Internet Services Delta Manual for HP-UX 11i Version 1.6

HP Part Number: 5969-4360
Published: E0402
Edition: First Edition

Legal Notices

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

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY

3000
Hanover Street

Palo Alto, California 94304 U.S.A.

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs, in their present form or with alterations, is expressly prohibited.

Copyright Notices

Trademark Notices

UNIX® is a registered trademark in the United States and other countries, licensed exclusively through The Open Group.

Contents

1 Sendmail 8.11.1 5
Chapter Overview 5
Sendmail 8.11.1 Features 5
Multiple Queue Directories 6
Enhanced Status Codes as defined in RFC 2034 6
DaemonPortOptions 7
ClientPortOptions 7
Spam Control Using MSA (RFC 2476) 8
Generating the Configuration File 8
SMTP Authentication based on RFC 2554 10
Virtual Hosting 10
LDAP-based Routing 12
New Option Value for QueueSortOrder - filename 13
New Option Value for Privacy Options - nobodyreturn 13
New or Enhanced Configuration Options 13
New Command Line Options 15
Finer spam control by using tags for the LHS of the access map 15
Changed Features 15
Compatibility with Previous Versions 16
Documentation 16
Known Problems and Workarounds 16
Limitations 17

2 WU-FTPD 2.6.1 18
Chapter Overview 18
WU-FTPD 2.6.1 Features 18
Virtual Hosts Support 18
privatepw 19
New Clauses for ftpaccess 19
Enabling RFC 1413 26
New Data Transfer Features 26
Field added to xferlog 26
New Command line Options 26
HP-specific Features 27
Command Line Options 27
Other Features 27
Compatibility with Previous Versions 28
Documentation 28

3 BIND 9.2.0 29
Chapter Overview 29
Summary of BIND 8.1.2 Features Supported on HP-UX 11i Version 1.6 29
Summary of BIND 9.1.3 Features Supported on HP-UX 11i Version 1.6 29
BIND 9.2.0 Features 30
Incremental Zone Transfer 30
DNSSEC 31
Dynamic DNS Update 32
TSIG-based Security 32
Lightweight Resolver Library and Daemon 32

Improved Logging Mechanism....33
Extended Configuration Syntax and Options....33
New Options in "options" Statement....33
New Options in "zone" Statement....35
New Option in "server" Statement....38
named-checkconf....38
named-checkzone....38
rndc....38
Generating rndc.conf File....40
New Command Line Options....40
Changed Features....42
HP-specific Options....42
Compatibility with Previous Versions of BIND....43
BIND 4.9.7 Compatibility....43
BIND 8.1.2 Compatibility....44
Documentation....44
Known Problems and Workarounds....45
Limitations....45

1 Sendmail 8.11.1

Availability of the new version of Sendmail 8.11.1 on various HP-UX platforms is summarized in the table below.

Table 1 Available Sendmail Versions

HU-UX	Software
11.00	Available as web upgrade.
11.i	Available as web upgrade
11i Version 1.6	Shipped with Sendmail 8.11.1

Chapter Overview

This chapter contains the following sections:

- Sendmail 8.11.1 Features
- Changed Features
- Compatibility with Previous Versions
- Documentation
- Known Problems and Workarounds
- Limitations

The above listed items are discussed in the following sections of this document for your reference.

Sendmail 8.11.1 Features

The features incorporated in Sendmail 8.11.1 on HP-UX 11i Version 1.6 are:

- Multiple Queue Directories
- Enhanced Status codes as defined in RFC 2034
- DaemonPortOptions
- ClientPortOptions
- Spam Control using Message Submission Agent (RFC 2476)
- Generating the Configuration File
- SMTP Authentication based on RFC 2554
- Virtual Hosting
- LDAP-based Routing
- New Option Value for QueueSortOrder
- New Option Value for Privacy Options
- New or Enhanced Configuration Options
- New Command Line Options
- Finer spam control by using tags for the LHS of the access map

The above features are discussed in the subsequent sections for your reference.

NOTE: `sendmail.cf` file is present in the directory "`usr/newconfig/etc/mail`". The user has to move `sendmail.cf` to the directory "etc/mail" before using any of the features listed below.

