numbers for the HTTP and HTTPS service. The default ports are, respectively, 80 and 443; these ports are probably already used by your web server, if you have one. You can use the Configuration Wizard to tell DeskNow to use different ports (ex. 8080 and 4443). The login page will then be accessible at an address like `http://www.server.com:8080`

8.2 Apache 1.3.x and 2.x

There are two ways to integrate DeskNow and Apache on the same port. In both ways, Apache acts as 'front-end' handling all requests, and forwarding those that are specific to DeskNow to DeskNow itself.

8.2.1 Using mod_proxy

This is the easiest way to integrate Apache and DeskNow.

1. set DeskNow to use port 8080 and 8443 for http and https (this is done via the Configuration Wizard, which in Windows is an icon in the DeskNow program folder, and in Linux/Unix is launched with the command `/var/desknow/bin/config.sh`)
2. enable mod_proxy in Apache i.e. make sure that the following lines are uncommented (i.e. don't have a # character at the beginning) in `httpd.conf`:

LoadModule proxy_module modules/mod_proxy.so

If these lines are present in your configuration file, make sure to uncomment them as well:

LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so

If you are using Apache 1.3.x, make sure that this line is present as well:
AddModule mod_proxy.c

- These lines are usually already present in default installations.
3. add this section at the bottom of `httpd.conf`

ProxyRequests Off
ProxyPass /desknow http://localhost:8080/desknow/
ProxyPassReverse /desknow http://localhost:8080/desknow/

Important: do not use a different name for the proxied directory, like:

ProxyPass /webmail http://localhost:8080/desknow/

This will affect some of the pages. You need to rename the desknow directory first. See the online Knowledge Base for more details.

4. Restart Apache (in most systems you can also type 'apachectl graceful' to restart it without interrupting current operations)
5. Now you can access DeskNow at <http://www.myserver.com/desknow/index.html>
6. (optional) to see the correct IP addresses of clients in DeskNow logs, add this line to /var/desknowdata/cfg/Web.cfg :

ProxiedIPHeader X-Forwarded-For

and restart DeskNow. See section 14.13.15 for a more detailed explanation of this step.

7. (optional) if you need to enable DeskNow only in a virtual domain of Apache, put the Proxy... directives in a VirtualHost section in httpd.conf, instead than in the main section.

Note that instructing Apache to use port 8080 (non SSL) is fine even if your browsers connect to Apache with SSL! Using SSL between Apache and DeskNow is, most of the times, unnecessary, and only slows down the communication. Probably Apache and DeskNow run on the same computer, so SSL between them is unnecessary. SSL is important only between the browser and Apache.

Note for SyncML: some SyncML clients have issues with proxying. The best solution is to replace 'localhost' on the notes above with the full internet name of the server, as seen by the SyncML clients (ex. use www.mydomain.com instead of localhost). Then, to make sure Apache will always find the address of DeskNow, simply add to your /etc/hosts or C:\WINDOWS\SYSTEM32\DRIVERS\ETC\HOSTS file this line:
127.0.0.1 www.mydomain.com
Alternatively, you can use the mod_jk connector explained below.

8.2.2 Using mod_jk

The Apache foundation also provides custom modules to connect Apache to Tomcat (the application server included in DeskNow in the default installation).

These modules should be used in high load environments, since they provide better performance than the method that uses mod_proxy.

More information can be found at the following address:
<http://jakarta.apache.org/tomcat/connectors-doc/>

8.3 Microsoft IIS on Windows Server

The Windows version of DeskNow includes support for IIS. The following sections illustrate how to configure IIS in different scenarios.

1. set DeskNow to use ports different from 80 and 443 (since they are usually used by IIS). For instance, use 8080 and 8443. This configuration is done via the Configuration Wizard, which is an icon in the DeskNow program folder
2. open the Command prompt and type the following commands:

```
cd "C:\Program Files\DeskNow\bin"
```

cscript iis.js –s "Default Web Site"

Replace "Default Web Site" with the name of your web site, as it appears in the IIS console. Always use the double quotes.

NB this is not the Internet address of the web site, but simply the mnemonic name shown in the IIS console. The default web site is simply "Default Web Site".

3. *only for IIS 6.0 (Windows 2003 Server):*
 - a. *open the IIS Manager*
 - b. *right click on 'Web Sites' and select Properties*
 - c. *in the Service tab, select 'Run WWW service in IIS 5.0 isolation mode' and press OK*
 - d. *right click on 'Web Service Extension' and select 'Add a new web service extension'*
 - e. *Input 'jakarta' (all lowercase, no quotes) as Extension name, use the Add button to pick C:\Program Files\DeskNow\bin\isapi_redirect.dll, and select 'Set extension status to Allowed'. Press OK*
4. *for all versions:*
restart IIS

Once this is done, you should be able to access DeskNow at <http://www.domain.com/desknw/index.html> (if www.domain.com is the internet address of your web site).

Tip: if you get this error when running the script: "Unable to find the IISFilters for ...", try manually creating a test filter in IIS, then run the script again.

9 Using your own SSL certificate

SSL is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See chapter 16 for information on how to purchase a DeskNow license.

This section is useful only if you access DeskNow directly. If you access DeskNow through a web server like Microsoft IIS or Apache, you need to configure SSL support in the web server, not in DeskNow.

When DeskNow is installed, it comes with a built-in test SSL certificate. This certificate is not signed by a Certification Authority normally trusted by web browsers (ex. Verisign, Thawte, etc.). This usually means that although SSL is enabled and communications using the HTTPS protocol are encrypted, your browser may display alert messages to inform you that the certificate is not fully trusted.

9.1.1 To simply stop browser alerts without changing certificates

These instructions apply to Microsoft Internet Explorer. The procedure for other browsers is usually very similar.

When you open DeskNow using HTTPS, the browser will popup an alert message. Click on "View certificate", then click on "Install Certificate".

9.1.2 If you're upgrading from a release before 2.7

SSL certificate management has been simplified since the 2.7 release, to make it easier to maintain your SSL certs when upgrading DeskNow. If you're upgrading DeskNow from an earlier release, you will need to download 2 extra files to your \desknowdata\ssl folder.

- <http://www.desknow.com/desknow/directfiles/desknow/downloads/sslupgrade/httpkeystore> (all OS)
- <http://www.desknow.com/desknow/directfiles/desknow/downloads/sslupgrade/updateHttpKeystore.bat> (Windows)
or
<http://www.desknow.com/desknow/directfiles/desknow/downloads/sslupgrade/updateHttpKeystore.sh> (Linux, Unix, Mac OSX)

9.1.3 To install a certificate signed by a Certificate Authority (Windows)

If you're using DeskNow on Windows, to install your own SSL certificate, signed by a trusted Certification Authority, follow these steps:

1. open a Command Prompt window and type the following commands in **bold**. NB if you copy and paste them, make sure that each command is on *one line*. Paste in Notepad first if unsure.
2. change the current directory to c:\desknowdata\ssl (adjust if you installed DeskNow in a different location): **cd "c:\desknowdata\ssl"**
3. delete the existing temporary certificate: **del httpkeystore**
4. create your own untrusted certificate:

"C:\Program Files\DeskNow\java\bin\keytool" -genkey -validity 730 -keystore httpkeystore -alias tomcat -keyalg RSA -storepass changeit

(note: if you want to set a password different from 'changeit', you will have to edit server.xml)

5. IMPORTANT: when prompted for your first and last name, *type the address of your server, for example: www.myhost.com*
6. answer all the following questions
7. when requested for a key password, press ENTER
8. IMPORTANT: make a backup of your httpkeystore file just generated. This contains your private key. Losing this file from now on means losing the certificate.
9. generate a certificate signing request:

"C:\Program Files\DeskNow\java\bin\keytool" -certreq -alias tomcat -file mycert.csr -keystore httpkeystore -storepass changeit

10. using the generated mycert.csr file, request a trusted certificate from a Certification Authority (es [Verisign](#), [Thawte](#)). Follow their instructions on how to submit the csr.
11. the Certification Authority will return you a trusted certificate, in the form of a www.mydomain.com.crt file (the file name could change, but the extension should be .crt). This process could take a few days.
12. *If your certification authority is GoDaddy*, you will need to import an extra certificate first. This is the gd_bundle.crt certificate that is included in their reply email. To import this extra cert:

"C:\Program Files\DeskNow\java\bin\keytool" -import -alias root -file gd_bundle.crt -keystore httpkeystore -storepass changeit

13. import the trusted certificate in your keystore:

"C:\Program Files\DeskNow\java\bin\keytool" -import -alias tomcat -file www.mydomain.com.crt -keystore httpkeystore -storepass changeit -trustcacerts

14. update DeskNow's work keystore: **updateHttpKeystore.bat**
15. stop and restart DeskNow.
16. important: when you upgrade DeskNow, make sure to run **updateHttpKeystore.bat** again

9.1.4 To install a self-signed certificate (Windows)

If the default self-signed certificate is not suitable for you, you can create a new self-signed certificate.

1. open a Command Prompt window and type the following commands in **bold**. NB if you copy and paste them, make sure that each command is on *one line*. Paste in Notepad first if unsure.
2. change the current directory to c:\desknnowdata\ssl (adjust if you installed DeskNow in a different location): **cd "c:\desknnowdata\ssl"**
3. type the following command:

"C:\Program Files\DeskNow\java\bin\keytool" -selfcert -validity 730 -keystore httpkeystore -alias tomcat -keyalg RSA -storepass changeit -dname "cn=www.mydomain.com, ou=Administration, o=My Company,

c=UK"

(use appropriate values for domain, organizational unit, company name and country code)

4. update DeskNow's work keystore: **updateHttpKeystore.bat**
5. stop and restart DeskNow.
6. important: when you upgrade DeskNow, make sure to run **updateHttpKeystore.bat** again

9.1.5 To install a certificate signed by a Certificate Authority (Unix/Linux/Mac OSX)

If you're using DeskNow on Unix or Linux, to install your own SSL certificate, signed by a trusted Certification Authority, follow these steps. Please adjust according to the path of your Java installation.

1. open a console window and type the following commands in **bold**. NB if you copy and paste them, make sure that each command is on *one line*. Paste in a text editor first if unsure.
2. become superuser: **su -**
3. **cd /var/desknowdata/ssl**
4. delete the existing temporary certificate: **rm httpkeystore**
5. create your own untrusted certificate:

/usr/java/j2sdk1.4.2/bin/keytool -genkey -validity 730 -keystore httpkeystore -alias tomcat -keyalg RSA -storepass changeit

(note: if you want to set a password different from 'changeit', you will have to edit server.xml)

6. IMPORTANT: when prompted for your first and last name, *type the address of your server, for example: www.myhost.com*
7. answer all the following questions
8. when requested for a key password, press ENTER
9. IMPORTANT: make a backup of your httpkeystore file just generated. This contains your private key. Losing this file from now on means losing the certificate.
10. generate a certificate signing request:

/usr/java/j2sdk1.4.2/bin/keytool -certreq -alias tomcat -file mycert.csr -keystore httpkeystore -storepass changeit

11. using the generated mycert.csr file, request a trusted certificate from a Certification Authority (ex. [Verisign](#), [Thawte](#)). Follow their instructions on how to submit the csr.
12. the Certification Authority will return you a trusted certificate, in the form of a www.mydomain.com.crt file (the file name could change, but the extension should be .crt). This process could take a few days.
13. *If your certification authority is GoDaddy*, you will need to import an extra certificate first. This is the gd_bundle.crt certificate that is included in their reply email. To import this extra cert:

/usr/java/j2sdk1.4.2/bin/keytool -import -alias root -file gd_bundle.crt -keystore httpkeystore -storepass changeit

14. import the trusted certificate in your keystore:

```
/usr/java/j2sdk1.4.2/bin/keytool -import -alias tomcat -file  
www.mydomain.com.crt -keystore httpkeystore -storepass changeit -  
trustcacerts
```

15. update DeskNow's work keystore: **./updateHttpKeystore.sh**
16. stop and restart DeskNow.
17. important: when you upgrade DeskNow, make sure to run **updateHttpKeystore.sh** again

9.1.6 To install a self-signed certificate (Unix/Linux/ Mac OSX)

If the default self-signed certificate is not suitable for you, you can create a new self-signed certificate. Please adjust the commands according to the path of your Java installation.

1. open a console window and type the following commands in **bold**. NB if you copy and paste them, make sure that each command is on *one line*. Paste in a text editor first if unsure.
2. become superuser: **su -**
3. **cd /var/desknowdata/ssl**
4. type the following command:

```
/usr/java/j2sdk1.4.2/bin/keytool -selfcert -keystore httpkeystore -alias  
tomcat -storepass changeit -dname "cn=www.mydomain.com,  
ou=Administration, o=My Company, c=UK"
```

(use appropriate values for domain, organizational unit, company name and country code)

5. update DeskNow's work keystore: **./updateHttpKeystore.sh**
6. stop and restart DeskNow.
7. important: when you upgrade DeskNow, make sure to run **updateHttpKeystore.sh** again

9.1.7 SSL for SMTP, POP3 and IMAP

DeskNow can accept SSL encrypted SMTP, POP3 and IMAP connections. The SSL certificate used for POP3 operations is located in C:\desknowdata\ssl\pop3keystore (/var/desknowdata/ssl/pop3keystore), and is a self signed certificate. The certificate for the SMTP protocol is in the file smtpkeystore (same location). The certificate for the IMAP protocol is in the file imapkeystore.

Most mail clients, exactly like a browser, display a security warning when connecting to a mail server with a self-signed certificate (note: Microsoft Outlook and Outlook Express do not seem to display warnings for SMTP, only for POP3) . To use your own, trusted certificate, simply create or import your own HTTP SSL keystore following the instructions in the previous sections. Then copy it over the smtpkeystore, pop3keystore and imapkeystore files. This assumes that the server name you use for the SMTP, POP3 and IMAP services is the same as the name you use for http (ex. www.domain.com).

You can also store completely different SSL certificates in the store (following the same instructions for the HTTP keystore – but use the name 'pop3keystore', 'imapkeystore' or 'smtpkeystore' instead of 'httpkeystore' and do not give the updateHttpKeystore command). This is useful for instance if you want the web certificate to be in the name of www.domain.com , the POP3 certificate to be in the name of pop3.domain.com, and the SMTP certificate in the name of smtp.domain.com .

9.1.8 SSL for virtual domains

Due to the nature of the HTTPS protocol, it is generally *not* possible to have multiple SSL certificates for virtual domains associated to the same IP address. There can be only one SSL certificate per IP address. This constraint is not specific to DeskNow, but applies to every web server.

For full explanation and configuration details, please refer to our online knowledge base.

10 External authentication

External authentication is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See chapter 16 for information on how to purchase a DeskNow license.

DeskNow can check authentication credential of users against external repositories, like LDAP directories, ActiveDirectory, Databases, etc.

DeskNow can also automatically create accounts that do not exist in DeskNow but exist in the external directory, as soon as they're needed.

For security reasons, the admin user of each domain cannot be authenticated with the external authentication process. This allows server management even in the case the connection with the service providing the external authentication is unavailable.

When you enable external authentication for a system where some user accounts already exist, make sure that the external repository knows about these accounts, otherwise the accounts will be inaccessible. As mentioned above, this is not necessary for administrator users, because those accounts remain independent from external authentication.

The external authentication uses a plugin that can perform any operation it is necessary to provide the authentication functionality. The plugin consists of one or more java classes.

Default plugins are included in DeskNow to authenticate against an ActiveDirectory, against ActiveDirectory using Kerberos secure authentication, and against a generic Kerberos KDC.

DeskNow can cache positive results from external authentication directories. This typically reduces network traffic and increases performance, but can cause delays in propagating changes when an account's password is changed in the external directory. See section 14.10.9 for more details.

10.1 ActiveDirectory authentication

With AD authentication, DeskNow can authenticate users of every domain against an AD server. The user accounts must each belong to a domain that corresponds to the virtual domain setup in DeskNow.

To configure DeskNow to use Active Directory for external authentication, see section 14.10.8 .

A classic configuration setup in the Security.cfg file would be:

```
ExternalAuthentication.Enable TRUE
ExternalAuthenticationAutomaticAccountCreation.Enable TRUE
ExternalAuthentication.AuthenticationProvider
com.desknow.control.authentication.impl.ActiveDirectoryAuthenticationProvider
ExternalAuthenticationCustomProperties.ServerAddress adserver.domain.com
ExternalAuthenticationCustomProperties.ServerPort 389
ExternalAuthenticationCustomProperties.BrowserUsername Administrator@domain.com
ExternalAuthenticationCustomProperties.BrowserPassword password
```

If your AD tree is more complex than most cases, you may specify additional parameters to tell DeskNow where in the AD tree are the accounts for a domain, using CustomQuery parameters.

The simplest form is if you want to create all accounts in the same community:

```
ExternalAuthentication.CustomProperties.SearchSubtrees TRUE
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community Default
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch "DC=mydomain,DC=com"
```

This will enable autocreation of every account which has a userPrincipalName attribute (the default for AD accounts) in any AD branch of the mydomain.com, ex. "OU=Tech,DC=mydomain,DC=com" or "OU=Sales,DC=mydomain,DC=com".

But it is possible to be more specific, and for instance enable auto creation only for users in the "OU=Tech,DC=mydomain,DC=com" branch – and not in "OU=Sales,DC=mydomain,DC=com", and not even in sub branches like "OU=Database,OU=Tech,DC=mydomain,DC=com":

```
ExternalAuthentication.CustomProperties.SearchSubtrees FALSE
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community Default
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch
"OU=TechDC=mydomain,DC=com"
```

(Please note that in the above examples the Branch parameter must be on one line. It goes to a new line in this document for formatting reasons).

You can also create accounts in different communities for different AD branches:

```
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community Sydney
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch
"OU=Sydney,DC=mydomain,DC=com"
```

```
ExternalAuthentication.CustomProperties.CustomQuery.2.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Community Melbourne
ExternalAuthentication.CustomProperties.CustomQuery.2.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Branch
"OU=Melbourne,DC=mydomain,DC=com"
```

Finally, you can merge 2 AD domains into one DeskNow domain:

```
ExternalAuthentication.CustomProperties.SearchSubtrees TRUE
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community Default
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain addomain1.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch "DC=addomain1,DC=com"
```

```
ExternalAuthentication.CustomProperties.CustomQuery.2.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Community Default
ExternalAuthentication.CustomProperties.CustomQuery.2.RemoteDomain addomain2.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Branch "DC=addomain2,DC=com"
```

Every CustomQuery parameter has 4 mandatory sub-parameters:

- LocalDomain is the DeskNow domain
- Community is the community in which an account will be created if not existing
- RemoteDomain the AD domain in which to lookup the account
- Branch the AD branch in which the account information is located (see below for more information on how to find the AD branches in your domain)

Note that the DeskNow domain name (LocalDomain) does not have to be the same as the AD domain (RemoteDomain).

To find out where in your AD tree are the accounts, use the following command in the AD server (you must have the permission to browse the AD):

```
ldifde -f ldif.txt
```

The resulting ldif.txt file will list every entry with its full location. To quickly search for the entry corresponding to an user, search for username@domain in the text file.

You can also enable autocreation only for accounts that contain a particular LDAP attribute. For instance, to only allow members of the 'OkToEmail' builtin group:

```
ExternalAuthentication.CustomProperties.CustomQuery.1.AttributeFilters.1.Name  
"memberOf"  
ExternalAuthentication.CustomProperties.CustomQuery.1.AttributeFilters.1.Value  
"cn=OkToEmail,ou=Groups,o=MyCompany"
```

Since AttributeFilters are specified for each CustomQuery, you can use them to tell DeskNow in which domain and community to create the account, based on the presence of the attribute.

If you specify multiple AttributeFilters for the same CustomQuery, they will be considered in *logical or*, i.e. it is enough that one matches.

If AutomaticAccountCreation is enabled, by default accounts are created in the default community.

You can specify to use a different community:

```
ExternalAuthentication.AutomaticAccountCreation.Community "Another community"
```

However this setting will be overridden by the community set in any CustomQuery setting.

By default every account will be created in the 'Normal' user class. To create new accounts in a different class, use:

```
ExternalAuthentication.AutomaticAccountCreation.UserClass "Other user class"
```

You can also define what disk quota will be given to new accounts:

```
ExternalAuthentication.AutomaticAccountCreation.DefaultDiskQuota 100
```

The value is expressed in Mb. Use -1 for unlimited disk quota (the default).

10.2 ActiveDirectory with Kerberos authentication

With AD with Kerberos authentication, DeskNow can authenticate users of every domain against an AD server. The user accounts must each belong to a domain that corresponds to the virtual domain setup in DeskNow.

To configure DeskNow to use Active Directory for external authentication, see section 14.10.8 .

A classic configuration setup in the Security.cfg file would be:


```
ExternalAuthentication.Enable TRUE
ExternalAuthenticationAutomaticAccountCreation.Enable TRUE
ExternalAuthentication.AuthenticationProvider
com.desknow.control.authentication.impl.ActiveDirectoryWithKerberosAuthenticationP
rovider
ExternalAuthentication.CustomProperties.ServerAddress adserver.domain.com
ExternalAuthentication.CustomProperties.ServerPort 389
ExternalAuthentication.CustomProperties.BrowserUsername Administrator@domain.com
ExternalAuthentication.CustomProperties.BrowserPassword password
ExternalAuthentication.CustomProperties.KRB5ConfFile
$(DESKNOWDATA)\kerberos\krb5.conf
ExternalAuthentication.CustomProperties.JAASConfFile
$(DESKNOWDATA)\kerberos\desknow_jaas.conf
```

Additionally, edit `\desknowdata\kerberos\krb5.conf` and replace `MYDOMAIN.COM` with your domain name, and `server.mydomain.com` with the address of your AD server (it can also be `localhost`). You can add multiple domain names-realms. Important: the domain name here must be all in uppercase letters. If you don't have `\desknowdata\kerberos` (because you first installed an older version of DeskNow), please contact our Support staff.

Finally, make sure that the account `Administrator@domain.com` (or whatever account you use to browse AD) is set to use DES encryption to store the password. This is done in Windows Administration Tools/Active Directory users and computers/<account>/Properties/Account and checking the option 'Use DES encryption types for this account'. If this was unchecked, you need to reset the account's password to force Windows to use DES to store it.

10.3 Kerberos authentication

This authentication plugin can check user credentials against a standard Kerberos KDC. Note that it is NOT possible to automatically create accounts using this plugin.

A classic configuration setup in the `Security.cfg` file would be:

```
ExternalAuthentication.Enable TRUE
ExternalAuthenticationAutomaticAccountCreation.Enable FALSE
ExternalAuthentication.AuthenticationProvider
com.desknow.control.authentication.impl.KerberosAuthenticationProvider
ExternalAuthentication.CustomProperties.KRB5ConfFile
$(DESKNOWDATA)\kerberos\krb5.conf
ExternalAuthentication.CustomProperties.JAASConfFile
$(DESKNOWDATA)\kerberos\desknow_jaas.conf
```

Additionally, edit `\desknowdata\kerberos\krb5.conf` and replace `MYDOMAIN.COM` with your domain name, and `server.mydomain.com` with the address of your KDC server (it can also be `localhost`). You can add multiple domain names-realms. Important: the domain name here must be all in uppercase letters. If you don't have `\desknowdata\kerberos` (because you first installed an older version of DeskNow), please contact our Support staff.

10.4 LDAP authentication

With LDAP authentication, DeskNow can authenticate users of every domain against an LDAP server. The user accounts must each belong to a domain that corresponds to the virtual domain setup in DeskNow.

To configure DeskNow to use LDAP for external authentication, see section 14.10.8 .

A classic configuration setup in the Security.cfg file would be:

```
ExternalAuthentication.Enable TRUE
ExternalAuthenticationAutomaticAccountCreation.Enable TRUE
ExternalAuthentication.AuthenticationProvider
com.desknow.control.authentication.impl.LdapAuthenticationProvider
ExternalAuthentication.CustomProperties.ServerAddress ldapserver.domain.com
ExternalAuthentication.CustomProperties.ServerPort 389
ExternalAuthentication.CustomProperties.BrowserUsername
"cn=Manager,dc=mydomain,dc=com"
ExternalAuthentication.CustomProperties.BrowserPassword password
```

If your LDAP tree is more complex than most cases, you may specify additional parameters to tell DeskNow where in the LDAP tree are the accounts for a domain, using CustomQuery parameters.

For instance, if some user accounts for the domain mydomain.com are to be found under the "OU=Sydney,DC=mydomain,DC=com" branch and some other in the "OU=Melbourne,DC=mydomain,DC=com" branch (LDAP users registered under the mydomain.com domain) you can add these lines:

```
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community community
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch
"OU=Sydney,DC=mydomain,DC=com"
```

```
ExternalAuthentication.CustomProperties.CustomQuery.2.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Community community
ExternalAuthentication.CustomProperties.CustomQuery.2.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.2.Branch
"OU=Melbourne,DC=mydomain,DC=com"
```

Please note that in the above examples the Branch parameter must be on one line. It goes to a new line in this document for formatting reasons.

Every CustomQuery parameter has 4 sub-parameters:

- LocalDomain is the DeskNow domain
- Community is the community in which an account will be created if not existing
- RemoteDomain the LDAP domain in which to lookup the account
- Branch the LDAP branch in which the account information is located

Note that the DeskNow domain name (LocalDomain) does not have to be the same as the LDAP domain (RemoteDomain).

You can also specify different settings for every domain, and even map 2 LDAP domains to a single DeskNow domain:

```
ExternalAuthentication.CustomProperties.CustomQuery.3.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.3.Community hkcommunity
ExternalAuthentication.CustomProperties.CustomQuery.3.RemoteDomain domain2.com
ExternalAuthentication.CustomProperties.CustomQuery.3.Branch "OU=Hong
Kong,DC=domain2,DC=com"
```

DeskNow assumes that the attribute that contains the username is 'uid'. If the username is stored in a different attribute, you can add this line:

```
ExternalAuthentication.CustomProperties.UIDAttribute username
```

If AutomaticAccountCreation is enabled, by default accounts are created in the default community.

You can specify to use a different community:

```
ExternalAuthentication.AutomaticAccountCreation.Community "Another community"
```

However this setting will be overridden by the community set in any CustomDomainQuery setting.

In addition, you can also specify the parameter

```
ExternalAuthentication.CustomProperties.SearchSubtrees TRUE
```

to indicate that the search for an account should be carried over any sub-branch.

So you could have:

```
ExternalAuthentication.CustomProperties.SearchSubtrees TRUE
ExternalAuthentication.CustomProperties.CustomQuery.1.LocalDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Community community
ExternalAuthentication.CustomProperties.CustomQuery.1.RemoteDomain mydomain.com
ExternalAuthentication.CustomProperties.CustomQuery.1.Branch "DC=mydomain,DC=com"
```

And with this catch both users in the "OU=Sydney,DC=mydomain,DC=com" and "OU=Melbourne,DC=mydomain,DC=com" branches.

You can also enable autocreation only for accounts that contain a particular LDAP attribute. For instance, to only allow members of the 'OkToEmail' builtin group:

```
ExternalAuthentication.CustomProperties.CustomQuery.1.AttributeFilters.1.Name
"memberOf"
ExternalAuthentication.CustomProperties.CustomQuery.1.AttributeFilters.1.Value
"cn=OkToEmail,ou=Groups,o=MyCompany"
```

Since AttributeFilters are specified for each CustomQuery, you can use them to tell DeskNow in which domain and community to create the account, based on the presence of the attribute.

If you specify multiple AttributeFilters for the same CustomQuery, they will be considered in *logical or*, i.e. it is enough that one matches.

By default every account will be created in the 'Normal' user class. To create new accounts in a different class, use

```
ExternalAuthentication.AutomaticAccountCreation.UserClass "Other user class"
```

You can also define what disk quota will be given to new accounts:

```
ExternalAuthentication.AutomaticAccountCreation.DefaultDiskQuota 100
```

The value is expressed in Mb. Use -1 for unlimited disk quota (the default).

To determine the full name of the person, by default DeskNow uses the CN attribute, and assumes that the first word is the first name, and the last word is the last name. If your directory has specific attributes to hold the first and last name of a person, you can use these lines:

```
ExternalAuthentication.CustomProperties.UseCNAttribute FALSE
ExternalAuthentication.CustomProperties.GNAttribute givenName
ExternalAuthentication.CustomProperties.SNAttribute sn
```

DeskNow can also automatically create mail aliases from the LDAP data. For instance, if the LDAP account contains the following attribute:

```
mail: alias@domain.com
```

```
add:
```

ExternalAuthentication.CustomProperties.MailAliasAttribute mail

If the account contains:
mail: smtp:alias@domain.com

add:
ExternalAuthentication.CustomProperties.MailAliasAttribute mail
ExternalAuthentication.CustomProperties.MailAliasAttributePrefix "smtp:"

DeskNow will import any aliases found in the LDAP data, unless the alias already exists in the system.

10.5 Custom authentication

It is possible to implement your own authentication mechanism. For more information, please see the control/api directory in the /docs folder of the DeskNow distribution.

10.6 Switching back to internal authentication

If you decide to switch off external authentication, and rely only on DeskNow's own authentication, the default password for the accounts that were previously under external control is set to "DeskNowExternalPassword". Make sure this password is changed as soon as possible.

11 Self registration

Self registration is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. A DeskNow ASP license is then required (*not* DeskNow Professional). See chapter 16 for information on how to purchase a DeskNow license.

It is possible to enable user self registration into DeskNow, i.e. to give users to subscribe and register their own account. Additionally, it is possible to configure self-registered accounts to expire after a certain period, unless they are converted in full accounts by the administrator.

11.1 Types of self registration

There are three forms of self registration available:

- **User self registration:** this allows a person to register an user account in DeskNow. You predetermine the domain and community for self-registered users, their disk quota and their user class.
- **Community self registration:** this allows a person to register a community and an user account in DeskNow. The created user account will be the manager of the community, and it will be able to create new users in that community. You predetermine the user's disk quota, the community disk quota, and the user's class.
- **Domain self registration:** this allows a person to register a new virtual domain. An admin account for that domain will be automatically created. You predetermine the domain's disk quota. As admin of the new domain, the person will be able to create new accounts in the domain.

11.2 Configuration

All the aspects of Self registration are configured via the SelfRegistration.cfg configuration file. See section 14.16 for more details.

11.3 Personalization

The self registration process involves the following steps:

- A person visits a web page with a form where he/she can input the necessary self registration information. This is the self-registration form, in the self-registration page.
- When the person presses the submit button in the form, the form posts the data to DeskNow, which processes the request.
- If any error occurs (ex.: the user has chosen an username already in use), DeskNow will output an error page describing the problem, with a 'Back' button that will bring back to the self-registration page.
- If, on the other hand, the self-registration is successful, a success page is displayed, containing the self-registration confirmation, and a link to the login page.

11.3.1 Personalization of the self-registration pages

You can customize the self-registration page to suit your needs. You can actually embed the form in any page of your website.

An example of self registration pages for the three different types of self-registration (user, community and domain) is provided in the folder C:\Program Files\DeskNow\desknow\jsp\original\selfregistrationsample (/var/desknow/desknow/jsp/original/selfregistrationsample in Unix/Linux).

11.3.2 Personalization of the confirmation pages

The confirmation pages are dynamic JSP files, because they typically contain dynamic data (ex. the username generated).

These files are the following:

SelfRegisterUserSuccess.jsp is the confirmation page for user self-registration

SelfRegisterCommunitySuccess.jsp is the confirmation page for community self-registration

SelfRegisterDomainSuccess.jsp is the confirmation page for domain self-registration

These files are located in C:\Program Files\DeskNow\desknow\jsp\original ((/var/desknow/desknow/jsp/original). You can customize them to suit your needs, by simply editing them with a text editor or an HTML editor. Pay attention to the dynamic tags, delimited by <% and %>.

12 Language and timezone

DeskNow is a multi-language product, which means that the web interface can be accessed in many languages. This section explains how to configure DeskNow for multi-language access, and how to translate DeskNow to your language, if a translation is not already available.

12.1 Default language and user language

The languages support of DeskNow is configured in the configuration file `International.cfg` (see section 12.2).

For every language *besides the pre-installed ones*, two lines must be present in the file, defining the language code, name and the location of the translation file.

```
Example:
Language.it.name      Italian
Language.it.file     $(LANGUAGE_BASE)/desknow_it.txt
```

In the above example, a new language is defined, where:

- “it” is the language code
- “Italian” is the language name
- the file `desknow_it.txt`, in the specified folder, is the translation file

The translation file is a text file where there is one key translation per line.

```
Example:
New contact = Nuovo contatto
```

You can configure the default language for DeskNow. This is the language that will be used by default for new users. The setting is described in section 14.14.1.

Users can choose to use a different language by changing the appropriate setting in the Preferences page. Every user can choose his/her own language.

Note: you may want to check the emails sent are encoded using the appropriate charset. See section 14.11.84 for more information about setting the server’s charset encoding for outgoing email.

12.2 Adding a language to DeskNow

If your language is not available in the list of available languages (because it is not configured), visit the support page of DeskNow on the web to look for additional languages (see section 17).

If still no one has produced a translation file for your language, you can create one yourself, and share it with the community of DeskNow users. Or collaborate with other users that are looking for such translation, and share the work. Our public forums have a forum dedicated to the exchange of information regarding language translations for DeskNow.

IMPORTANT: before starting a translation, please announce it on the public forums, so that others can see that you're working on a translation for your language, and can avoid duplicate work. In the same spirit, check the forums to see if someone is already working on a translation in your language.

To create a language for DeskNow, look for the ISO 639 language code of your language. This information can be found on the web, ex. at <http://www.ics.uci.edu/pub/ietf/http/related/iso639.txt>

It is important to use an official language code, because DeskNow will use the code to translate dates accordingly.

Then copy the file `desknow_TEMPLATE.txt` (located in Program Files\DeskNow\desknow\WEB-INF\international) to the language file that you want to create (ex. `desknow_fr.txt` for French).

Then add the configuration for your language to `International.cfg`.

Example:
`Language.fr.name` `French`
`Language.fr.file` `$(LANGUAGE_BASE)/desknow_fr.txt`

Then open the file using Windows Notepad or any other **UTF-8** editor. It is mandatory to save the file using the UTF-8 encoding so that DeskNow can read any type of character you type, especially non-latin characters. The UTF-8 encoding ensures that DeskNow can display correctly virtually every character of every known language (NB your browser / OS might need to be configured with additional fonts for some languages).

Windows Notepad is the most common editor that can be used for UTF-8, so if you are using Windows we recommend it for editing DeskNow language files.

At this point, all you have to do is translating all the lines provided in the file in your language.

12.2.1 Translation tips

Try to be consistent with the terms you choose. For instance, refer to 'Message Boards' using always the same term in your language.

If not sure where a particular term (ex. "xyz" is used, use UltraEdit or the Search function of Windows to look for a file in \Program Files\DeskNow\desknow\jsp\original that contains that text. Some terms are system messages, and are defined in DeskNow itself.

The {0}, {1}, etc words are placeholders for words that DeskNow will fill in. Do not translate them, only the statement in which they are contained.

Example:
`Invalid parameter: {0}= Parametro non valido: {0}`

Some terms begin with the \$ character. They are used to translate the names of common objects (ex the email 'in' folder, the 'sent' folder, the 'Announcements' message board, and so on). Translate them as shown in the following example:

\$mailfolder\$drafts= bozze
("bozze" is the Italian translation of "drafts")

12.2.2 Testing your progress

You can use the DeskNow HTTP API to reload the translation file at any time, without having to restart Desknow. This is useful to test your translation.

In short, just open the following URL in a browser:

http://www.yourdesknowserver.com/desknow/admin?pwd=password&action=main_i18nreload

where "password" is the password of the admin user.

NB this will pick up changes that occurred in the translation files, but not in International.cfg . If you modify International.cfg you need to restart DeskNow.

12.2.3 Share your work

We encourage you to share your work with other DeskNow users. Use the DeskNow public forums on the web to communicate your intention of working on a translation, and submit translations so that they can be used by others, and can be included with the main DeskNow distribution. We usually offer discount vouchers for DeskNow licenses for translations in languages that we don't have. Please contact us for more details.

12.2.4 What happens when a new version of DeskNow is released

If you have submitted your translation to us, when a new release of DeskNow is ready to be released we will generate an updated translation file that includes your translation and eventual new terms used in the new functionalities. In this way, your translation will not be lost, and only the new terms will need to be translated. This is usually a very easy task, often consisting of no more than 20 lines of new text.

12.3 Time zone

DeskNow can use any default time zone, potentially different from the time zone of the computer on which it is running. The default time zone is used for the timestamps in log files, and for new user accounts.

The default time zone can be set using the Configuration Wizard (in graphic mode), or by manually changing the ServerTimeZone parameter (see section 14.2.1).

Every user can choose a different time zone for his / her account. All the dates and times in DeskNow are adjusted according to the time zone of the user viewing the information. This means that an appointment in a shared calendar will appear at different times to two different users that work with different time zones. This is essential to organize conference calls, etc.

13 Clustering and load balancing

Clustering is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See chapter 16 for information on how to purchase a DeskNow license.

It is possible to run instances of DeskNow on multiple servers, for the same DeskNow site.

13.1 Advantages

By clustering multiple DeskNow servers to process requests and traffic, two main advantages are achieved:

- Improved performance for sites with high load (many users, many emails/hour)
- High availability: if a server goes down for any reason, the service is not interrupted, and it is provided by the other available servers

Very often for small and medium enterprises the high availability benefit is the most relevant, so you may want to setup a DeskNow cluster even if you have relatively few users and performance is not an issue.

13.2 Overview

A DeskNow server performs multiple functions:

- SMTP service
- POP3 service
- Web access service (including WebDAV)
- XMPP (instant messaging) service

It is possible to setup a cluster of servers in which every server performs all the functions, or a cluster in which specialized servers perform different functions (ex. 4 servers for SMTP and POP3, 6 servers for Web access, etc).

IMPORTANT: currently it is not possible to cluster the XMPP/Instant messaging service, which means that one and only one server in the cluster can be used for instant messaging. This is a known limitation that will be addressed in future releases of DeskNow.

The most common and easiest type of cluster is a cluster in which every server performs all the functionalities (apart from XMPP), and this is the cluster described in the rest of this chapter. If you intend to build a cluster with specialized servers, contact our Support service.

13.3 Load balancing

Load balancing is the operation of distributing the workload across servers in the cluster.

'Workload' in DeskNow comes in two forms:

- Client requests (HTTP, SMTP, POP3, etc)
- Batch jobs (cleaning up the trash folders periodically, creating and sending the spam reports, scanning the calendar database to find events for which an alert must be sent to the user, etc.

13.3.1 Load balancing client requests

Load balancing of client requests can be achieved by an external load balancer, either software or hardware, or, in its simplest form, by DNS (ex. a round robin DNS, in which the name www.domain.com is resolved to different IP addresses, in rotation).

13.3.2 Load balancing batch jobs

Load balancing of batch jobs is performed transparently by the DeskNow servers in the cluster. The servers use the heartbeat to discover how many servers are in the cluster, and to partition the batch job in smaller chunks, with every server handling one chunk.

13.3.3 Load balancing MX records

It is a good practice to have more than one MTA (Mail Transfer Agent: a SMTP server that can accept mail for a domain) per domain. If you have a cluster of DeskNow servers, you should configure the MX records in your DNS domain records to point to the different mail servers in your cluster

13.3.4 Load balancing instant messaging

DeskNow currently does not support load balancing of instant messaging. Therefore you need to select one server of the cluster to serve as IM server.