Multiple Queue Directories

This feature facilitates the parallel processing of mails by spreading process loads across multiple disks, thereby improving the queue performance, which is impacted by the number of entries in the queue directories. Unix files take a long time to open when number of entries in the directories exceed 1000.

In order to use multiple directories, the '`QueueDirectory`' option in the sendmail.cf file needs to be supplied with a value ending with *.

For example, in the configuration file, if you specify:

O QueueDirectory=/var/spool/mqueue/g*

All the directories or links to directories that begin with 'g' will be used. If there are five directories, g1, g2, g3, g4, and g5, Sendmail will use all the five directories when the Sendmail daemon is restarted. The mails are randomly assigned to the queue directories. The queue directory structure should not be changed when Sendmail is running.

Individual flushing of the mail queues can also be done by specifying the following on the command line:

sendmail -1 -o QueueDirectory=/var/spool/mqueue/g1

sendmail -q -O QueueDirectory=/var/spool/mqueue/g3

A new queue file naming system is also introduced in this release. The algorithm used to name files ensures that the names will be unique for 60 years. The queued items can be moved between queues with ease.

Enhanced Status Codes as defined in RFC 2034

This feature provides an official SMTP extension to deliver the Enhanced Mail System Codes for messages. These system codes have been derived from RFC 1893. In the earlier versions of Sendmail, messages during SMTP sessions were represented in 3-digit numeric codes like "550 Host Unknown", "220 Service Ready" etcetera.

The new system builds on existing 3-digit codes with three parts, each separated by a dot. For example, "2.1.1" or "5.1.2", where the first part is a single digit, length of second and third parts can be between 1 and 3.

The following are the different classes of return status:

a) 2.X.X

SuccessThe message was delivered.

b) 4.X.XPersistent Transient FailureSome temporary event (perhaps a full disk drive) could have caused the mail transfer to fail, but sending the same message in the future may be successful.

c) 5.X.XPermanent FailureThis message cannot be delivered. The headers of the message or format could be wrong.

RFC 1893 specifies an explicit set of values for the second and third digits (or groups of digits) of the returned status as well. "4.3.1" would mean the server temporarily rejected the message because its disks are full. "5.7.2" means the mail was bounced because it was sent to a list that the sender is not authorized to send mail to.

DaemonPortOptions

This option can be used to customize the daemon's SMTP service. The default value for the field 'Family' is 'inet' even if DaemonPortOptions is not defined or value for the 'Family' is not specified in the DaemonPortOptions setup.

By default, the `DaemonPortOptions` appears in the sendmail.cf file as:

O DaemonPortOptions=Name=MTA, Family=inet

O DaemonPortOptions=Port=587, Name=MSA, M=E

NOTE: For more information on MSA, read the "Spam Control using Message Submission Agent" section below.

The fields currently supported by Sendmail for DaemonPortOptions are:

- Family = inet
- Address = IP address or hostname
- Port = Port number/name
- Listen = Listen queue size
- M(Modifiers - Flags) = Table 1-2 lists and describes the various flags.

Table 2 Flags

Flag	Description
a	require authentication
b	bind to interface through which mail has been received
c	perform hostname canonification on the message
f	require fully qualified hostname
h	use the interface name for the outgoing `HELO` command
C	do not perform hostname canonification on the message
E	do not allow ETRN
u	disable fully qualified address for From:address

- SendBufSize = Send buffer size
- RcvBufSize = Receive buffer size
- Name = Name of the agent(MTA or MSA)

ClientPortOptions

This option is similar to DaemonPortOptions, but is meant only for outgoing connections.

The steps to set this option are same as those for the DaemonPortOptions except that the option name ClientPortOptions should be specified in lieu of DaemonPortOptions.

By default, it appears in the sendmail.cf file as:

```
#O ClientPortOptions=Address=0.0.0.0
```

Spam Control Using MSA (RFC 2476)

Message Submission Protocol is a means for MUAs to introduce new messages into the message transfer agent routing network. Messages being submitted by MUAs, in some cases, may be unfinished. Unfinished messages need to be completed by the MSA before submitting to the MTA. It also helps in implementing authenticated submission, including off-site submission by authorized users such as travellers.