If you're using a load balancer, all the IM traffic (usually port 5222 and 5223) should be directed to that server. If you're using DNS based load balancing, create a DNS record like `xmpp.domain.com`, and customize the DeskNow pages to use that as server address for the Messenger applet (or use that as server address in your IM client).

13.4 Storage

DeskNow stores data in two main forms: files and database records.

13.4.1 Database

The database must be accessible by all the servers in the cluster.

The database *can* be clustered as well, but it does not have to: the clustering of DeskNow and the clustering of the database are two different things.

However, a clustered database improves performance and availability.

It is not possible to setup a DeskNow cluster using the embedded database. You need to use an external database.

13.4.2 Files

DeskNow stores different types of files:

- Configuration files (.cfg files, SSL certificates, etc.)
- User files (generic files, attachments)
- Messages in mail queues
- Log files

Some of these files need to be shared, (e.g. accessible by all the instances of DeskNow in the cluster), some others must be private (e.g. every instance handles its own files).

The files that must be shared across the DeskNow servers are the configuration files and the user files. The mail queue files must be kept to each server, with the only exception of the POP3 inboxes. Log files are private as well.

13.5 Server to Server communication

Servers in a cluster automatically communicate with each other, to organize sharing of administrative batch tasks (ex. removing old messages from the trash folder, sending the spam report, etc.).

NB this does not provide load balancing of the normal activity, which must be done via a front end load-balancer or a DNS based load balancing. It is only used to load balance batch jobs like the ones mentioned above.

13.5.1 Members of the cluster

Members of the cluster are aware of each other by means of apposite entries in the configuration files. For every server that is part of the cluster, there will be a line like this in Clustering.cfg:

```
Heartbeat.Server.1 192.168.2.100
Heartbeat.Server.2 192.168.2.101
```

13.5.2 Heartbeat

To inform other servers of its presence, a clustered DeskNow server periodically emits a heartbeat in the form of a UDP packet. The packet is sent to all the servers in the cluster, at the configured addresses. For a cluster to work properly, all the servers in the cluster must be able to receive each other's heartbeat.

Heartbeats configuration is described in section 14.17.

13.5.3 New server notice

When a DeskNow server receives the first heartbeat from another server, it will generate a line in the Admin log, informing of the event.

13.5.4 Missing server alert

When a server stops receiving heartbeats from another server for more than a certain time (configurable), it logs an InternalError message in its logs.

Note that if a server never received an heartbeat from the other server, it will not report its disappearance.

13.6 Caching

Every DeskNow server caches particular information to improve performance and avoid database access.

When running in a cluster, the cached copy of some information may not be up to date with changes performed in other servers of the cluster, until the copy expires from the cache, and the relevant data is reloaded from the database.

For this reason, when running in a cluster, objects in the cache have a shorter validity. Nonetheless, some changes may still take up to 10 minutes (but usually not more than 5) to reflect in other servers. Typical examples of cached information are:

- user shares
- user disk usage
- user classes
- user preferences
- mailing list settings
- user groups
- antivirus settings

13.7 Cluster setup

13.7.1 Database setup

Most databases can limit access based on the IP address of the accessing application.

You should ensure that all servers that will need to access the database with the 'desknowserver' account will be authorized to do so.

For instance, in PostgreSQL you should edit the file `/var/lib/pgsql/dat/pg_hba.conf` and add the line:

```
host desknow desknowserver 192.168.0.1 255.255.255.255 password
```

To authorize access from 192.168.0.1.

In MySQL, you *should not* need to do anything in particular if you followed the DeskNow instructions. Otherwise, to enable access from 192.168.0.1, you need to issue the following command in the mysql console:
`grant all privileges on desknow.* to desknowserver@192.168.0.1 identified by 'password';`

13.7.2 Installing the first server

- Create a shared folder (Windows) / NFS mount (Unix/Linux) where to store `/desknowdata`. This can be on a dedicated file server/NAS, or simply on the first server. It is important that, for ease of installation, all servers will be able to access this drive with the same name (ex: `\\server01\shared` on windows, or `/mnt/shared` on Unix/Linux, etc...).

NB a Windows service cannot normally use network drives like X: . But it can, with some adjustments, access UNC paths like [\\server01\share](#) . So if you're setting up the cluster in Windows, avoid using network drives for the shared folder, but use the direct UNC path instead

- Install the first server of the cluster as normal, configuring it so that the `/desknowdata` directory is located on the shared drive (ex `\\server01\shared\desknowdata` or `/mnt/shared/desknowdata` , instead of the usual `/var/desknowdata` or `c:\desknowdata`).
- Edit `/cfg/Clustering.cfg` and set the parameter `Enable` to `TRUE`
- In `Clustering.cfg` add the addresses of the servers in your cluster. If the servers have an interface on a LAN, use the LAN address. Example:

```
Heartbeat.Server.1 192.168.2.100
Heartbeat.Server.2 192.168.2.101
```

- Create a directory for mail queues on the *private* (i.e. not shared) disk of the first server (ex: C:\privatedesknowdata or /var/privatedesknowdata/mail). Every server will have its own mail queues.
- Edit /cfg/Mail.cfg and change the %SET%MAIL_BASE parameter to the path of the mail queue directory.

Example:

```
%SET%MAIL_BASE C:\privatedesknowdata\mail
```

- Create a directory c:\privatedesknowdata\log
- Edit /cfg/Log.cfg (on the shared drive!) and set
BaseFileName c:\privatedesknowdata\log\log
AuditBaseFolder c:\privatedesknowdata\log\audit

This will make every server to use its own private directory for logging.

- Edit /cfg/Main.cfg (on the shared drive) and add this line:
DeskNowTempFolder c:\privatedesknowdata\temp
- (only if you first installed DeskNow before version 2.4) Edit /cfg/Mail.cfg and set the *AntiSpam.Bayes.DataFile* parameter to point to your private queues:

Example:

```
AntiSpam.Bayes.DataFile $(MAIL_BASE)/nospam/nospam.adb
```

- (optional) If you want to use different SMTP, IMAP and POP3 SSL certificates for each server, adjust the parameters *EmbeddedServer.SMTPSSLKeyStore* , *EmbeddedServer.IMAPSSLKeyStore* and *EmbeddedServer.POP3SSLKeyStore* accordingly.

13.7.3 Starting the first server

After having configured the database in the usual way, start the first DeskNow server, and check that the system is working properly (ex. login, send/receive emails, get POP3 emails, etc). It is recommended to start DeskNow in console mode (see Advanced menu on Windows, or 'var/desknow/bin/catalina.sh run' on Linux/Unix).

IMPORTANT: if using Windows, in order to start DeskNow as a service, the service must be run with a user account that has permission to access the shared drive (ex. [\\server01\share](#)). To do so, open the Services section of the Control Panel, right click on the DeskNow service, and open the Properties page. In the Log On section, select an account with access to the shared drive, and input the password necessary to login in that account.)

13.7.4 Adding other servers

- Make sure that the new server can access the shared drive with the same path as it is accessible from the first server.
- Install DeskNow on the new server, make sure to use the shared /desknowdata as data folder (ex \\server01\share\desknowdata)
- Create a directory for mail queues on the *private* (i.e. not shared) disk of the new server (ex: C:\privatedesknowdata or /var/privatedesknowdata/mail). The directory must be at the same path that is used by the first server. This allows for the configuration files to be shared across all the servers, unchanged.
- Create a directory for log files on the *private* (i.e. not shared) disk of the new server (ex: C:\privatedesknowdata\log or /var/privatedesknowdata/log). The

directory must be at the same path that is used by the first server. This allows for the configuration files to be shared across all the servers, unchanged.

- If you intend to use an antivirus, make sure that it is installed on every server in the cluster, and that the command line to access it is the same for every server.
- Start the new server normally. After a little while (depending on your configuration, but by default it should be less than 1 minute) a notification message should appear in the log.Admin file of the first server, informing that it has detected the heartbeat of the new server.
- Test the new server normally (i.e. accessing it by its direct address, not through the load balancer), and verify all its main functions (web,SMTP.POP3,etc).

13.8 Cluster administration

13.8.1 Monitoring

It is possible at any time to monitor the status of the cluster, i.e. to see which servers are recognized as part of the cluster. Simply login to a server as Super Administrator, and click on Administration / Cluster information.

13.8.2 Configuration changes

Configuration changes to the cluster are usually done either via the Administration console, or by manually editing the configuration files (see section 14).

Since the configuration files are shared, the changes will be picked up automatically by all servers, with a delay of 5 minutes at max (this can be adjusted – see section 14.17.2).

Note that, as in a single server installation, some changes done via the Administration console, and all changes done via direct editing of .cfg files, are not effective until the server is restarted. You need to restart all the servers in the cluster to make these changes effective.

13.8.3 Backup

Backup of data in the cluster involves backing up the following elements:

- Backup of the database
- Backup of the shared drive

Backup of the mail queues of each server – these usually contain transient files (at most, retry files for failed deliveries, or quarantined files), so it is up to the system administrator whether to backup these files or not, but a daily backup would probably have little use, since relevant files stay in these queues for a short time.

You can use the embedded backup facility in the Administration panel. Make sure that the backup path points to a shared drive accessible by every server in the cluster. If you choose to have scheduled backups, the servers in the cluster will automatically elect a server to perform the scheduled backup, at the appropriate time.

13.9 Removing a server

If a server in the cluster fails and cannot be restarted, or you simply want to remove it from the cluster, you may want to move all the files in its mail queue to the mail queue of another server, to let it process the messages that were to be processed by the removed server.

When a server is removed, the other servers in the cluster detect its removal in a matter of seconds (depending on the heartbeat settings), and will adjust themselves accordingly.

14 Configuration reference

This chapter provides a reference for all the configuration parameters of DeskNow server.

14.1 Configuration files

The configuration parameters are contained in several files, all with extension “.cfg”, that are located in \$(DESKNOWDATA)/cfg. \$(DESKNOWDATA) is the directory where all DeskNow data is stored, for example “C:\desknowdata” or “/var/desknowdata”.

If you make any change to any configuration file, you must restart DeskNow for the change to take effect.

The most common configuration settings can also be changed by logging in as Administrator, and clicking on the ‘Administration’ item in the DeskNow menu.

14.1.1 Simple installation

To edit the configuration files, from the Windows Start menu select “Programs/DeskNow/Configuration/” and then one of the “Edit xyz.cfg” items.

14.1.2 Advanced installation

To edit the configuration files, open them with a normal text editor.

14.1.3 Format of the configuration files

Configuration files are normal text files. Parameters are expressed by a parameter name followed by a parameter value, like:

Username *desknow_server*

If the parameter value needs to include spaces, you can enclose it in double quotes:

Description.1 *"Your public web pages"*

The character # marks the beginning of a **comment**. Everything on a line after the character # is ignored by DeskNow.

Configuration files are **case sensitive**.

14.2 Main.cfg

This is the main configuration file, which mainly includes other configuration files, described in the following sections. It does contain some basic settings.

14.2.1 ServerTimeZone

See section 12.3 for an overview of how time zones are used in DeskNow.

You can use this property to change the default time zone. DeskNow usually detects the system timezone automatically, but you may want to change this.

See the file `timezones.txt` in the `/docs` folder for possible values to use for this property. You can also use fixed GMT timezones, like GMT-8 or GMT+05:30 . Fixed GMT timezones do not consider daylight saving.

Example:
`ServerTimeZone Australia/Sydney`

It is also possible (and easier) to use the Configuration Wizard in graphical mode to change the time zone.

14.2.2 Network.LocalIP

You can set one or more IP addresses to be detected by DeskNow as local interface. *By default DeskNow detects local interfaces automatically, so you generally should not need to use this setting.*

In some special cases the autodetection may fail, or you may want to disable some specific interfaces from being used by DeskNow at all (note that you can normally configure DeskNow to use only specific interfaces from the web administration console). This setting lets you completely bypass network autodetection.

Example:
`Network.LocalIP.1 127.0.0.1`
`Network.LocalIP.2 192.168.9.1`

14.3 DeskNowBase.cfg

Defines the base path for the DeskNow directories and files.

14.3.1 %SET%DESKNOWDATA

This is a variable setting, used by many configuration files. The value set for the variable should be the full path of the “desknowdata” folder that you copied from the CD.

Advanced configurations may not store all the files under the same base, and therefore may not need this variable.

Example:

```
%SET%DESKNOWDATA      "c:\desknowdata"
```

14.4 Log.cfg

Defines what is logged, and where. Most of these options can be configured via the web interface, from the administrator's account.

14.4.1 BaseFileName

There is a different log file for every enabled logging flag (see below). Every day a new file is opened, automatically. The flag and date are automatically appended to what you specify here.

Example: if you specify "c:\desknodata\log\log", DeskNow will produce log files like "c:\desknodata\log\log.SMTP.2001-07-19.txt"

14.4.2 AuditBaseFolder

The base folder path for all audit logs.

14.4.3 FileANDConsole

Specify whether log messages must be sent to the console in addition to the log files. Can be either TRUE or FALSE.

14.4.4 CleanupDays

Sets the number of days after which old log files are automatically deleted.

14.4.5 LogListener

Defines a Java class that will be notified of all the log messages. See section 5.3.2

14.4.6 Debug

Enable / disable the logging of debug messages. Can be either TRUE (enable) or FALSE (disable).

IMPORTANT: debug logging can generate very big log files.

Recommended: FALSE

14.4.7 InternalError

Enable / disable the logging of internal server errors. Internal server errors can be due to incorrect configuration (i.e. physical file folders missing, incorrect configuration files, etc) or to problems in the server's code. If an internal server error occurs, please examine carefully the log file. If the information displayed does not point to a configuration problem, please contact the support service, providing all relevant information (i.e. what operation caused the problem, etc.).

Can be either TRUE (enable) or FALSE (disable).

Recommended: TRUE.

14.4.8 Security

Enable / disable the logging of security related events (failed logins, detected virus files, suspect activity and so on). Can be either TRUE (enable) or FALSE (disable).

Recommended: TRUE.

14.4.9 Login

Enable / disable the logging of logins. Can be either TRUE (enable) or FALSE (disable). Failed login attempts are logged under the "Security" flag.

14.4.10 Admin

Enable / disable the logging of messages useful to the Administrator. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.11 Database

Enable / disable the logging of database activity. Can be either TRUE (enable) or FALSE (disable).
IMPORTANT: database logging can generate very big log files and slow down server operations, and should be used only for performance tuning.
Recommended: FALSE

14.4.12 Messenger

Enable / disable the logging of activity of the DeskNow Instant Messaging server. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.13 MessengerDebug

Enable / disable the detailed logging of activity (including all the messages exchanged) of the DeskNow Instant Messaging server. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.14 MessengerNIODebug

Enable / disable the detailed logging of Native IO activity (socket polling) of the DeskNow Instant Messaging server. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.15 MessengerHTTPDebug

Enable / disable the detailed logging of HTTP activity of the DeskNow Instant Messaging server (HTTP is typically used when a direct connection is not possible due to a firewall). Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.16 SMTPIn

Enable / disable the logging of activity related to mail received through the DeskNow SMTP server. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.17 SMTPInDebug

Enable / disable the detailed logging of activity related to mail received through the DeskNow SMTP server. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.18 SMTPOut

Enable / disable the logging of activity related to mail sent by the DeskNow SMTP server. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.19 SMTPOutDebug

Enable / disable the detailed logging of activity related to mail sent by the DeskNow SMTP server. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.20 POP3

Enable / disable the logging of activity related to the DeskNow POP3 server. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.21 POP3Debug

Enable / disable the detailed logging of activity related to the DeskNow POP3 server. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.22 MailIngestion

Enable / disable the logging of activity related to the processing of incoming mail (antispam, filters, etc), both received via SMTP or from external POP3 accounts. Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.23 MailIngestionDebug

Enable / disable the detailed logging of activity related to the processing of incoming mail (antispam, filters, etc), both received via SMTP or from external POP3 accounts. Can be either TRUE (enable) or FALSE (disable).
Recommended: FALSE

14.4.24 Internationalization

Enable / disable the logging of errors related to internationalisation (ex. missing translations, etc.). Can be either TRUE (enable) or FALSE (disable).
Recommended: TRUE

14.4.25 IMAP

Enable / disable the logging of activity (mainly connections) related to the IMAP service.
Recommended: TRUE

14.4.26 IMAPDebug

Enable / disable the detailed logging of activity related to the IMAP service.
Recommended: FALSE

14.4.27 IMAPProtocol

Enable / disable the logging of all the requests and responses processed by the IMAP server.
Recommended: FALSE

14.4.28 Audit

Logs all the 'write' operations (i.e. operations that modify data like contacts, calendars, etc) performed by users.

14.4.29 AuditIM

Logs all the messages sent by users via Instant Messaging.

14.5 Database.cfg

Defines how DeskNow connects to the database. Additionally, defines how IDs are allocated (advanced feature).

14.5.1 JdbcDriver

Java class implementing the JDBC driver. Consult the driver documentation for more details. The class must be on the CLASSPATH of the Java Virtual Machine running DeskNow.

Example: `com.microsoft.jdbc.sqlserver.SQLServerDriver`

14.5.2 Url

URL addressing the database. Consult the JDBC driver documentation for more details.

Example: `jdbc:microsoft:sqlserver://localhost:1433;DatabaseName=desknow`

14.5.3 Username

The username to access the database. The account must have read, write and DDL privileges.

Example: `desknow_server`

14.5.4 Password

The password to access the database.

Example: `password`

14.5.5 NumConnections

Number of simultaneous connections to the database. If the number of concurrent requests at a given time is higher than this value, some requests will have to wait (and possibly be aborted if they wait too long: see `ConnectionPoolTimeout`). The server reserves one connection for special internal uses. This connection will not be available to fulfil normal client requests.

Generally, the number of database connections should be equal to the number of QMail, ExternalPOP3 and SMTP threads specified in `Mail.cfg` (see 14.11) PLUS the number of concurrent client requests that the server is expected to serve.

14.5.6 ConnectionPoolTimeout

Timeout before an operation is interrupted waiting for a connection to be available (expressed in seconds).

14.5.7 IDFactory.DBAllocatedPoolSize

(documentation incomplete, leave default values for now)

14.5.8 IDFactory.ChunckedPoolSize

(documentation incomplete, leave default values for now)

14.5.9 IDFactory.NumProducers

(documentation incomplete, leave default values for now)

14.6 Zroots.cfg

This section is reserved only to advanced configurations. In most cases, access to the native file system can be achieved by enabling the Administrator to access the native file system (section 14.10.4), and letting the Administrator share native folders with the users.

Defines multiple root bases for the native filesystem.

A zroot is a base path onto which a vroot is mapped (see Vroots.cfg).

A zroot usually points to a directory in the native filesystem.

Under this directory, for every user account there is a directory named with the username of the account.

Example:

The zroot "webfiles" is mapped to the folder "c:\desknowdata\userfolders\webfiles".

The vroot WebFiles is mapped to the zroot "webfiles" , and is visible to the user (as defined in Vroots.cfg).

As a result, when user "joe.smith" logs in, he will see "WebFiles" in the DeskNow tree, and the files contained in it are the files actually contained in "c:\desknow\userfolders\webfiles\joe.smith"

Thanks to DeskNow double-layered virtual filesystem, it is very easy to manage user's data.

For instance, if a single disk is not large enough to store all the files, it is possible to store all the "WebFiles" files in a second disk, just by changing the native path of the associated zroot.

The default configuration, however, maps all the zroots under {DESKNOWBASE}/userfolders.

Syntax:

<zroot> <native directory path>

as a convention, zroots are all in lowercase

It is strongly discouraged to have zroots that differ only by their case.

Example:

%SET%ZROOT_BASE \$(DESKNOWDATA)/userfolders

webfiles \$(ZROOT_BASE)/web

14.7 Vroots.cfg

This section is reserved only to advanced configurations. In most cases, access to the native file system can be achieved done by enabling the Administrator to access the native file system (section 14.10.4), and letting the Administrator share native folders with the users.

Defines multiple roots in the virtual file system of a user.
A vroot is a "virtual root" for the virtual file system of a user.

Not all the vroots are visible to the users: some of them are used for internal services only (such as storing email attachments, and so on).
Every vroot is mapped to a zroot (see file Zroots.cfg), which in turn is mapped to a physical directory on the native filesystem.

The vroots specified here are created for every new account.
Once the account is created, the mappings for that account are saved in the database, and not read from this file.

For the reason above, it is very easy to distribute the load among multiple disks, as the system grows:

for instance if you already have 1000 users and you want to use a second disk for the mail attachments of new users, it is sufficient to create a new zroot "inmail2" (in file Zroots.cfg) and then change the vroot "InMail" in this file to point to "inmail2" instead of "inmail".

While old users will keep using the old disk, new accounts will use the new disk.

Some vroots are "system" vroots, and are needed by the server to work properly. You can chose how to map them to zroots, but you cannot remove them. The following are system vroots: InMail, Mbm, Cts, Temp, ServerIcons, Wombats.

Syntax:

<vroot> <zroot>

As a convention, zroots are all in lowercase, whereas vroots are not.
It is strongly discouraged to have vroots that differ only by their case.

14.8 Registration.cfg

14.8.1 MinCommunityNameLength

Minimum length for a community name. Creation of communities with a shorter name will be refused.

14.8.2 MinUserNameLength

Minimum length for a username. Creation of user accounts with a shorter username will be refused.

14.8.3 MinPasswordLength

Minimum length for a user password. Attempts to set or change a user password shorter than this value will fail.

14.8.4 MinPasswordNonLetters

Minimum number of non-letter characters in the password. The default is 0.

14.8.5 MinPasswordLetters

Minimum number of letter characters in the password. The default is 0.

14.8.6 InitialFiles section

This section instructs the server to copy to the user file system some predefined files when the account is created.

Syntax:

InitialFiles.<vroot> <source path>
(copy all the files in <source path> to the user's vroot)

Example: InitialFiles.WebFiles \$(DESKNOWDATA)/initialwebpages

14.8.7 ReservedNames section

Defines names that cannot be registered as usernames or community names.

Example:

```
ReservedNames.root          1
ReservedNames.superman     1
```

Note that it is impossible anyway to create users or communities with a name that is already used by another user or community in the same domain. Different domains can have user accounts and communities with the same name.

14.9 VisibleFolders.cfg

Defines which parts of the user's file system are shown in the client.

Example:

The following section states that the vroot "WebFiles" is to be shown on the client, under the "Files" node, with the name "Web Pages", and the description "Your public web pages".

The folder is published on the web with the url

"http://public.desknow.com/<username>"

NB It is the administrator's responsibility to set up the web server so that the folder is actually published!

```
Vroot.1      WebFiles
PrettyName.1 "Web pages"
Description.1 "Your public web pages"
URL.1       http://public.desknow.com/
```

The following section states that the vroot "MyDocs" is to be shown on the client, under the "Files" node, with the name "Personal Documents", and the description "Your personal documents".

The folder is NOT published on the web.

```
Vroot.2      MyDocs
PrettyName.2 "Personal Documents"
Description.2 "Your personal documents"
```

IMPORTANT: in many circumstances, the default DeskNow DirectFiles feature is more than enough to let users publish their documents on the web. The configuration presented here, however, illustrates an example of more complex uses of DeskNow.

14.10 Security.cfg

Grants / denies particular access to features.

14.10.1 ResetAdminPassword

This option must be used to reset the password of the administrator of the default domain (the “super administrator”). Every other password can be changed by this user.

To reset the administrator password, simply add a line like this to the Security.cfg configuration file:

```
ResetAdminPassword newpassword
```

And restart DeskNow. DeskNow will make “newpassword” the new password for the Administrator, and remove the line from the Security.cfg file, to protect it.

14.10.2 AllowSuLoginFromCommunityManager

Enables / disables su-like login for community managers (see section 3.1.16). Possible values are TRUE and FALSE.

14.10.3 MaxFileUploadSize

Determines the maximum allowed size for an upload operation. This is the sum of the sizes of all files uploaded in a single operation. This includes files uploaded as attachments.

The value is expressed in Mb.

14.10.4 ShowServerFiles

Enables / disables access to all the files on the native filesystem of the server.

Possible values are TRUE and FALSE.

If TRUE, the Administrator account will have a **[All server files]** folder in the Files section. Under this folder are visible all the files on the server’s file system, and all the normal operations can be performed on them. As Administrator you can also share parts of the native file system to users (ex. drives that contains documents).

14.10.5 AllowCrossDomainSharing

Enables (TRUE / disables (FALSE) sharing of objects (Calendars, file folders, etc.) across domains. The default setting is FALSE (disable).

14.10.6 TrackLastLoginTimeFromExternalConnections

If this property is set to TRUE, the ‘Last login’ time recorded for every user includes logins from mail clients (POP3, IMAP, SMTP Authentication) and SyncML clients.

If this property is set to FALSE, only logins to the web interface will be considered for the ‘Last login’ property. The default is FALSE.

14.10.7 EnableShareForcingAcrossDomains

If enabled (TRUE), when the Super Administrator creates a share that applies to virtual domains, the link to this share will be automatically enforced on accounts of those domains. The default setting is FALSE (disable).

14.10.8 ExternalAuthentication.Enable

Enable (TRUE) / disable (FALSE) the use of an external authentication mechanism to authenticate user logins and verify the existence of accounts. This can be used to plug in a custom authentication method that can integrate with other systems. Note that a plugin for Active Directory authentication is already available, and sample configuration parameters are shown in the Security.cfg file.

The default value is FALSE.

14.10.9 ExternalAuthentication.CacheValidityTime

Set the amount of time, in seconds, for which DeskNow will cache positive results from the external authentication provider. This will reduce network traffic, but would cause a small delay in propagating the changes that are made to the account in the external directory (ex. change of password). To disable caching, set this value to 0.

The default value is 120 (2 minutes).

14.10.10 ExternalAuthenticationAutomaticAccountCreation.Enable

Enable (TRUE) / disable (FALSE) the possibility to automatically create user accounts that exist in the external authentication domain, but don't exist in DeskNow. This functionality must be supported by the Authentication plugin (see the API). The built-in Active Directory plugin supports this functionality.

The default value is FALSE.

14.10.11 ExternalAuthenticationAutomaticAccountCreation.ImportADMailAliases

Enable (TRUE) or disable (FALSE) autocreation of mail aliases saved as attributes in ActiveDirectory, in the form proxyAddresses: SMTP:alias@domain.com . This is the format stored by Microsoft Exchange. When this option is TRUE, DeskNow will automatically create such aliases unless an alias (including with wildcards) or an account with the same address already exists.

The default value is FALSE.

14.10.12 ExternalAuthenticationAutomaticAccountCreation.ImportIMAPMail.Enable

Enable (TRUE) or disable (FALSE) automatic import of mail from an external IMAP server when an account is autocreated. The default value is FALSE.

14.10.13 ExternalAuthenticationAutomaticAccountCreation.ImportIMAPMail.Host

Address (name or IP) of the IMAP server to import mail from upon account autocreation.

14.10.14 ExternalAuthenticationAutomaticAccountCreation.ImportIMAPMail.Port

Port number of the IMAP server to import mail from upon account autocreation.

14.10.15 ExternalAuthentication.AutomaticAccountCreation.Community

Sets what community the new user accounts created via automatic account creation will be created in. If this property is not set, accounts will be created in the default community. This setting can be overridden by CustomQuery.Community settings.

14.10.16 ExternalAuthentication.AutomaticAccountCreation.UserClasses

Sets what user class the new user accounts created via automatic account creation will be created in. If this property is not set, accounts will be created in the 'Normal' user class.

14.10.17 ExternalAuthentication.AutomaticAccountCreation.DefaultDiskQuota

Specifies the default disk quota (soft disk quota) to assign to new accounts created via external authentication. The value is expressed in Mb. Use -1 for unlimited (the default).

14.10.18 ExternalAuthentication.AuthenticationProvider

The full name (including package) of the class that provides the custom authentication logic. See docs/control/api/index.html for more information on how to implement a custom plugin.

For instance, the built-in Active Directory plugin is
`com.deskknow.control.authentication.impl.ActiveDirectoryAuthenticationProvider`

Other providers are available. Please see section 10.

If using a custom provider written by you, please make sure that the class is available in the CLASSPATH. We recommend storing the plugin class (and all its related classes, if any) in a jar file and storing this file in `\desknowdata\lib`.

14.10.19 ExternalAuthentication.CustomProperties.ServerAddress

The server address (IP or literal) of the Active Directory or LDAP server.
The default value is localhost.

14.10.20 ExternalAuthentication.CustomProperties.ServerPort

The port used to connect to Active Directory or LDAP server.
The default value is 389.

14.10.21 ExternalAuthentication.CustomProperties.BrowserUsername

This parameter is valid only when using the Active Directory or LDAP authentication plugin.

The username of an account that is allowed to browse the Active Directory server. This is necessary only when ExternalAuthentication.AutomaticAccountCreation.Enable is set to TRUE, because this account is used to lookup user accounts in the directory.

14.10.22 ExternalAuthentication.CustomProperties.BrowserPassword

This parameter is valid only when using the Active Directory or LDAP authentication plugin.

The password for the BrowserUsername (see above).

14.10.23 ExternalAuthentication.CustomProperties.SearchSubtrees

When this parameter is set to TRUE, DeskNow will search for an account in the specified branch and in all its sub-branches.

Example: if the branch is "CN=Users,DC=domain,DC=com" and this parameter is set to TRUE, DeskNow will search users in "CN=Users,DC=domain,DC=com" and any sub-branch like "OU=Managers,CN=Users,DC=domain,DC=com", "OU=Sales,CN=Users,DC=domain,DC=com", and so on.

The default value is FALSE.

14.10.24 ExternalAuthentication.CustomProperties.AuthenticationMechanisms

This parameter is valid only when using the Active Directory authentication plugin. This parameter specifies what authentication mechanisms should be attempted when connecting to the Active Directory server. Mechanisms must be separated by a space.

Some valid mechanisms are:

DIGEST-MD5 this is the recommended by the LDAP 3 specs

CRAM-MD5

GSSAPI Also known as Kerberos v. 5

SIMPLE Plain text password

All the mechanisms will be attempted in the order provided, and the first successful one will be used. If you know exactly what mechanisms are enabled in your AD server, you should make sure that it is the first of the list, so that lookups will be faster.

The default value is "DIGEST-MD5 CRAM-MD5 GSSAPI SIMPLE"

14.10.25 ExternalAuthentication.CustomProperties.CustomQuery.x

Set a custom location in the LDAP tree for where to search for accounts in a domain. See section 10 for more information.

14.10.26 ExternalAuthentication.CustomProperties.UIDAttribute

Sets the name of attribute in the LDAP directory that holds the username.

DeskNow assumes that the attribute that contains the username is 'userid'. If the username is stored in a different attribute, you can add this line:

```
ExternalAuthentication.CustomProperties.UIDAttribute username
```

14.10.27 ExternalAuthentication.CustomProperties.UseCNAttribute

To determine the full name of the person in an LDAP directory, by default DeskNow uses the CN attribute, and assumes that the first word is the first name, and the last word is the last name. If your directory has specific attributes to hold the first and last name of a person, you can disable this and use the GNAAttribute and SNAAttributes (see below):

```
ExternalAuthentication.CustomProperties.UseCNAttribute FALSE
```

14.10.28 ExternalAuthentication.CustomProperties.GNAAttribute

Specifies what is the attribute in the LDAP directory that holds the person's first name (if the CN parsing is not used). The default value is 'givenname'.

```
ExternalAuthentication.CustomProperties.GNAAttribute givenname
```


14.10.29 ExternalAuthentication.CustomProperties.SNAttribute

Specifies what is the attribute in the LDAP directory that holds the person's last name (if the CN parsing is not used). The default value is 'sn'.

`ExternalAuthentication.CustomProperties.SNAttribute sn`

14.10.30 IMAPAutoImport.Enable

Enables/disables the automatic import of accounts from an external IMAP server. The default is FALSE (disable). Once the autoimport is finished, it is recommended to set this flag to FALSE for security reasons. **NB** it is not possible to use this feature in conjunction with external authentication (Active Directory, LDAP, etc). When using IMAP autoimport, you must turn off external authentication. You can turn it on again once imap migration is completed and you have turned it off.

14.10.31 IMAPAutoImport.ExternalIMAPServer.Host

The internet address (name or IP) of the remote IMAP server from which to autoimport accounts.

14.10.32 IMAPAutoImport.ExternalIMAPServer.Port

The internet port of the remote IMAP server from which to autoimport accounts. The default is 143 if SSL is FALSE, 993 if SSL is TRUE.

14.10.33 IMAPAutoImport.ExternalIMAPServer.SSL

Whether to use (TRUE) or not (FALSE) SSL when connecting to the external IMAP server.

14.10.34 FailedLoginAttemptTracker.enable

Enables (TRUE)/disables (FALSE) the mechanism to prevent login attempts from an IP address after too many failed attempts during a period of time.

When enabled, the rule is: if during the last *CheckPeriod* minutes an IP address has failed to login for *TriggerAttempts* times, then that IP address is banned from logging in for the following *LockoutPeriod* minutes.

The default is FALSE (disabled).

14.10.35 FailedLoginAttemptTracker.CheckPeriod

See above.

14.10.36 FailedLoginAttemptTracker.LockoutPeriod

See above.

14.10.37 FailedLoginAttemptTracker.TriggerAttempts

See above.

14.10.38 RemoveSharesPointingToNonExistingFolders

If this property is set to TRUE, DeskNow will delete a file share if the corresponding folder can no longer be found on disk. This is the default behaviour. However in some rare circumstances, when the folder is mounted on NFS, it could be reported as 'non existing' even if it is. If you're using DeskNow with NFS, and experience the 'disappearing' of file shares (eg. the setting where permissions are stored, not the files themselves!), then try setting this to FALSE.

14.11 Mail.cfg

Specifies the parameters for Mail management.

DeskNow can integrate with SMTP servers, POP3 servers and directly with Qmail servers, or can run as stand-alone mail server.

DeskNow can also disable external email, preventing users to send emails to the Internet, and functioning as internal messaging system.

NOTE: the most important mail settings of DeskNow can be configured by the Configuration Wizard. If you have used the Windows installer, you can run the wizard from the DeskNow program group in the Start Menu.

If you have used the manual installer, the wizard can be run by changing the current directory to the bin directory contained in the package, and executing:
.config.sh <path of desknowdata> (Unix/Linux)
config.bat <path of desknowdata> (Windows)

Example:
`./config.sh /var/desknowdata`
`config.bat c:\desknowdata`

14.11.1 Domain

This parameter is now obsolete, and replaced by virtual domain management. It has no relevance.

14.11.2 Administrator

All mail problems (malformed messages, etc.) will be notified to this email address.

Example:
`administratorEmail myname@myisp.com`

14.11.3 AddressSeparators

Specify the characters (each single one of them) are treated as address separators for the To, Cc, Bcc fields typed in by the user.

Example:
`AddressSeparators ;,`

14.11.4 ExternalPOP3

This section refers to the ExternalPOP3 Daemon that is inbuilt in DeskNow. This daemon is responsible of periodically fetching emails on behalf of the users from their **External accounts**.

14.11.5 ExternalPOP3.delay

When DeskNow starts, the daemon checks all the external accounts for all the users, and retrieves new emails. Then it waits for the amount of time specified by this parameter (expressed in seconds), and then repeats the cycle.

Example:
`ExternalPOP3.delay 600`

14.11.6 ExternalPOP3.fetchThreads

Sets the maximum number of concurrent threads that are used to access external POP3 servers and download new emails. Under light load, DeskNow will automatically reduce the number of threads in use as necessary.

14.11.7 ExternalPOP3.parseThreads

Sets the maximum number of concurrent threads that are used to decode and process new emails retrieved by the fetchThreads. Since mail decoding is usually less network dependant, the number of parseThreads should generally be lower than that of fetchThreads.

14.11.8 ExternalPOP3.rootPath

DeskNow uses this folder to process all the emails retrieved before they are assigned to the proper account.

14.11.9 ExternalPOP3.retryDelay

Sets the delay in seconds between two attempts to manage those emails that DeskNow was not able to decode or assign to an user (ex. because the user's disk usage has reached its limit).

14.11.10 ExternalPOP3.retryMaxCycles

Sets the number of times that an email will be processed before it will be considered undeliverable. Error emails are saved in the **error** folder under the rootPath. Since the ExternalPOP3Daemon is effectively a POP3 client, it will NOT automatically send an error message back to the sender.

14.11.11 ExternalPOP3.CopyFetchedMailToPOP3Inbox

If this flag is TRUE and the DeskNow POP3 server is enabled, DeskNow will store a copy of the mail retrieved from external POP3 accounts into your DeskNow POP3 inbox.

14.11.12 ExternalPOP3.Timeout

Sets the timeout in seconds for a connection to a POP3 server. When retrieving emails from an external server, if DeskNow does not receive any data from the server for longer than the time specified here, it will close the connection. Some mail servers can be very slow if they have tens of thousands of messages, so in this case you may want to increase the default value.

The timeout value is expressed in seconds. The default value is 90 seconds.

14.11.13 ExternalPOP3.quarantinePath.SizeLimit

Sets the maximum size of the quarantine folder where messages downloaded from external POP3 accounts are stored if infected by virus. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.

The value is expressed in Mb. The default is 100 Mb.

14.11.14 ExternalPOP3.quarantinePath.NumberLimit

Sets the maximum number of files allowed in the quarantine folder where messages downloaded from external POP3 accounts are stored if infected by virus. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes

older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.
The default is 1000 files. Note that many Linux filesystems cannot handle more than 32,000 files in a folder.

14.11.15 ExternalPOP3.errorPath.SizeLimit

Sets the maximum size of the folder where messages downloaded from external POP3 accounts are stored if they could not be assigned to an inbox. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.
The value is expressed in Mb. The default is 100 Mb.

14.11.16 ExternalPOP3.errorPath.NumberLimit

Sets the maximum number of files allowed in the error folder where messages downloaded from external POP3 accounts are stored if they could not be assigned to an inbox. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.
The default is 1000 files. Note that many Linux filesystems cannot handle more than 32,000 files in a folder.

14.11.17 DirectIn

This section refers to the daemon that processes messages delivered by a mail server to a specific folder, using the qmail message format. This includes, of course, the **qmail** mail server itself, or the DeskNow integrated mail server.

14.11.18 DirectIn.use

Enables / disables the DirectIn Daemon. you can disable it if you are not using DeskNow as mail server, or are not using **qmail** as backend mail server. Since this is also used for internal mails, *this flag should always be set to TRUE*, unless you have very good reasons to do otherwise.
Possible values are TRUE (enable) or FALSE (disable).

14.11.19 DirectIn.delay

When DeskNow starts, the daemon checks the specific mail folder (see below for the rootPath setting) for new messages, and parses and assigns all the messages it has found. Then it waits for the amount of time specified by this parameter (expressed in seconds), and then repeats the cycle.

14.11.20 DirectIn.parseThreads

Sets the number of concurrent threads that are used to decode and process new emails retrieved by the daemon. Since mail decoding is usually very CPU and disk intensive, the number of parseThreads should generally be low, since very little performance would be gained by increasing the parallelism.