The messages received on port 587 are regarded as "submitted messages". MSAs may implement message rejection rules, i.e. if an MSA is unable to determine a return path for the submitting user, from a valid MAIL FROM, a valid source IP address, or based on authenticated identity, then the MSA will immediately reject the message, as it gives the user and MUA direct feedback.

Sendmail 8.11.1 supports RFC 2476, a protocol for message submission. The anti-spam rulesets have been enhanced to improve the anti-spam capabilities. The RFC proposes a new standard for the Message Submission Agent (MSA). This is designed to replace the more general-purpose mail transfer agent (MTA) as the first service to which a Mail User Agent (MUA) connects to deliver a mail message. The RFC also describes how the usual protocols for SMTP service should be tightened up at the point where mail enters the system, rather than being routed from one site to another.Sendmail 8.11.1 also serves as a powerful tool to authenticate and control mail messages.

By default, MSA is defined in the sendmail.cf file as:

```
O DaemonPortOptions=Name=MSA, Port=587, M=E
```

where Port 587 is reserved for email message submission.

A Message Submission Agent still uses all of the same rulesets for processing the message (and therefore still allows message rejection via the check_* rulesets). In accordance with the RFC, the MSA will ensure that all domains in the envelope are fully qualified if the message is relayed to another MTA. It will also enforce the normal address syntax rules and log error messages. In addition to the above, you can request authentication before the messages are accepted by MSA by using the M=a modifier in the DaemonPortOptions.

NOTE: MSA can be turned off in the sendmail.cf file using the option, '`no_default_msa`' in gen_cf. For more information, refer to "`no_default_msa`" option below.

The `XUSR SMTP` command as well as the'-U' (initial user submission) command line option are deprecated. Mail user agents are expected to start using MSA for initial user message submission from now onwards. XUSR may be removed in future releases. The next release of sendmail will assume that any message submitted from the command line is an initial user submission and act accordingly.

Generating the Configuration File

gen_cf is a UNIX shell script, which is installed in the '/usr/newconfig/etc/mail/cf/cf' directory. This script cannot be copied to a different directory and executed, as it uses the macros defined in the `/usr/newconfig/etc/mail/cf` directory to generate the sendmail.cf file.

This script provides many options that will enable a specific ruleset. The input file for this script will be the *.m4 files defined in the `/usr/newconfig/etc/mail/cf` directory. The user can specify the output file and later incorporate site-specific changes (if any) in the output file.

NOTE: The output file generated by gen_cf, sendmail.cf.gen can be later copied or moved to `/etc/mail/sendmail.cf` file.

IMPORTANT: The entries in the sendmail.cf file preceded by a '#' (hash) are commented by default.

In addition to the options provided in Sendmail 8.9.3 release, the following new options have been added in the gen_cf script:

- `dnsbl`

  This new DNS-based black list option replaces '`rbl`', the RealtimeBlackhole List feature that was included in Sendmail 8.9.3 release. The rbl option is deprecated now. The `dnsbl` option avoids the possible confusion between RealtimeBlackhole List and other DNS-based Blacklist servers like ORBS. It takes the name of the Blacklist server and also an optional rejection message as arguments.

  `dnsbl` can be included multiple times in the sendmail.cf file, thereby allowing sites to subscribe to multiple Blacklist servers. The Blacklist server verifies the IP address of the incoming connection and rejects all the `SMTP` commands if the address is blacklisted. An error message is also displayed.

- `delay_checks`

  This option delays the anti-spam checks by Sendmail until it issues the `SMTP RCPT` command. Mails from certain addresses that might have been blocked by other anti-spam checks are received. In these cases, deferred checks are not done.

  By using `delay_checks`, the rulesets check_mail and check_relay will not be called when a client connects or issues a `MAIL` command, respectively. Instead, those rulesets will be called by the check_rcpt ruleset; they will be skipped if a sender has been authenticated using a "trusted" mechanism, i.e. one that is defined via the list of AuthMechanisms. If check_mail returns an error, the `RCPT TO` command will be rejected with that error. If it returns some other result starting with $#, then check_relay will be skipped. If the sender address (or a part of it) is listed in the access map and it has a RHS of OK or RELAY, then check_relay will be skipped.

- `relay_mail_from`

  This option can be used to facilitate relaying through a user's machine. The sender's name which is listed as "RELAY" in the access map (tagged with From:) can be specified using this option. The domain portion of the mail sender is also checked, when the optional argument 'domain' is provided.

- `ldap_routing`

  This option can be used to implement the ldap-based email recipient routing. This provides a method for re-routing addresses with a domain portion in class {LdapRoute} to either a different mail host or to a different address.

  NOTE: For more information, refer to "LDAP-based Routing" section.

- `no_default_msa`

  This option can be used to generate the configuration file without 'DaemonPortOptions' option for Message Submission Agent (MSA) daemon. The sendmail.cf configuration file will not contain the following line:

  ```
  O DaemonPortOptions=Port=587,
  Name=MSA, M=E
  ```

- `receive_only`

  This option generates a sendmail.cf file with a new set of rules called 'check_compat'. You can only receive mail messages, but cannot send them. Two new flags have been added in the `/etc/rc.config.d/mailservs` file. They are:

  1. `SENDMAIL_RECVONLY`

     This flag must be set to '1' in order to use "`receive_only`" feature.

  2. `SENDMAIL_SENDONLY`

     This flag may not be set to any value.

NOTE: Sendmail 8.11.1 depot will install the `mailservs` file in the `/usr/newconfig/etc/rc.config.d`. You need to manually move this file to `/etc/rc.config.d/` in order to use this feature.

The priorities for these flags are defined in the `/usr/newconfig/etc/rc.conig.d/mailservs` file.

- `send_only`

  This option generates a sendmail.cf file without the `'check_compat'` ruleset. You can only send mail messages, but cannot receive them.

  The `SENDMAIL_SENDONLY` flag in `/etc/rc.config.d/mailservs` file must be set to '1' in order to use 'send_only' feature.

SMTP Authentication based on RFC 2554

Sendmail 8.11.1 supports `SMTP AUTH` as defined in RFC 2554 (SMTP Service Extension for Authentication), which is based on SASL (Simple Authentication and Security Layer - RFC 2222). SMTP authentication provides a robust tool to control relaying with maximum flexibility.

The authentication protocol exchange consists of a series of server challenges (otherwise known as a ready response) and client answers that are specific to the authentication mechanism.

The `AUTH` parameter to the `MAIL FROM` command is set as:

```
MAIL FROM: from-addr AUTH=addr-spec
```

The addr-spec contains the identity that submitted the message to the delivery system. If the server trusts the authenticated identity of the client to assert that the message was originally submitted by the supplied addr-spec, then the server must supply the same addr-spec in an AUTH parameter when relaying the message to any server that supports the AUTH extension.

The list of authentication mechanisms for AUTH can be specified in the option, AuthMechanisms in the sendmail.cf file. By default, it appears in the sendmail.cf file as:

#O AuthMechanisms=GSSAPI KERBEROS_V4 DIGEST-MD5 CRAM-MD5

A new option to set AUTH parameter in `MAIL FROM` command has been added in the sendmail.cf file. By default, this appears as:

#O AuthOptions

If this option is set to 'A', the AUTH= parameter for the `MAIL FROM` command will be issued only when authentication succeeds.

DaemonPortOptions has one sub-option called 'modifiers' (M), one of which is 'a'. This instructs the daemon to necessitate authentication for all connections to it.

By default, it appears in the sendmail.cf file as:

#O DefaultAuthInfo=/etc/mail/default-auth-info

The `DefaultAuthInfo` option sets the filename which contains the authentication information for outgoing connections by default. It must contain the authorization id (userid), the authentication id (authid), the password (plain text), and the realm to use, each on a separate line. This information must be readable only by root (or the trusted user). If no realm is specified, $j will be used.

Virtual Hosting

Sendmail 8.11.1 imposes better control over virtusertable, which provides a domain-specific form of aliasing and also allows multiple domains to be hosted on a single machine.

With this feature, users can have their own domain names and receive mails using these domain names with a single host. You are required to obtain a new (available) domain name and set up name servers for that domain. Then, you must configure MX records for your new domain.