14.11.21 DirectIn.rootPath

DeskNow uses this folder to process all the emails retrieved before they are assigned to the proper account. In particular, DeskNow looks for new messages in the **base** folder located under the directory specified here. The embedded DeskNow mail server, or a qmail mail server, should be configured to deliver all the email for the domain into this folder.

14.11.22 DirectIn.retryDelay

Sets the delay in seconds between two attempts to manage those emails that DeskNow was not able to decode or assign to an user (ex. because the user's disk usage has reached its limit).

14.11.23 DirectIn.retryMaxCycles

Sets the number of times that an email will be processed before it will be considered undeliverable. Error emails are saved in the **error** folder under the rootPath. In case the email was addressed to an account that does not exist in DeskNow, an error email will be sent back to the sender.

14.11.24 DirectIn.RecipientPrefix

Specifies the default prefix that identify the mail recipient in a mail message in qmail format. The default value should not be changed unless there is a very good reason to do so.

14.11.25 DirectIn.SendBounceMessagesOnError

Specifies whether DeskNow should send a bounce email back if an email could not be assigned to an user (ex. the user has reached the full disk quota, etc). Note that this settings is only considered if the problem was not due to the fact that the email was infected by a virus. Bounce messages for viruses can be enabled/disabled in the Administration/Antivirus page, so that it is possible to enable bounces for normal problems, and disable them for infected files (useful in case of virus worms spreading on the Internet). The default value is TRUE (enable sending bounce messages). Note that this setting does NOT affect email received via external POP3 accounts, only mail that arrives directly in DeskNow's local queue (typically via SMTP).

14.11.26 DirectIn.quarantinePath.SizeLimit

Sets the maximum size of the quarantine folder where messages arrived from the normal SMTP queue are stored if infected by virus. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.

The value is expressed in Mb. The default is 100 Mb.

14.11.27 DirectIn.quarantinePath.NumberLimit

Sets the maximum size of the quarantine folder where messages arrived from the normal SMTP queue are stored if infected by virus. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.

The default is 1000 files. Note that many Linux filesystems cannot handle more than 32,000 files in a folder.

14.11.28 DirectIn.errorPath.SizeLimit

Sets the maximum size of the folder where messages arrived from the normal SMTP queue are stored if they could not be assigned to an inbox. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.

The value is expressed in Mb. The default is 100 Mb.

14.11.29 DirectIn.errorPath.NumberLimit

Sets the maximum number of files allowed in the error folder where messages arrived from the normal SMTP queue are stored if they could not be assigned to an inbox. This prevents an excess of infected files from filling up the server's disks. DeskNow periodically deletes older files from this folder (every 31 days by default) anyway, but if this limit is reached, old files are forcefully deleted immediately.

The default is 1000 files. Note that many Linux filesystems cannot handle more than 32,000 files in a folder.

14.11.30 Webmail.allowExternalDelivery

Enables/disables sending emails to the outside world from the web interface. Use FALSE to implement a closed system for internal communication only.

Possible values are TRUE (enable) or FALSE (disable).

14.11.31 Webmail.CheckInternalRecipientExistence

Enables/disables checking for the existence of the recipient when using webmail.

When sending emails via webmail, if the recipient is in a local domain, desknow can check if the recipient user exists in that domain, and warn immediately if it doesn't. This helps correcting immediately typos. In some configurations, especially when using an external mail server, you may want to disable this check (setting this to FALSE).

Possible values: TRUE (check user existence) or FALSE (don't check: if the recipient mail server complains about the recipient, a mail delivery error message will be delivered back to the sender). The default is TRUE.

14.11.32 Webmail.MaxRecipients

Indicates the maximum number of recipients that can be specified in a single email sent via webmail. Note that a mailing list is considered as a single recipients. 0 means unlimited. The default value is 0.

14.11.33 EmbeddedServer.ServerName

Set the name that the DeskNow SMTP and POP3 server uses to greet clients. By default DeskNow uses the name of the computer. You can manually set this to something like mail.mydomain.com or similar.

14.11.34 EmbeddedServer.BannerSoftwareName

Set the name of the software to use in banner greetings. The default is DeskNow . You can change this to add a layer of security by hiding the software type in use.

14.11.35 EmbeddedServer.useSMTPIn

Enables / disables the use of the SMTP server.

Possible values are TRUE (enable) and FALSE (disable). Default: TRUE (enable).

14.11.36 EmbeddedServer.useSMTPPlain

Enables/disables plain (unencrypted) communication for the SMTP server. Default: TRUE (enable).

14.11.37 EmbeddedServer.SMTPDaemonPort

Defines the port on which the SMTP server will accept unencrypted connections. The internet standard is 25, so DO NOT CHANGE THIS unless you perfectly know what

you're doing. If this value is different from 25, the DeskNow SMTP server will not be able to receive emails from the Internet, unless there is some SMTP gateway in front.

14.11.38 EmbeddedServer.enableSecondarySMTPDaemonPort

Enables (TRUE) or disables (FALSE) the use of a second port for incoming SMTP connections. This is useful to let mail clients connect to the mail server when they're not allowed to connect to the normal SMTP port (firewalls, etc.). The default value is FALSE (disabled).

14.11.39 EmbeddedServer.secondarySMTPDaemonPort

Defines the port number for the secondary SMTP port (see above). The default value is 587.

14.11.40 EmbeddedServer.SMTPBindInterface

Defines the interface on which the SMTP server will accept unencrypted (normal) connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.SMTPBindInterface 162.4.12.65

14.11.41 EmbeddedServer.useSMTPSSL

Enables/disables SSL (encrypted) communication for the SMTP server. Default: TRUE (enable).

14.11.42 EmbeddedServer.SMTPSSLDaemonPort

Defines the port on which the SMTP server will accept encrypted connections. The default value is 465.

14.11.43 EmbeddedServer.SMTPSSLBindInterface

Defines the interface on which the SMTP server will accept encrypted connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.SMTPSSLBindInterface 162.4.12.65

14.11.44 EmbeddedServer.SMTPOutInterface

Defines the interface used by the SMTP delivery service to create outgoing connections to other SMTP servers. If no interface is specified (the default), the server will automatically pick an interface from the available ones. NB it is often useful to set this parameter on multihomed hosts, to comply with Reverse DNS checks.

Example:

EmbeddedServer.SMTPOutInterface 162.4.12.65

14.11.45 EmbeddedServer.routingThreads

Indicates the number of threads dedicating to routing (ie deciding whether a message is local to the domain, or needs to be delivered across the Internet).

Since this operation is not affected by network delays, the number of routingThreads should generally be low, since very little performance would be gained by increasing the parallelism.

14.11.46 EmbeddedServer.remoteDeliveryThreads

Indicates the maximum number of threads dedicated to delivering emails to the Internet, by connecting to the recipient domain's SMTP server and using the SMTP protocol to transfer the message. This operation is very dependant on network delays, so you should set this number not too low to benefit from parallelism. Under light load DeskNow will automatically reduce the number of threads in use as necessary.

14.11.47 EmbeddedServer.remoteDeliveryRetryDelay

When a message cannot be delivered it is placed in a retry queue. After the amount of time specified by this parameter (in seconds), a new delivery attempt is made.

14.11.48 EmbeddedServer.remoteDeliveryTimeout

Number of seconds after which DeskNow waits for a reply from an SMTP server before closing the connection. This parameter is important to avoid that a broken SMTP server locks up the delivery threads of DeskNow indefinitely.

14.11.49 EmbeddedServer.remoteDeliveryRetryMaxCycles

Number of attempts that DeskNow makes to deliver a message. If after all the attempts the message was still undeliverable, DeskNow will send back an error message to the sender.

14.11.50 EmbeddedServer.SMTPThreads

Indicates the maximum number of threads dedicated to receive emails through SMTP connections.

This number should be proportioned to the expected number of concurrent SMTP connection attempts received by DeskNow.

14.11.51 EmbeddedServer.SMTPInConnectionTimeout

Indicates the number of seconds the SMTP In service will wait for data from the other party before terminating the connection.

The default value is 120 seconds.

14.11.52 EmbeddedServer.SMTPMaxMessageSize

Indicates the maximum size of a message accepted by the SMTP server. NB this applies to both outgoing and incoming email. The size is expressed in bytes. If this setting omitted, there is no limit to the message size.

Note: attachments can take more space than their normal size when they are attached to a mail message, because of the MIME encoding. Allow roughly 35% extra space, i.e. an attachment that is 10Mb in size can cause the size of the encoded mail message to be up to 13.5 Mb.

14.11.53 EmbeddedServer.SMTPInMaxRecipients

Indicates the maximum number of recipients that the SMTP service will allow for an incoming connection. 0 means no limit. The default value is 0.

14.11.54 EmbeddedServer.SMTPInMaxEmailsSentPerUserPerDay

Indicates the maximum number of emails per day that an user can send.

The count is per server (in a cluster, every server has its own count – they don't add up). The count can only be kept if the user authenticates to the SMTP server (this means that SMTP authentication should ideally be enforced, by disallowing SMTP relay based on IP addresses).

The count is reset at midnight of every day.

The default value is 1000.

14.11.55 EmbeddedServer.SMTPInMaxEmailsSentPerUserPerHour

Indicates the maximum number of emails per hour that an user can send.

The count is per server (in a cluster, every server has its own count – they don't add up). The count can only be kept if the user authenticates to the SMTP server (this means that SMTP authentication should ideally be enforced, by disallowing SMTP relay based on IP addresses).

The count is reset at the beginning of every hour (0 minutes).

The default value is 1000.

14.11.56 EmbeddedServer.rootPath

DeskNow uses this folder as a workspace for all the emails processed by the mail server. In particular, POP3 inboxes are kept under the /pop3 folder under the root.

14.11.57 EmbeddedServer.usePOP3

Enables / disables the POP3 service. Unless you really need to keep using traditional email clients, we suggest disabling POP3. In this way users will benefit of a more integrated work environment (mail, files, calendar, collaboration) without the duplication of an external email client. The web mail provided by DeskNow has all the features (and sometimes more) of a traditional mail client, and is not subject to mail viruses, etc.

14.11.58 EmbeddedServer.usePOP3Plain

Enables/disables plain (unencrypted) communication for the POP3 server. Default: TRUE (enable).

14.11.59 EmbeddedServer.POP3DaemonPort

Defines the port on which the POP3 server will accept unencrypted (normal) connections. The default value is 110, if you change this you will need to change the default value in the mail clients that connect to this service.

14.11.60 EmbeddedServer.POP3BindInterface

Defines the interface on which the POP3 server will accept unencrypted (normal) connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.POP3BindInterface 162.4.12.65

14.11.61 EmbeddedServer.usePOP3SSL

Enables/disables SSL (encrypted) communication for the POP3 server. Default: TRUE (enable).

14.11.62 EmbeddedServer.POP3SSLDaemonPort

Defines the port on which the POP3 server will accept encrypted connections. The default value is 995, if you change this you will need to change the default value in the mail clients that connect to this service.

14.11.63 EmbeddedServer.POP3SSLBindInterface

Defines the interface on which the POP3 server will accept encrypted connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.POP3SSLBindInterface 162.4.12.65

14.11.64 EmbeddedServer.POP3Threads

Maximum number of threads dedicated to serve concurrent POP3 connections.

14.11.65 EmbeddedServer.POP3ConnectionTimeout

Indicates the number of seconds the POP3 service will wait for data from the other party before terminating the connection.
The default value is 120 seconds.

14.11.66 EmbeddedServer.ForceInboxStore

If this flag is TRUE, DeskNow will store incoming messages in the POP3 inbox even if the POP3 service is disabled. This is useful if you want to disable the POP3 service only temporarily. If this flag is FALSE and usePOP3 is FALSE, DeskNow will not store messages in the user's inbox, but will ingest them directly in the database, for webmail use only.

This flag is TRUE by default on installations performed after release 2.2 (included).

14.11.67 EmbeddedServer.authorizeRelay

By default, the DeskNow mail server blocks any attempt of mail relay (i.e. the SMTP server does not accept mail directed to the internet from IP addresses different from 127.0.0.1). This is very important to prevent abuse by mail spammers.

If you want mail clients or other applications to be able to send mails to the internet through the DeskNow SMTP server, you can do two things (**you can do one of them, or both**):

1) explicitly authorize them by adding their IP address or subnet to the configuration file. You can add multiple IPs or subnets.

Example:

```
EmbeddedServer.authorizeRelay.1    65.14.13.122
EmbeddedServer.authorizeRelay.2    65.14.13.123
EmbeddedServer.authorizeRelay.3    65.14.14.
```

(Note the final dot in the third line)

*This will enable computers with IP 65.14.13.122, 65.14.13.123 and any computer in the subnet 65.14.14.**

2) enable SMTP authentication: see the following section 14.11.69

14.11.68 **EmbeddedServer.AuthorizeRelayFromThisServer**

If this flag is TRUE, all local IP addresses of this server are implicitly authorized to relay. The default value is FALSE.

14.11.69 **EmbeddedServer.excludeRelay**

This settings excludes particular IP addresses from the 'authorizeRelay' rule above.

Example:

<code>EmbeddedServer.excludeRelay.1</code>	<code>65.14.14.12</code>
<code>EmbeddedServer.excludeRelay.2</code>	<code>65.14.14.13</code>

14.11.70 **EmbeddedServer.SMTPIPBlackList**

This setting lists the IP addresses or subnets that are not authorized to communicate to this server's SMTP server. Any connection coming from an IP address in this list will be immediately dropped.

Example:

<code>EmbeddedServer.SMTPIPBlackList.1</code>	<code>65.14.13.122</code>
<code>EmbeddedServer.SMTPIPBlackList.2</code>	<code>65.14.13.123</code>
<code>EmbeddedServer.SMTPIPBlackList.3</code>	<code>65.14.14.</code>

(Note the final dot in the third line)

14.11.71 **EmbeddedServer.acceptSMTPAuthentication**

If this flag is set to TRUE, DeskNow will accept SMTP authentication to authorize mail relay even from IP addresses not specifically authorized (see 14.11.67).

In other words, if this flag is TRUE users from anywhere will be able to send emails to any address on the internet, by setting their own DeskNow username and password in the email client connection properties. For example, in Microsoft Outlook these settings are under the "Outgoing Mail Server: My server requires authentication" section of the mail account.

Users of virtual domains (i.e. domains other than the default) must authenticate using `username@domainname` (2 dash characters) as their SMTP username (ex.: `joe@company1.com`).

For installation performed after release 2.2, this flag is TRUE by default.

14.11.72 **EmbeddedServer.AddReceivedHeaderForAuthorizedSenders**

If this flag is set to TRUE, DeskNow will add a "Received" header to mail messages received from a sender that is authorized to relay. If the flag is FALSE, the header will be omitted. Omitting the header provides less information about the source of the message to the receiving party. The header is always added to messages received by external senders that are not authorized to relay.

This flag is TRUE by default.

14.11.73 **EmbeddedServer.verifySMTPIdentity**

If this flag is set to TRUE *and* `EmbeddedServer.acceptSMTPAuthentication` is set to TRUE, DeskNow will check the identity of the sender before allowing him/her to send emails.

In other words, if a user authenticates as 'joe' and then tries to send an email where the From field is 'mark@xyz.com', then DeskNow will reject the message and will write a warning message in the security log file.

14.11.74 EmbeddedServer.SMTPForwardHost

This setting instructs DeskNow to forward all the outgoing emails (i.e. those that are not considered to be local) to another SMTP server, instead of trying to deliver them directly to the recipients' mail servers. Specify the address of the external SMTP server to use. This setting is disabled by default. Also note that DeskNow will still deliver to the local inboxes messages that it considers 'local', i.e. directed to one of its users.

14.11.75 EmbeddedServer.SMTPForwardPort

This setting specifies the port used to connect to the forward SMTP server. It is ignored if EmbeddedServer.SMTPForwardHost is not used. The default is 25.

14.11.76 EmbeddedServer.SMTPForwardUsername

This setting specifies the username used to connect to the forward SMTP server, if it requires SMTP authentication. It is ignored if EmbeddedServer.SMTPForwardHost is not used. The default is to not use SMTP authentication to send emails to the forward server. If you specify this setting, you need to specify the password as well (see below).

14.11.77 EmbeddedServer.SMTPForwardPassword

This setting specifies the password used to connect to the forward SMTP server, if it requires SMTP authentication. It is ignored if EmbeddedServer.SMTPForwardHost is not used, or if EmbeddedServer.SMTPForwardUsername is not used.

14.11.78 EmbeddedServer.useRemoteDeliveryForExistingLocalAccounts

When this setting is TRUE, DeskNow will use remote delivery or forwarding even for mail addressed to accounts that exist in the local DeskNow system. This option is useful when an external server is handling all mail for your domain, and DeskNow will use it as a central point (optionally getting mail back via External POP3). The default setting is FALSE, which means that if an email is directed to an existing local account, the email is processed locally.

14.11.79 EmbeddedServer.useRemoteDeliveryForNonExistingLocalAccounts

When this setting is TRUE, DeskNow will accept mail for a local domain even if the specific mailbox is not present locally. It will then route mails to such mailboxes using the normal routing policy (MX lookup, or SMTP forwarding). This setting is useful when DeskNow hosts only some of the addresses of a domain, and an external server (typically the SMTP forward server) hosts more mailboxes. Also see 14.11.31 to disable the check of local usernames in the webmail interface. The default setting is FALSE, which means that if an email is directed to a local domain, but the recipient mailbox does not exist, the email is rejected immediately.

14.11.80 EmbeddedServer.SynchronizeWebmailWithPOP3Read

If this flag is TRUE and the DeskNow POP3 server is enabled, DeskNow will mark messages as read in the WebMail interface whenever they are downloaded by a POP3 client. This helps synchronizing the status of messages, so that users know which messages they have already read.

NB there is no way to know when a message has been actually READ by the user in a POP3 client. If the flag is set to TRUE DeskNow will mark the message as read when the POP3 client DOWNLOADS the message.

Possible values are TRUE (enable) or FALSE (disable).

The default value for this flag, for new installations created using release 1.3 or above, is TRUE.

14.11.81 EmbeddedServer.SynchronizeWebmailWithPOP3Delete

If this flag is TRUE and the DeskNow POP3 server is enabled (or ForcelInboxStore is TRUE), DeskNow will delete messages from the WebMail interface whenever they are deleted from the inbox by a POP3 client. This helps synchronizing the status of messages, especially in situations when POP3 is the main access method, but users need access to webmail when they are out of the office.

The deletion works also in the other direction: when the user deletes a message using the webmail interface, the message is deleted from the POP3 inbox.

Possible values are TRUE (enable) or FALSE (disable).

The default value for this flag is TRUE.

14.11.82 EmbeddedServer.DetectTimeLimitedEmailAddresses

If this flag is TRUE and the DeskNow email server is enabled, DeskNow will detect and filter out time limited email addresses, to prevent spam.

See the DeskNow online help for a complete description of time-limited email addresses.

Possible values are TRUE (enable) or FALSE (disable).

The default value for this flag is TRUE.

14.11.83 EmbeddedServer.DNSServer.x

DeskNow usually automatically detects the address of the DNS servers available. In some rare circumstances, however, it does not detect the correct addresses, or the DNS servers detected are just DNS proxies, which do not give full support of mail record (MX) lookups.

You can use this property to manually set the DNS servers to use.

Example:

EmbeddedServer.DNSServer.1 192.168.1.12

You can indicate multiple DNS servers, for redundancy:

EmbeddedServer.DNSServer.2 192.168.1.13

...etc

14.11.84 EmbeddedServer.EnableReverseDNS

If this flag is set to TRUE, whenever the DeskNow SMTP server receives a connection, it will perform a reverse DNS lookup to try to find the hostname of the connecting computer (based on its IP address). This operation may slow down the SMTP operations, if reverse DNS is not set up correctly, but it can be useful in finding out about the origin of spam email, as the extra information will be included in the headers.

The default value is FALSE.

14.11.85 EmbeddedServer.EnableBounceMessages

If the embedded mail server is not able to deliver a message (unable to connect to the recipient SMTP server, or the recipient SMTP answers with an error, such as recipient not found, etc.), it usually “bounces” a Message Delivery Error email back to the sender, to inform of the problem, and retries the delivery at a later time.

If this flag is set to TRUE, the bounce message is sent, whereas if this flag is set to FALSE no bounce message is sent, and the sender is not informed of the problem. Delivery errors are in any case logged in the log files.

The default value is TRUE (send the bounce messages).

14.11.86 EmbeddedServer.EnableFirstBounceMessage

Enables (TRUE) / disables (FALSE) the bounce message that is sent when the first delivery attempt fails. If this setting is TRUE and DeskNow could not deliver the message just sent, a bounce message is generated. The delivery will then be retried for a number of times specified in the configuration. If all retry attempts fail, a second, final, bounce message is generated. If this setting is false, only the final bounce message (if necessary) will be sent. The default value is TRUE.

NB if EnableBounceMessages is FALSE, no bounce messages (first or final) will be sent at all, regardless of this setting.

14.11.87 EmbeddedServer.POP3SSLKeyStore

The keystore file containing the SSL certificate to use for encrypted POP3 communications with mail clients. This is set by default to be the file `\desknowdata\ssl\pop3keystore`. See section 9.1.7.

14.11.88 EmbeddedServer.POP3SSLKeyStorePassword

The password for the POP3 SSL keystore. The default value is “changeit”.

14.11.89 EmbeddedServer.SMTPSSLKeyStore

The keystore file containing the SSL certificate to use for encrypted SMTP communications with mail clients. This is set by default to be the file `\desknowdata\ssl\smtp3keystore`. See section 9.1.7.

14.11.90 EmbeddedServer.SMTPSSLKeyStorePassword

The password for the SMTP SSL keystore. The default value is “changeit”.

14.11.91 EmbeddedServer.IMAP.Enable

IMAP is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See chapter 16 for information on how to purchase a DeskNow license.

Enables (TRUE) / disables (FALSE) the IMAP server.

14.11.92 EmbeddedServer.IMAP.MaxConnections

Maximum number of concurrent connections. Note that some IMAP clients may create more than one connection per account.

14.11.93 EmbeddedServer.IMAP.Plain.Enable

Enables (TRUE) / disables (FALSE) the IMAP service on unencrypted connections.

14.11.94 EmbeddedServer.IMAP.Plain.BindInterface

Defines the interface on which the IMAP server will accept unencrypted connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.IMAP.Plain.BindInterface 162.4.12.65

14.11.95 EmbeddedServer.IMAP.Plain.Port

Defines the port on which the IMAP server will accept unencrypted connections. The default value is 143, if you change this you will need to change the default value in the mail clients that connect to this service.

14.11.96 EmbeddedServer.IMAP.SSL.Enable

Enables (TRUE) / disables (FALSE) the IMAP service on encrypted connections.

14.11.97 EmbeddedServer.IMAP.SSL.BindInterface

Defines the interface on which the IMAP server will accept unencrypted connections. If no interface is specified (the default), the server will accept connections on every network interface.

Example:

EmbeddedServer.IMAP.SSL.BindInterface 162.4.12.65

14.11.98 EmbeddedServer.IMAP.SSL.Port

Defines the port on which the IMAP server will accept encrypted connections. The default value is 993, if you change this you will need to change the default value in the mail clients that connect to this service.

14.11.99 EmbeddedServer.IMAP.AuthenticatedStateTimeout

Timeout (in seconds) after which an idle connection in the IMAP authenticated state is automatically closed. The IMAP IDLE rfc recommends 30 minutes, however Outlook mail clients do not refresh the connection every 30 minutes, so the recommended settings would see them getting disconnected and not notifying users of new emails. Therefore the default is set to 10 hours.

14.11.100 EmbeddedServer.IMAP.NonAuthenticatedStateTimeout

Timeout (in seconds) after which an idle connection in the IMAP non-authenticated state (i.e. before a successful login) is automatically closed. The default is set to 90 seconds.

14.11.101 EmbeddedServer.IMAPSSLKeyStore

The keystore file containing the SSL certificate to use for encrypted IMAP communications with mail clients. This is set by default to be the file `\desknowdata\ssl\imapkeystore`. See section 9.1.7.

14.11.102 EmbeddedServer.IMAPSSLKeyStorePassword

The password for the IMAP SSL keystore. The default value is "changeit".

14.11.103 EmbeddedServer.ExternalSMTPFilter.enable

Enables (TRUE)/ disables (FALSE) filtering done by an external SMTP-based mail filter. This is equivalent to SMTP-based filtering in Postfix (http://www.postfix.org/FILTER_README.html). When this option is enabled, you must also specify host and port of the filter server (see below). With external SMTP filtering DeskNow can use any external SMTP-based filter that works with Postfix. Note that the external SMTP filter does not need to send messages back to DeskNow using a different SMTP port. It can simply use the normal SMTP port used by DeskNow. Just make sure that DeskNow is configured to authorize mail relay for the IP address that the SMTP filter will be sending from, and configure the IPmatch parameter (see below).

14.11.104 EmbeddedServer.ExternalSMTPFilter.host

Hostname or IP address of the external SMTP filter.

14.11.105 EmbeddedServer.ExternalSMTPFilter.port

TCP port to use to connect to the external SMTP filter. The default is 10025.

14.11.106 EmbeddedServer.ExternalSMTPFilter.IPmatch

The IP address that the SMTP filter will send mail to from. This is typically 127.0.0.1 if the SMTP filter is on the same server as DeskNow, and sending mail to localhost. DeskNow will consider mail coming from this IP address as already filtered, and will not send it to the SMTP filter, avoiding a loop.

14.11.107 EmbeddedServer.BlockLocalMailFromSpoofing

If this parameter is TRUE, DeskNow will not accept incoming mail via SMTP with the 'MAIL FROM:' of a local address, if the connection is not authorized to relay. This helps blocking address spoofing.

14.11.108 EnableBackgroundClean

DeskNow periodically can empty the trash and spam folders of user accounts (ex. "delete messages from the 'deleted' folder after 7 days"). If this setting is set to TRUE, the background cleaning is performed. If this setting is set to false, the background cleaning is not performed, and the preference is not available to users. Very large ISPs should set this to FALSE, as it is a feature that can be resource intensive. The default value is TRUE.

14.11.109 OutputCharset

The default character encoding to use when encoding mail before sending them via the web interface. Ideally you would want to use UTF-8, which can encode every character of every known language. Unfortunately some old mail readers don't understand UTF-8.

The following character sets can be used with DeskNow:

Big5
Big5-HKSCS
EUC-CN
EUC-JP
euc-jp-linux
EUC-KR
EUC-TW

GB18030
GBK
ISCII91
ISO-2022-CN-CNS
ISO-2022-CN-GB
ISO-2022-KR
ISO-8859-1
ISO-8859-13
ISO-8859-15
ISO-8859-2
ISO-8859-3
ISO-8859-4
ISO-8859-5
ISO-8859-6
ISO-8859-7
ISO-8859-8
ISO-8859-9
JIS0201
JIS0208
JIS0212
Johab
KOI8-R
Shift_JIS
TIS-620
US-ASCII
UTF-16
UTF-16BE
UTF-16LE
UTF-8

DeskNow normally attempts to detect the correct character set from your system, but you can override it by setting one directly. Users can also choose their favourite character set individually.

14.11.110 InputCharset

Defines the character set used to decode received emails when it is not specified in the mail headers.

Some mail servers and mail clients (especially used in East Asia) are not standard compliant, and do not encode headers even when sending emails with non-latin characters.

Example:
InputCharset EUC-KR

The default value is ISO-8859-1 .

14.11.111 GuessInputCharset

Some mail servers and mail clients (especially used in East Asia) are not standard compliant, and do not encode headers even when sending emails with non-latin characters.

If this setting is enabled (TRUE), DeskNow will attempt to *guess* the appropriate charset for a mail header, from its byte content. If it fails, it will resort to the InputCharset setting above (if present).

The default value is FALSE.

14.11.112 AntiSpam.Bayes.use

Enables (TRUE) or disables (FALSE) bayesian analysis to classify emails (see section 4.2.3). The default is TRUE.

14.11.113 AntiSpam.Bayes.EnableLearning

Enables (TRUE) or disables (FALSE) the possibility to train the Bayesian engine.

The default is TRUE. It is also possible to enable / disable engine training for particular user classes (see 3.1.6).

14.11.114 AntiSpam.Bayes.DataFile

Specifies the location where the bayesian engine stores its parameters. If this file does not exist, DeskNow will store in its place a default parameters file. This is useful if you want to re-set the parameters.

14.11.115 AntiSpam.Bayes.MaxTokens

Specifies the maximum number of tokens that the engine should keep in the datafile. This parameter is used to prevent the datafile growing indefinitely. The default value is 50,000.

14.11.116 AntiSpam.Bayes.DefaultProbabilityThreshold

Defines the default probability threshold (see section 4.2.3) to be set for new accounts. This value must be a number between 0 and 1 (with 0 and 1 most likely to be useless values).

The default value is 0.7.

14.11.117 AntiSpam.DNSBL.Service.x

Defines 0 or more DNSBL services that DeskNow will query to determine if the sending SMTP server is trustworthy (see **Error! Reference source not found.**). Note that since the DNSBL service uses the DNS protocol, DeskNow must have access to a DNS server. "x" is a progressive number (1, 2, etc.)

Example:

AntiSpam.DNSBL.Service.1 bl.spamcop.net

AntiSpam.DNSBL.Service.2 dnsbl.sorbs.net

14.11.118 AntiSpam.DNSBL.Reject

Determines whether the DeskNow SMTP server should immediately reject messages from blacklisted SMTP servers (TRUE), or if it should allow them through, and store them in the 'Spam' folder (FALSE).

The default value is FALSE.

14.11.119 AntiSpam.DNSBL.Reject.ExceptAllowedToRelay

Exception to the rule above – does not check DNSBL if the connection is authorized to relay (trusted IP address or authenticated sender).

The default value is TRUE.

14.11.120 AntiSpam.CopySpamToPOP3

If this parameter is TRUE, messages that DeskNow classifies as spam will be made available to POP3 clients. If it is FALSE, they will not be stored in the POP3 inbox.

The default value is TRUE (spam messages will be copied via POP3), because this is the most conservative approach. See also section 4.2.9 for spam filtering in mail clients.

14.11.121 AntiSpam.SendDailySpamSummary

If this parameter is TRUE, DeskNow will send a daily spam summary to all the users, reporting the number of spam emails detected, and reminding them to check the spam folder to see if any legitimate mail was wrongly classified as spam.

The default value is TRUE (the summary will be sent). Every user can individually disable the summary in the mail preferences.

14.11.122 AntiSpam.EnableSpamSummary

Enable (TRUE) or disable (FALSE) the spam summary, overriding any other setting. This property differs from the SendDailySpamSummary in the sense that this one blocks any possibility of sending the summary, preventing users to set a personal preference. The default is TRUE.

14.11.123 AntiSpam.DailySpamSummary.Hour

Set the hour (0-23) at which the spam report will be sent (if enabled).
The default value is 3 (AM).

14.11.124 AntiSpam.DailySpamSummary.Minutes

Set the minutes (0-59) at which the spam report will be sent (if enabled).
The default value is 0.

14.11.125 AntiSpam.POP3SubjectSpamFlag

When DeskNow classifies a message as spam, it moves it to the user's spam folder, and add the following header to the message headers:

X-Spam-Flag: YES

To further identify the message as spam to some mail clients, you can set this parameter to make DeskNow prepend some short text to the message subject.

Example:
AntiSpam.POP3SubjectSpamFlag "****SPAM****"

Note: this setting does not affect the webmail interface, but only mail clients.

14.11.126 AntiSpam.Greylisting.Enable

Enable (TRUE) or disable (FALSE) the greylisting functionality. The default is disable (FALSE).

14.11.127 AntiSpam.Greylisting.MinimumDelay

Minimum delay (in minutes) that a remote greylisted server has to wait before a delivery retry will be accepted. The default value is 10.

14.11.128 AntiSpam.Greylisting.Validity

Once a remote server is in the greylist, and the minimum delay has elapsed, messages from this IP address will be accepted for the amount of time expressed by this property (in minutes). After this amount of time, the IP address will be removed from the greylist and the next delivery attempt from this IP address will cause a fresh greylisting. The default value is 1440 (one day).

14.11.129 AntiSpam.Greylisting.ExceptionIPs

A list of IP addresses (also subnets) for which greylisting will not be applied

Example:

```
AntiSpam.Greylisting.ExceptionIPs.1 65.14.13.122
AntiSpam.Greylisting.ExceptionIPs.2 65.14.13.123
AntiSpam.Greylisting.ExceptionIPs.3 65.14.14.
```

(Note the final dot in the third line)

*This will prevent DeskNow from greylisting servers connecting from IP 65.14.13.122, 65.14.13.123 and any computer in the subnet 65.14.14.**

14.11.130 AntiSpam.Greylisting.ExplanationMessage

This is the error message sent by the DeskNow SMTP server when a message is greylisted. The text "451 4.7.1 " is prepended as it is needed by the SMTP protocol. This message should point to a web page where a more detailed explanation of greylisting is provided. The default is "Delivery delayed. For more information, see <http://www.desknow.com/451.html>"

14.11.131 AntiSpam.Greylisting.SkipIfSPFPassed

If this property is set to TRUE, greylisting is not applied to sending mail server that pass the SPF test. This helps with large multi-hosted webmail services like gmail. The default value is TRUE.

14.11.132 AntiSpam.Greylisting.IncludeClassCSubnet

If this property is set to TRUE, IP addresses are greylisted by their class C subnet (the first 3 bytes of a IPv4 address). This means that if a domain has multiple SMTP servers in a subnet, if one passes greylisting all of the others will pass it as well. The default value is TRUE.

14.11.133 AntiSpam.Filters.NoSubjectHeader

If this property is set to TRUE, messages with no 'Subject' header line will be classified as spam. Note that normal messages with an empty subject *do* have a Subject header line. The default is FALSE.

14.11.134 AntiSpam.Filters.NoFromHeader

If this property is set to TRUE, messages with no 'From' header line will be classified as spam. The default is FALSE.

14.11.135 AntiSpam.Filters.InlineImagesBiggerThan

If this property is set to a value other than -1, messages containing inline images whose size (in bytes) is larger than this number will be classified as spam. The default is -1 (inline images of all sizes are allowed).

14.11.136 AntiSpam.WhiteList.LocalDomains

If this parameter is TRUE, emails with the From header of a local (DeskNow) domain will be automatically whitelisted. It is recommended that the anti-address spoofing option (see 14.11.107) is enabled as well. This parameter is TRUE by default.

14.11.137 AntiSpam.BlockFromHeaderSpoofing

If this parameter is TRUE, emails with a From header containing a local domain, but not coming from an authenticated sender, and not coming from an IP address authorized to relay, will be classified as spam. This setting does not apply to mail retrieved to POP3 accounts – only to mail received via SMTP. If this parameter is not specified, it is assumed FALSE by default.

14.11.138 ExternalIMAP.DisableAuthPlain

If this property is set to TRUE, DeskNow will not use the PLAIN authorization mechanism to login to a remote IMAP server for importing mail, even if the remote server advertises it. Some servers (like iMail) have issues in handling this protocol. The default is TRUE.

14.11.139 CheckReverseDNSPTR.HasIt

If this flag is TRUE, DeskNow will reject mail coming from senders not authorized to relay that do not have a reverse DNS entry (PTR record) associated to their IP address. Every mail server is required to have a reverse DNS entry, and many companies (for example AOL) perform this check. The default value is TRUE for new installations since DeskNow 3.1 .

14.11.140 CheckReverseDNSPTR.HELO

If this flag is TRUE, DeskNow will reject mail coming from senders not authorized to relay that do not have a reverse DNS entry (PTR record) associated to their IP address or whose PTR record does not match the hostname declared in the HELO or EHLO SMTP command. Every mail server is required to have a matching reverse DNS entry, however most are not configured to be compliant. This setting should therefore be used with caution, as it may reject a lot of emails. The default value is FALSE.

14.11.141 CheckReverseDNSPTR.MAILFROM

If this flag is TRUE, DeskNow will reject mail coming from senders not authorized to relay that do not have a reverse DNS entry (PTR record) associated to their IP address or whose PTR record does not match the domain name declared in the MAIL FROM SMTP command. This is a very stringent requirement, to be considered experimental. Most mail servers will not comply. This setting should therefore be used with caution, as it may reject a lot of emails. The default value is FALSE.

14.12 Messenger.cfg

Defines properties for the integrated instant messaging (Jabber/XMPP) server.

14.12.1 use

Enable or disable the integrated instant messaging server.

TRUE = enable

FALSE = disable

14.12.2 PlainConnector.use

Enables or disable plain (non encrypted) XMPP connections.

TRUE = enable

FALSE = disable

14.12.3 PlainConnector.port

Defines which port the server should listen to for plain XMPP connections. The IANA registered port number for XMPP is 5222, so it is recommended to use this.

Example:

PlainConnector.port 5222

14.12.4 PlainConnector.NIO

Enable / disable the use of Java NIO (Native Input / Output). NIO is recommended for medium-to large sites (> 100 connections) because it can scale well up to thousands of connections without excessive resource requirements.

Support of Java NIO could still be experimental on some Java versions, or on some OS.

TRUE = enable

FALSE = disable

By default, NIO is disabled.

14.12.5 SSLConnector.use

Enables or disable SSL encrypted XMPP connections.

TRUE = enable

FALSE = disable

14.12.6 SSLConnector.port

Defines which port the server should listen to for SSL XMPP connections. The IANA registered port number for XMPP is 5223, so it is recommended to use this.

Example:

SSLConnector.port 5223

14.12.7 WebConnector.use

Enable or disable connections through HTTP or HTTPS. These are useful if the client is behind a firewall, and direct connections on the standard ports are not possible.

HTTP/HTTPS connections can pass through normal proxies, but are slower than direct connections. The DeskNow messenger client tries to connect using direct connections first. If it fails, it tries using web connections.

TRUE = enable
FALSE = disable

14.12.8 RosterMode

Specifies what contacts should be automatically made available to a user (i.e. what contacts should be shown to him in the “roster”). This setting can have two possible values:

- **domain** to show all the users of the domain. Users are grouped by community.
- **community** to show only users of the same community

The default value is “domain”.

Example:
`RosterMode community`

14.12.9 DefaultPresenceAlert

Sets if a ‘user abc is online’ alert is sent to other online users via instant messaging by default. The default value is TRUE. Individual users can also change this in their messenger options.

14.13 Web.cfg

Defines properties for the HTML-based interface.

14.13.1 Layouts section

Defines the available layouts, and the default layout to be used

Layouts are defined via a set of JSP files.

Each layout resides in its own subfolder under the '/jsp' folder.

For instance, the files making the layout 'DeskNow Original' can be stored in /jsp/original

Example

Layout.1.Name "DeskNow original"

Layout.1.Id original

Layout.2.Name "DeskNow noframes"

Layout.2.Id noframes

Layout.Default original

14.13.2 Custom section

This section contains custom settings that are used by web designers in specific layouts.

As default, the only setting in this section defines the default color theme for the "original" layout.

Example

Custom.original.DefaultTheme orange

14.13.3 Custom.original.DefaultLongDatePattern

Specifies the default format for dates (when using the long format) to be used for new users.

This by default produces dates like "Friday, March 7 2003" (American convention).

You can set a different default date format. The syntax follows the convention of the Java class `java.text.SimpleDateFormat` (see <http://java.sun.com/j2se/1.4.1/docs/api>).

A typical default value, if you prefer the European convention with the day before the month, is the following:

Custom.original.DefaultLongDatePattern "EEEE, d MMMM yyyy"

NB. users can change this setting in their Preferences.

If you choose a non-standard date format, you should also check that it is

14.13.4 Custom.original.DefaultShortDatePattern

Specifies the default format for dates (when using the short format) to be used for new users.

This by default produces dates like "03/07/2003" for the 7th of March 2003 (American convention).

You can set a different default date format. The syntax follows the convention of the Java class `java.text.SimpleDateFormat` (see <http://java.sun.com/j2se/1.4.1/docs/api>).

A typical default value, if you prefer the European convention with the day before the month, is the following:

*Custom.original.DefaultShortDatePattern**"dd/MM/yy"*

NB. users can change this setting in their Preferences.

14.13.5 Custom.original.DefaultTimePattern

Specifies the default format for time to be used for new users.

This by default produces dates like "5:30 PM" (American convention).

You can set a different default time format. The syntax follows the convention of the Java class `java.text.SimpleDateFormat` (see <http://java.sun.com/j2se/1.4.1/docs/api>).

A typical default value, if you prefer the European convention of 24-hour notation, is the following:

*Custom.original.DefaultTimePattern**"HH:mm"*

NB. users can change this setting in their Preferences.

14.13.6 Custom.original.DefaultFontSize

Specifies the default font size to be used for new users.

Please see the table below for DeskNow's convention:

xx-small	9px
x-small	6pt
small	8pt
medium	9pt
large	10pt
x-large	12pt

Example:

*Custom.original.DefaultFontSize**"10pt"*

NB. users can change this setting in their Preferences.

14.13.7 Custom.LeftMenuWidth

Set the default width of the left-side menu tree (measured in pixels). The default is 170.

14.13.8 Branding.ApplicationName

Defines the name of the application shown in various parts of the web interface. Also note that the title bar of the browser is changed *per-domain* via the web administration interface. The default value is "DeskNow".

14.13.9 Branding.PocketPCTitle

Defines the name of the application shown in the PocketPC interface. Also note that the title bar of the browser is changed *per-domain* via the web administration interface. The default value is "DeskNow".

14.13.10 Branding.LinkMain

Defines the link that is used when the user clicks on the application name (typically in the home page, and in the menu). The default value is "<http://www.desknow.com>". Make sure to use the full URL, including the protocol (ex. <http://>).

14.13.11 Branding.LinkSupport

Defines the link that is used when the user clicks on the Get support link. The default value is "http://www.desknow.com/support.html". Make sure to use the full URL, including the protocol (ex. http://).

14.13.12 LoginPage.OpenInNewWindowByDefault

If this setting is TRUE, the default option in the login page is set to open the home page in a new window. If it is FALSE, the default option is to stay in the same window. After the first login, a cookie is saved in the client browser with the setting chosen by the user.
Default: FALSE

14.13.13 EnableTips

If this setting is TRUE, startup tips are enabled. Users can disable them. If it is FALSE, tips are not shown, and the user cannot choose to enable them. The default value is TRUE.

14.13.14 EnableAutoRefresh

Enables (TRUE) / disables (FALSE) the auto refresh feature. If this setting is true, by default the browser will connect to the server every 10 minutes to check for new emails, refresh the calendar, etc. This also prevents session timeouts due to inactivity. This however uses more resources on the server, especially with very large user bases. If this setting is false, auto refresh are not enabled. Note that the when emails arrive, if the user has IM alerts enabled, the browser will refresh anyway. In addition, the user can always perform a manual refresh by clicking the 'Check' link.
The default value is TRUE.

14.13.15 ProxiedIPHeader

When DeskNow is running behind an internal proxy (such as Apache mod_proxy), the HTTP requests are coming from the proxy itself.

To ensure that the client's IP address is used in the logs, and in all the reports, you can specify the name of an HTTP request header where the proxy (as most do) writes the client's *real* IP address.

For Apache mod_proxy, this header is ProxiedIPHeader.
So when using DeskNow with Apache mod_proxy a typical line would be:

ProxiedIPHeader X-Forwarded-For

14.13.16 NumFileUploadFields

Sets how many files can be uploaded in a single operation using the web interface. In other words, this number controls how many file input fields are shown in forms that allow the uploading of files (normal files, or attachments).
The default is 5.

14.14 International.cfg

This file contains all the necessary setting for the access of DeskNow in multiple languages.

See section 12 for a full overview of language support in DeskNow.

14.14.1 DefaultLanguage

Defines the default language assigned to new user accounts when they login. Users can choose a different language from the Preferences page.

The value of this setting must be the language code of one of the languages defined. See section 12 for more details.

14.14.2 Spelling.DefaultLexicon

Defines the default dictionary to use for spell-checking. Users can choose a different default dictionary from the Preferences page.

The following table illustrates the possible values:

Property value	Dictionary
DE	German
DK	Danish
EN-CA	English (Canada)
EN-UK	English (UK)
EN-US	English (USA)
ES	Spanish
FI	Finnish
FR	French
IT	Italian
NL	Dutch
NO	Norwegian
PT	Portuguese
PT-BR	Brazilian Portuguese
SV	Swedish

14.14.3 Spelling.UseTechLexicon

Enables (TRUE) / disables (FALSE) the inclusion of common technology terms in the spelling dictionary. The default value is TRUE (enable).

14.15 Backup.cfg

This file contains all the settings to control how DeskNow performs data backups. All the settings can also be changed by the Administrator via the web interface.

14.15.1 Automatic

Enable (TRUE) or disable (FALSE) automatic scheduled backups at a given time. Backups can also be started manually via the web interface.

Default: TRUE (for new

14.15.2 Hour

The hour of the day at which scheduled backups (if enabled) should start. The hour must be a number between 0 and 23.

14.15.3 Minute

The minute at which scheduled backups (if enabled) should start. Must be a number between 0 and 59.

14.15.4 Day.x

Determines on which days of the week scheduled backups (if enabled) are performed. "x" is a number between 0 and 6, where 0 is Sunday, 1 is Monday, etc.

Example:

To enable backups only Monday to Friday, use the following:

Day.0 FALSE

Day.1 TRUE

Day.2 TRUE

Day.3 TRUE

Day.4 TRUE

Day.5 TRUE

Day.6 FALSE

14.15.5 BasePath

Base directory where to store backups. For each backup, a directory with the full date and time will be created under this directory.

By default, backups are created in the directory `\desknowdata\backup`. DeskNow saves each backup as a subdirectory with a different name (ex. 20030725_0934 for a backup created on July 25th, 2003 at 9:34 AM).

It is recommended to change this parameter and point it to a network drive, so that backups are created on a different computer and drive.

14.15.6 EmbeddedDatabase

Enable (TRUE) or disable (FALSE) the backup of the embedded McKoi database when a backup operation is performed (either manually or scheduled).
Default: TRUE.

14.15.7 UserFolders

Enable (TRUE) or disable (FALSE) the backup of the userfolders directory. This directory contains all the files and attachments of all the users. NB carefully consider whether to backup this directory or not, since it can be very big.
Default: FALSE.

14.15.8 MailQueue

Enable (TRUE) or disable (FALSE) the backup of the mail queues directory. This directory contains all the mail inboxes (used for external mail clients), messages in the retry queue, etc. NB carefully consider whether to backup this directory or not, since it can be very big.
Default: FALSE.

14.15.9 CfgFiles

Enable (TRUE) or disable (FALSE) the backup of the directory where configuration files are usually kept. This includes the license keys.
Default: FALSE.

14.15.10 KeepCopies

This parameter sets how many past backups to keep. DeskNow saves each backup as a directory with a different name (ex. 20030725_0934 for a backup created on July 25th, 2003 at 9.34 AM). To recover disk space, DeskNow will only keep the latest backups, and delete the older ones. This parameter defines how many of the latest backups to keep.
Default: 3.

14.16 SelfRegistration.cfg

This file contains all the settings for the self registration process (see section 10). Self registration is disabled by default, for security reasons. You can enable it by editing this file.

14.16.1 AllowUserSelfRegistration

Enables (TRUE) or disables (FALSE) user self registration.
Default: FALSE

14.16.2 AllowCommunitySelfRegistration

Enables (TRUE) or disables (FALSE) community self registration.
Default: FALSE

14.16.3 AllowDomainSelfRegistration

Enables (TRUE) or disables (FALSE) domain self registration.
Default: FALSE

14.16.4 DefaultUserDiskQuota

Specifies the disk quota to be assigned to new user accounts created via self-registration. This applies to user self registration and community self-registration, which involves the creation of the manager user account. The disk quota is expressed in bytes. Specify -1 for unlimited disk quota.

14.16.5 DefaultCommunityDiskQuota

Specifies the disk quota to be assigned to new communities created via self-registration. This applies to community self-registration. The disk quota is expressed in bytes. Specify -1 for unlimited disk quota.

14.16.6 DefaultDomainDiskQuota

Specifies the disk quota to be assigned to new domains created via self-registration. This applies to domain self-registration. The disk quota is expressed in bytes. Specify -1 for unlimited disk quota.

14.16.7 DefaultDomainMaxUsers

Specifies the maximum number of users (including the domain administrator) for this domain. The domain administrator will not be allowed to create more users than the number specified here. Specify -1 for no limits. The default value, if this parameter is omitted, is -1.

14.16.8 DefaultDomainMaxSyncMLUsers

Specifies the maximum number of users of the SyncML functionality (including the domain administrator) for this domain. Specify -1 for no limits. The default value, if this parameter is omitted, is -1.

14.16.9 DefaultUserClass

Specifies the default user class (see section 3.1.6) to be used for an user account created via self-registration. This applies to user self registration and community self registration. Note that this does not apply to domain self-registration, because a new domain has only the Normal user class by default.

Also note that a community manager can create other users, assigning them one of the user classes defined in the domain.

Default: Normal

14.16.10 DefaultDomain

The default domain in which new user accounts and communities will be created. This applies to user self-registration and community-selfregistration.

This domain must exist.

Default: mydomain.com

14.16.11 DefaultCommunity

The default community in which new user accounts will be created. This applies to user self-registration only.

This community must exist, in the DefaultDomain specified.

Default: Default

14.16.12 Trial.EnableExpiration

Enables (TRUE) / disables (FALSE) automatic expiration of user accounts/communities/domains created with the self-registration mechanism. The default is FALSE (disable).

14.16.13 Trial.ExpireAfterDays

Number of days from creation that an account/community/domain created with self-registration will *expire*. An expired account is no longer accessible (user cannot login), but will still receive email, and can be enabled again. This makes sure that the customer has a last opportunity to fully subscribe to the service without any data loss.

Before expiring an account, DeskNow will send an email warning. It is possible to disable scheduled expiration of an account/community/domain by editing the corresponding object properties in the Administration panel.

14.16.14 Trial.DeleteAfterDays

Number of days from creation that an account/community/domain created with self-registration will *be deleted*. After an account is deleted, all its data is lost (unless stored in

backups). Before deleting an account, it is typically expired (see above), and before being expired, a notification message is sent. It is possible to disable scheduled deletion of an account/community/domain by editing the corresponding object properties in the Administration panel.

14.16.15 Trial.NotifyBeforeExpirationDays

Before expiring an account, DeskNow will send an email warning. This parameters sets how many days before expiration the warning is sent. For expiring user accounts, the email will be sent to the account itself. For expiring communities, the email will be sent to the community manager. For expiring domains, the email will be sent to the domain administrator.

14.16.16 Trial.UserNotificationSubject

When sending a warning email about pending account expiration, DeskNow will use this as the subject of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the account will expire.

14.16.17 Trial.UserNotificationMessage

When sending a warning email about a pending account expiration, DeskNow will use this as the text of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the account will expire.

14.16.18 Trial.CommunityNotificationSubject

When sending a warning email about a pending community expiration, DeskNow will use this as the subject of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the community will expire.

14.16.19 Trial.CommunityNotificationMessage

When sending a warning email about a pending community expiration, DeskNow will use this as the text of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the community will expire.

14.16.20 Trial.DomainNotificationSubject

When sending a warning email about a pending domain expiration, DeskNow will use this as the subject of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the domain will expire.

14.16.21 Trial.DomainNotificationMessage

When sending a warning email about a pending domain expiration, DeskNow will use this as the text of the email. The text can contain the string %EXPIRY_DATE% , which will be replaced with the actual date in which the domain will expire.

14.17 Clustering.cfg

14.17.1 Enable

Enable (TRUE) or disable (FALSE) support for clustering and load balancing. See section 13 for more details.

14.17.2 ConfigurationReloadCheckPeriod

When changing configuration properties in the Administration console, changes are stored to the shared configuration files. The changes are applied immediately to the server that handled the HTTP operation. The other servers will periodically check the configuration files to see if they've changed, and if so they will reload the properties automatically. Note that some properties require the servers to be restarted (these are clearly marked in the Administration console). In addition, all of the properties that do not appear in the console, but are available only in the configuration files, require the servers to be restarted.

14.17.3 Heartbeat.MulticastGroup

The multicast address to use. NB this does not need to be a valid IP on your network. Valid multicast addresses are between 224.0.0.1 and 239.255.255.255 . The default is 231.34.192.18

14.17.4 Heartbeat.MulticastPort

The UDP multicast port. The default is 6732.

14.17.5 Heartbeat.NetworkInterface

The network interface to bind to. The default is all interfaces.

14.17.6 Heartbeat.Interval

The interval at which DeskNow must emit heartbeats. This value is expressed in milliseconds. The default value is 20000.

14.17.7 Heartbeat.Validity

If a server does not receive heartbeats from another server for more than this time, it considers the other server as down and no longer part of the cluster.

This value is expressed in milliseconds.

This value must be at least 4*Interval, to allow transient network problems.

The default value is 100000.

14.18 SyncML.cfg

14.18.1 ShowAllEvents.StoreName

Set the name for the store that lists all the events in all calendars. The default is "events", which matches the default setting on most phones.

14.18.2 ShowAllTasks.StoreName

Set the name for the store that lists all the tasks in all calendars. The default is "tasks", which matches the default setting on most phones.

14.18.3 ShowAllCalendars.StoreName

Set the name for the store that lists all the events and tasks in all calendars. Some devices do not have the settings to separate events and tasks, so can only use a single store. For these devices, users should use this store.

The default is "calendars".

14.18.4 ShowAllContacts.StoreName

Set the name for the store that lists all the contacts in all folders. The default is “contacts”, which matches the default setting on most phones.

14.18.5 AddTimeZoneInfoForPDAs

Set the default preference value that determines whether DeskNow should include the TZ vCal property (local timezone) in events sent to PDAs and mobile phones. Setting this to TRUE enables correct syncing even when the PDA is set to a different timezone than the user account. However some mobile phones have problems with this property and will sync with an incorrect time. The default value is TRUE. Users can individually change this preference in the Preferences/SyncML page.

14.18.6 UseZuluTimeForPDAs

Set the default preference value that determines whether DeskNow should use ‘zulu’ (GMT) times when syncing with PDAs and mobiles. Most devices are not timezone-aware, and so using zulu times would result in incorrect event times on the phone. However some devices do support timezones, and cannot work properly without zulu times. This property is set to FALSE if not specified. Every user can change the setting for his own account in Preferences/SyncML.

14.18.7 DefaultPDASyncWithMyCalendar

If set to TRUE, when a new account is created it is automatically set to sync the ‘My Calendar’ tasks and events with PDAs. The default is FALSE.

14.18.8 DefaultPDASyncWithContacts

If set to TRUE, when a new account is created it is automatically set to sync the ‘Contacts’ folder with PDAs. The default is FALSE.

14.18.9 AutoExpiryOfDeleteRecordsDays

Number of days after which info on deleted items is cleared from the SyncML server. When an item (ex. an event) is deleted, the SyncML server needs to keep track of its deletion so that when SyncML clients connect it can notify them of the deletion. This setting defines for how long this record should be kept. It should be higher than the number of days between syncs with clients. The default value is 90 days – all is fine as long as sync clients connect at least every 90 days.

14.18.10 LogSession.Enable

Enable/disable logging of raw data of SyncML sessions. When this option is enabled (and the other LogSession parameters are set, see below), DeskNow will write to log files all data exchanged with SyncML clients at http level. This is normally used only to provide debug information to the developers.

14.18.11 LogSession.BaseFolder

Path to a folder where log files for SyncML sessions will be written to.

14.18.12 LogSession.SessionValidity

Validity (in seconds) of a SyncML session for log purposes. If a gap of a longer time is detected between the response sent by DeskNow and a new request coming from the client, the new request is considered part of a new session and will be logged to a new log file.

15 Administration API

The administration API, except for the Internationalization Refresh command, is available for the default DeskNow Lite license only for a trial period of time (30 days from the date of installation). After this period, it will not be available. See section 16 for information on how to purchase a DeskNow license.

However, some commands are always enabled, to ease the life of every system administrator. These commands are: Re-create user directory, Recalculate mail folder statistics, Reload I18N files, Emulate the expiration of the trial period.

15.1 Overview

DeskNow offers an API (Application Programming Interface) that allows to automate many system operations via an external program/script. For instance, it is possible to automatically create users, verify their username/password, send instant messages, post announcements, etc.

The API is completely based on HTTP, so that it can be used by virtually any scripting or programming language.

An API invocation is simply an HTTP request (GET or POST), like this:

http://www.mydomain.com/desknow/admin?pwd=password&action=im_sendalert&username=joe&domain=mydomain.com&message=emergency%20evacuation&alertcode=1

This invocation sends an instant messaging alert to user joe@mydomain.com.

You can test API invocations by typing them directly into a web browser, or creating simple web forms to input the various fields.

URL Encoding: when using the GET method (i.e. when all the request parameters are in the URL), the space character needs to be expressed as %20, as per HTTP protocol specifications. Other special characters may need to be encoded as well. For a useful encoding tool, see <http://www.blooberry.com/indexdot/html/topics/urlencoding.htm>.

The **pwd** parameter must always be provided, and the value must be the password of the admin user (of the default domain).

All the parameter names are case-sensitive.

It is also possible to use the SSL (https) protocol for the API invocations, for increased security.

15.1.1 Invocation result

If the operation is successful, DeskNow will return an HTTP 200 response status, and will also return a more descriptive text in the response body.

If the operation is unsuccessful, the HTTP response status will be different from 200, and a descriptive text will be included. Note that by default Internet Explorer is configured to 'Show friendly HTTP error messages' which will hide the response text in case of an error. You can disable that setting in the Tools/Options/Advanced panel, or use a different browser.

15.2 Create user

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_createuser	Yes
username		Yes
password		Yes
domain		No. Use default domain if omitted.
communityname	Name of the community for the new user	Yes
diskquota	"Soft" disk quota, in bytes. Note: 10Mb = 10*1024*1024	No. Unlimited disk quota if omitted.
userclass	The name of the user's class	No. Assume Normal if omitted.
firstname		No
middlename		No
lastname		No

15.3 Create community

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_createcommunity	Yes
managerusername		Yes
communityname		Yes
domain		No. Use default domain if omitted.
diskquota	"Soft" disk quota, in bytes. Note: 10Mb = 10*1024*1024	No. Unlimited disk quota if omitted.

15.4 Create domain

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_createdomain	Yes
domain		Yes
adminpassword	Password of the admin user for the new domain	Yes
diskquota	"Soft" disk quota, in bytes. Note: 10Mb = 10*1024*1024	No. Unlimited disk quota if omitted.
maxusers	Maximum number of users allowed for this domain (Requires ASP license).	No. Unlimited users if omitted.
maxsyncmlusers	Maximum number of users of the syncml functionality allowed for this domain (Requires ASP license).	No. Unlimited users if omitted.

Note: this method will create automatically the admin user for the new domain.

15.5 Delete domain

IMPORTANT: use this command with caution. The domain will be deleted immediately, with all the users in it and their data. No confirmation will be asked, since this is meant to be used as a batch command. This command cannot be undone.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_deletedomain	Yes
domain	Name of the domain	Yes. It is not possible to delete the default domain.

15.6 Delete community

IMPORTANT: use this command with caution. The community and all user accounts in it will be deleted immediately, with all their data. No confirmation will be asked, since this is meant to be used as a batch command. This command cannot be undone.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_deletecommunity	Yes
community	Name of the community	Yes
domain		No. Use default domain if omitted.

15.7 Delete user

IMPORTANT: use this command with caution. The user account will be deleted immediately, with all its data. No confirmation will be asked, since this is meant to be used as a batch command. This command cannot be undone.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_deleteuser	Yes
username		Yes
domain		No. Use default domain if omitted.

15.8 Update user

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_updateuser	Yes
username	The username of the user to change.	Yes
domain		No. Use default domain if omitted.
password	The new password to set for the	No. Unchanged if omitted.

	user. NB do not use this field if you are using external authentication.	
diskquota	"Soft" disk quota, in bytes	No. Unchanged if omitted.
userclass	The name of the user's class	No. Unchanged if omitted.
newusername	The new username (if changing). It is highly recommended that the user is not currently logged in while this operation is performed.	No. Unchanged if omitted.
firstname		No. Unchanged if omitted.
middlename		No. Unchanged if omitted.
lastname		No. Unchanged if omitted.

15.9 Create user class

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_createclass	Yes
domain		No. Use default domain if omitted.
name	Name of the user class to create	Yes.
use_email	TRUE if enabled. FALSE (or just omit) if disabled.	No
my_calendar		No
create_calendar		No
share_object		No
use_external_pop3		No
create_file_root		No
create_contact_folder		No
create_message_board		No
change_password		No
create_mail_folder		No
edit_mail_filter		No
edit_preferences		No
use_calendars		No
use_messenger		No
use_files		No
use_contacts		No
use_messageboards		No
learn_antispam		No
send_emailoutside		No
edit_personaldetails		No
use_synclml		No
edit_identities		No
use_rss		No

15.10 Add user to user group

Parameter name	Description	Required
pwd	Password of the admin user	Yes

action	user_addtogroup	Yes
username	The username of the user to change.	Yes
domain		No. Use default domain if omitted.
name	Name of the user group to add this user to. The group must exist in the user's domain.	Yes.

15.11 Authenticate user

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_authenticate	Yes
username	Username of the user to be authenticated	Yes
password	Password of the user to be authenticated	Yes
domain	Domain of the user to be authenticated	No. Use default domain if omitted.

The HTTP response will have a status 200 (OK) if the username/password combination is authenticated, or 403 (Forbidden) if it is not valid.

15.12 Create calendar

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	calendar_createcalendar	Yes
username	Username of the user that will own the calendar	Yes
domain	Domain of the user	No. Use default domain if omitted.
name	Name of the new calendar	Yes

15.13 Empty calendar

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	calendar_emptycalendar	Yes
username	Username of the user that will own the calendar	Yes
domain	Domain of the user	No. Use default domain if omitted.
name	Name of the calendar to empty	Yes

WARNING: this command will delete all the events and tasks in a calendar.

15.14 Create file folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	file_createfolder	Yes
username	Username of the user that will own the folder	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	The full path of the folder, as it would appear in the web interface as page title (less the 'Files' name). Example: My documents/Personal	Yes

15.15 Create mail folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_createfolder	Yes
username	Username of the user that will own the folder	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	Complete path of the new folder. Subfolders are separated by the / character. Ex.: in/company	Yes

15.16 Delete old emails from mail folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_deleteemails	Yes
username	Username of the user that owns the folder	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	Complete path of the folder. Subfolders are separated by the / character. Ex.: in/company	Yes
before	Delete only messages received before the given date and time. The format is yyyyMMdd_HHmm, so for instance 20031225_1300 is 1PM of the 25 th December 2003	No. If this parameter is not present, all messages in the folder will be deleted.

Please note that this action cannot be undone. Messages are deleted immediately.

15.17 Create mail filter

Parameter name	Description	Required
----------------	-------------	----------

pwd	Password of the admin user	Yes
action	mail_createfilter	Yes
username	Username of the user that will own the filter	Yes
domain	Domain of the user	No. Use default domain if omitted.
name	Name of the new filter	Yes

Note: a filter in itself is ineffective. You need to add to it at least one match and one action (see below).

15.18 Delete mail filter

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_deletefilter	Yes
username	Username of the user that will own the filter	Yes
domain	Domain of the user	No. Use default domain if omitted.
name	Name of the filter to delete	Yes

15.19 Add address to a mailing list

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_addmailinglistaddress	Yes
domain	Domain of the list	No. Use default domain if omitted.
listname	Name of the mailing list	Yes
address	Email address to add	Yes

15.20 Remove address from a mailing list

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_removemailinglistaddress	Yes
domain	Domain of the list	No. Use default domain if omitted.
listname	Name of the mailing list	Yes
address	Email address to remove	Yes

15.21 Add action to a mail filter

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_addfilteraction	Yes

username	Username of the user that owns the filter	Yes
domain	Domain of the user	No. Use default domain if omitted.
filtername	Name of the existing filter	Yes
filteraction	One of the following: <ul style="list-style-type: none"> • delete • markread • copy • move • copytoboard • movetoboard • copytofiles • movetofiles • forward 	Yes
destination	Depends on the action. For instance if the action is 'forward', this may be 'joe@domain.com'. See the web interface for examples.	No if the action is 'delete' or 'markread'. Yes in every other case.

15.22 Add match to a mail filter

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_addfiltermatch	Yes
username	Username of the user that owns the filter	Yes
domain	Domain of the user	No. Use default domain if omitted.
filtername	Name of the existing filter	Yes
field	One of the following: <ul style="list-style-type: none"> • tocc • from • body • subject • header 	Yes
pattern	The text that is matched against the field. The filter will be applied if the field contains exactly the pattern (unless negate is TRUE, which reverses the logic).	Yes
negate	Whether the condition is negated. If TRUE, the filter will be applied if the field does <i>not</i> contain the pattern.	No. Default to FALSE.

15.23 Recalculate mail folder statistics

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_recalculatefolders	Yes

username	Username of the user whose mail folders will be affected by this operation	Yes
domain	Domain of the user	No. Use default domain if omitted.

This method recalculate the folder statistics (number of messages and number of unread messages) for all the mail folders of the user. These statistics are normally kept updated automatically, but can get out of sync in particular cases.

15.24 Validate email address

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_validatecpt	Yes
address	Email address to check	Yes

This method checks if the specified email address is a valid email address that maps to a local inbox (this includes user accounts, local aliases, global aliases, mailing lists). This method is useful for SMTP gateways that need to check whether incoming mail is directed to a legitimate address. If the given address is valid, the server will return a normal HTTP 200 status response. If the address is not valid, the server will return an HTTP response with the status set to 403 (Forbidden).

15.25 Import email message

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_importmail	Yes
username	Username of the user whose mail folders will be affected by this operation	Yes
domain	Domain of the user	No. Use default domain if omitted.
[FILE]	This must be a file uploaded using the encoding "multipart/form-data" standard. Note that all other parameters should not be encoded (ex if posting via an HTML form, put them in the form action target)	Yes. The file must be in rfc 822 format.
path	Complete path of the folder. Subfolders are separated by the / character. Ex.: in/company. Missing folders are automatically created if not existing.	Yes

15.26 Import mbox mail folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_importmbox	Yes

username	Username of the user whose mail folders will be affected by this operation	Yes
domain	Domain of the user	No. Use default domain if omitted.
filepath	The full filepath pointing to the mbox file. It must be located on the DeskNow server. Ex. /var/mail/user/Inbox	Yes. The file must be in mbox format.
path	Complete path of the folder. Subfolders are separated by the / character. Ex.: in/company. Missing folders are automatically created if not existing.	Yes

This method imports a mbox mail folder into the given folder for the given user. Typically used as building block for automatic import tools.

15.27 Import mail from a remote IMAP account

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_importimapaccount	Yes
username	Username of the user whose mail folders will be affected by this operation	Yes
domain	Domain of the user	No. Use default domain if omitted.
host	Internet address of the remote IMAP server from which to import mail. Ex imap.isp.com or 192.168.1.2	Yes
port	Internet port to use to connect to the remote server	No. 143 is used by default if omitted (993 if ssl is true)
ssl	Whether to use SSL to connect to the remote server. Use 'true' to enable SSL.	No. If this parameter is omitted, SSL is not used.
remoteusername	Username to use to login to the remote mail server for the account to import	Yes
remotepassword	Password to use to login to the remote server	Yes

This method imports all mail messages stored on a remote IMAP server into an account. Mail folders are automatically created in the local account when necessary. No existing messages will be deleted in the local account. No messages are removed/changed from the remote account. **NB** the mail import is a *scheduled* process. This means that this command will return immediately, and the actual import will happen in the background. This also means that if you stop DeskNow while the download is in progress, the import will start again after you restart DeskNow – this may cause message duplication. To

check the progress of the import, check the log.Admin file (and optionally the log.Debug log file).

Important: make sure that the disk quota of the account that you're importing into allows enough space for the mail that you're importing. If the account is new, it is enough to use the same disk quota as is used on the remote account.

15.28 Create external POP3 account

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_createexternalpop3account	Yes
username	Username of the user that will be linked to this account	Yes
domain	Domain of the user	No. Use default domain if omitted.
pop3host	The internet address of the pop3 server	Yes
pop3port	The TCp port of the pop3 service	No. Default is 110
pop3username	The username to use for the pop3 connection	Yes
pop3password	The password to use for the pop3 connection	Yes
ssl	TRUE if DeskNow should connect via SSL. Note that typically the port for ssl is 995, so you should specify it in the parameters.	No. Default is FALSE (no SSL).
automatic	TRUE if DeskNow should check this account automatically, at fixed times	No. Default is TRUE (check automatically)
leaveonserver	TRUE if DeskNow should leave a copy of the messages on the remote pop3 account	No. Default is TRUE (leave a copy)
onlogin	TRUE if DeskNow should check the account when the user logs in	No. Default is FALSE (do not check at login)
path	Complete path of the mail folder where messages will be downloaded. Subfolders are separated by the / character. Ex.: in/company. Missing folders are automatically created if not existing.	Yes

15.29 Create local mail alias

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_addlocalalias	Yes
domain	Domain where the alias will be created	Yes
alias	The alias to map to an existing address. Ex. 'sales'	Yes

target	An existing address. Ex. 'john'	Yes
--------	---------------------------------	-----

15.30 Create global mail alias

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_addglobalalias	Yes
alias	The alias to map to an existing address. See web interface for examples.	Yes
target	An existing address. See web interface for examples.	Yes

15.31 Post event to a calendar

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	calendar_createevent	Yes
username	Username of the user that owns the calendar	Yes
domain	Domain of the user	No. Use default domain if omitted.
calendarname	Name of the existing calendar	Yes
title		Yes
location		No
notes		No
fullday	'true' if this is a full-day event.	No. 'false' if omitted
start	The start time of the event. The format is yyyyMMdd_HHmm, so for instance 20031225_1300 is 1PM of the 25 th December 2003	Yes
stop	The stop time of the event. The format is yyyyMMdd_HHmm, so for instance 20031225_1300 is 1PM of the 25 th December 2003	Yes

15.32 Post task to a calendar

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	calendar_createtask	Yes
username	Username of the user that owns the calendar	Yes
domain	Domain of the user	No. Use default domain if omitted.
calendarname	Name of the existing calendar	Yes
title		Yes
location		No
notes		No
due	The due time of the event.	No. The task has no due

	The format is yyyyMMdd_HHmm, so for instance 20031225_1300 is 1PM of the 25 th December 2003	time if this parameter is omitted
--	---	-----------------------------------

15.33 Post to message board

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mboard_post	Yes
username	Username under whose credentials the message will be posted. The user must have write permission to the messageboard.	Yes
domain	Domain of the posting user	No. Use default domain if omitted.
messageboardname	Name of the messageboard, as seen from the user. If this is a "system" messageboard, the name must be the English version.	Yes
subject	Subject of the message. Use URL encoding where necessary.	Yes
body	Body of the message. Use URL encoding where necessary.	Yes

15.34 Create contact folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	contact_createfolder	Yes
username	Username of the user under which the contact folder will be created	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	The full path to the parent folder. Ex '/' to create the folder under the base node (Contacts), or '/subfolder1/subfolder2'	Yes.
name	Name of the new folder to create	Yes

15.35 Delete contact folder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	contact_deletefolder	Yes
username	Username of the user that owns the folder to be deleted	Yes
domain	Domain of the user	No. Use default domain if omitted.

path	The full path of the folder. Ex '/subfolder1/subfolder2' to delete the folder subfolder 2. Note that it is not possible to delete the root folder ('Contacts')	Yes.
------	---	------

15.36 Add contact

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	contact_addcontact	Yes
username	Username of the user under which the contact will be created	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	The full path to the parent folder. Ex '/' to add the contact under the base node (Contacts), or '/subfolder1/subfolder2'	Yes.
first	First name	No.
middle	Middle name	No.
last	Last name	No.
email	Email address	No.
address1	Home address	No.
address2	Home address	No.
city	Home city	No.
state	Home state	No.
postcode	Home postcode	No.
country	Home country	No.
phone	Home phone	No.
mobile	Mobile phone	No.
fax	Business fax	No.
company	Company/Organization	No.
notes	Notes	No.
wphone	Business phone	No.
email2	Second email address	No.
web	Web page url	No.
hfax	Home fax	No.
pager	Pager	No.
waddress1	Business address	No.
waddress2	Business address	No.
wcity	Business city	No.
wstate	Business state	No.
wpostcode	Business postcode	No.
wcountry	Business country	No.
jobtitle	Job title	No.
department	Department	No.
custom1	Custom field 1	No.
custom2	Cusomt field 2	No.
custom3	Custom field 3	No.
custom4	Custome field 4	No.

title	Title	No.
suffix	Suffix	No.
nickname	Nickname	No.
spouse	Spouse name	No.
vfb	URL of free/busy info	No.
office	Office	No.
manager	Manager name	No.
assistant	Assistant name	No.
profession	Profession	No.
birthday	Birthday	No.
anniversary	Wedding anniversary	No.
eventdelivery	Delivery mode of event invitations. 0=ical (Outlook, Notes, DeskNow) 1=text only 2=do not send invitations Default is 0.	No.
assistantphone	Assistant's phone number	No.
wpobox	Business PO Box	No.
hpobox	Home PO Box	No.

15.37 Delete contact

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	contact_deletecontact	Yes
username	Username of the user under which the contact will be deleted	Yes
domain	Domain of the user	No. Use default domain if omitted.
path	The full path to the parent folder. Ex '/' if the contact to delete is in the base node (Contacts), or '/subfolder1/subfolder2' if it is in subfolder2, etc.	Yes.
name	Name of the contact. Either the first name or last name of the contact must match this. It is recommended to use the last name where available.	Yes
emailaddress	Full email address of the contact, exactly as stored in the contact	At least one of emailaddress, phone is required. Use both if possible.
phone	A phone number (home, work) of the contact	At least one of emailaddress, phone is required. Use both if possible.

15.38 Send IM alert

Parameter name	Description	Required
pwd	Password of the admin user	Yes

action	im_sendalert	Yes
username	Username of the recipient user	Yes
domain	Domain of the recipient user	No. Use default domain if omitted.
alertcode	The alertcode can be passed to the JavaScript code in the applet page, to trigger events on the browser.	No
message	Body of the message. Use URL encoding where necessary.	Yes

Note: the alert will not be delivered if the user is not online.

15.39 Send IM broadcast

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	im_sendbroadcast	Yes
domain	Recipient domain. Omit to broadcast to all domains.	No. Broadcast to all domains if omitted.
message	Body of the message. Use URL encoding where necessary.	Yes

Note: the alert will be delivered only to users connected to the IM service.

15.40 Re-create user directory

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_recreateuserdirectory	Yes
domain		No. Default domain if omitted.

This method recreates the User Directory in the Administrator's Contacts.

15.41 Clear user's SyncML data

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_clearsyncdata	Yes
username		Yes
domain		No. Use default domain if not specified.

15.42 Recalculate user's disk usage

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_recalcdiskusage	Yes
username		Yes
domain		No. Use default domain if not specified.

15.43 Recalculate disk usage for all users of a domain

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_recalcdomaindiskusage	Yes
domain		No. Default domain if omitted.

Note: this operation may be slow.

15.44 Recalculate disk usage for all users in the system

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_recalldiskusage	Yes

Note: this operation may be slow.

15.45 Set user preference

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	user_setuserpreference	Yes
username		Yes
domain		No. Use default domain if not specified.
name	Name of the preference. Please consult the public forums or DeskNow's premium support for help.	Yes
value	Value to set for the preference	Yes

Note: this method lets the Administrator change programmatically user preferences. Note that you should consult with our tech support before attempting to use this method, to ensure that the names and values that you use are valid.

15.46 Set domain preference

Parameter name	Description	Required
----------------	-------------	----------

pwd	Password of the admin user	Yes
action	user_setdomainpreference	Yes
domain	The preference will be set for <i>all users of this domain</i> .	No. Use default domain if not specified.
name	Name of the preference. Please consult the public forums or DeskNow's premium support for help.	Yes
value	Value to set for the preference	Yes

Note: this method lets the Administrator change programmatically user preferences for *all the users of a given domain*. Note that you should consult with our tech support before attempting to use this method, to ensure that the names and values that you use are valid.

15.47 Enable autoresponder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_enableautoresponder	Yes
username		Yes
domain		No. Use default domain if not specified.
mode	Either 'always' or 'range' (enable only between two dates)	Yes.
begin	Day (included) at which the autoresponder will begin, in yyyyMMdd format	Yes if mode is 'range'.
end	Day (included) at which the autoresponder will end, in yyyyMMdd format	Yes if mode is 'range'.
subject	Autoresponse subject. Make sure to use URL encoding for special characters like space (%20).	No. Use preexisting if omitted
text	Autoresponse text. Make sure to use URL encoding for special characters like new line (%0d) and space (%20).	No. Use preexisting if omitted
onlyifsenttome	Send response only if email directly addressed to the user (not mailing list). TRUE or FALSE.	No. FALSE if omitted

15.48 Disable autoresponder

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_disableautoresponder	Yes
username		Yes
domain		No. Use default domain if not specified.

15.49 Reload I18N files

This command forces DeskNow to reload all the translation files for the various languages. See section 12 for more details.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	main_i18nreload	Yes

15.50 Clear the system's cache

This command completely clear's all of DeskNow internal cache.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	main_clearsystemcache	Yes

15.51 Emulate the expiration of the trial period

This command forces DeskNow to emulate the termination of the trial period in a free license. It is useful to test what happens when the trial period expires. Note that this may affect some settings and functionalities. To return to the trial period features, you need to restart DeskNow.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	main_el	Yes

15.52 View the content of the Bayesian dictionary

This command shows the content of the bayesian dictionary, displaying each token with the associated counters for ham and spam. It is intended for diagnostic purposes only.

Parameter name	Description	Required
pwd	Password of the admin user	Yes
action	mail_viewbayesdictionary	Yes

16 Licensing DeskNow

DeskNow is provided by default with the **DeskNow Lite** license.

This license is **FREE**, does not expire and can be used by an unlimited number of users.

Some of its advanced features however, are only available for a trial period of 30 days after its installation. At the end of this period, you can either choose to continue using DeskNow without these features, or to purchase a commercial license.

To purchase a commercial license or to see a full feature comparison list, please visit <http://www.desknow.com/desknowmc/buy.html> .

17 Support

DeskNow provides both open-style support using public discussion forums, and fast, premium support for customers that purchased a commercial license or Service Agreement.

Please visit <http://www.desknow.com/support.html> to access DeskNow Support.