NOTE: You must know how to setup DNS before implementing this feature. For information on setting up DNS, refer to "Installing and Administering Internet Services" manual, posted on http://docs.hp.com.

The following steps describe how to set up virtual hosting:

- Assume 'mydomain.com' as the new domain name. If the mail server, which serves the new domain name has a full time connection to the internet, include the following line in the db.domain file.

  mydomain.com. IN MX 10 mymailserver.mydomain.com.

  Otherwise, you will need to have another machine to queue mails for your domain. Include the following lines in the db.domain file:

  mydomain.com. IN MX 10 mymailserver.mydomain.com.

  mydomain.com. IN MX 20 othermailserver.otherdomain.com.

  Now you must set up Sendmail.

  The generic-hpux10.mc file in `/usr/newconfig/etc/mail/cf/cf/generic...mc` is used for generating the configuration file. In the generic-hpux10.mc file, the version id string and the DOMAIN () flag can be modified to contain 'mydomain.com'.

- Create a file mydomain.com.m4 in `/usr/newconfig/etc/mail/cf/domain/` directory. This file must be similar to the `/usr/newconfig/etc/mail/cf/domain/generic.m4` file, with the version id containing 'mydomain.com'.
- Generate the `sendmail.cf.gen` file using `gen_cf` utility with '`virtusertable`' option and move this file to `/etc/mail/sendmail.cf`.

  NOTE: For more information on gen_cf, read the "Generating the Configuration File" section above.

- Create the virtusertable in `/etc/mail` directory. A sample virtusertable may look like:

  joe@mydomain.com jschmoe

  jane@mydomain.com jdoe@othercompany.com

  @mydomain.com jschmoe

  In the first example, the address joe@mydomain.com will be mapped to the local user jschmoe, jane@mydomain.com to the remote user jdoe@othercompany.com, and any other address in mydomain.com will also be mapped to jschmoe.

- Build the virtusertable database file in the command line as follows:

  ```
  # makemap dbm /etc/mail/virtusertable < /etc/mail/virtusertable
  ```

  If you wish to reverse-map local users for out-bound mail, you will need to generate sendmail.cf file with '`genericstable`' option in addition to '`virtusertable`' option.

  You must generate the genericstable similar to the virtusertable, but with the entries reversed.

  Example:

  jschmoe joe@yourdomain.com

- Add your domain name to `/etc/mail/sendmail.cw` file.
- Kill and Restart Sendmail.

  Now you should be able to receive mails at mydomain.com.

IMPORTANT: 'Virtual Hosting' feature provides better support for ISPs that offer queueing services to dial-up customers as queue-runs no longer wait for the dial-up server connection attempts to time out.

LDAP-based Routing

This feature can be used to implement the LDAP-based re-routing. This provides a method to re-route addresses with a domain portion in class {LDAPRoute} to either a different mail host or a different address. The domains can be added to the class {LDAPRoute} as given in the examples below.

Ensure that you set up a domain for LDAP routing. Assume that your domain is "yyy.com". Add the following line in the sendmail.cf file:

```
C{LDAPRoute}yyy.com
```

or

```
F{LDAPRoute}/etc/mail/ldap-domain-file
```

where `/etc/mail/ldap-domain-`file contains the domains.

The LDAPDefaultSpec option in the sendmail.cf file sets the default LDAP map specification. This needs to be set before defining LDAP maps. The settings will be used for all LDAP maps unless they are specified in the individual map specification ('K' command). By default, it appears in sendmail.cf file as follows:

```
O LDAPDefaultSpec=-h localhost
```

localhost can be replaced by your LDAP server name.

The following are the switches commonly used by most applications:

- '-b' - ldap search base

  "Directory" in ldap "tree" where the search begins. For example,

  ```
  -b "o=hp.com"
  ```

- '-d' - bindDN

  The BindDN parameter is used to specify the DN value for the LDAP bind request. For example,

  -d"cn=ldap://:389,dc=edat104,dc=atl,dc=hp,dc=com"

- '-h' - ldap servers

  Space separated string of servers which support ldap at your site. For example,

  ```
  -h "ldap1.hp.com
  ldap2.hp.com"
  ```

- '-p' - port numbers

  Port numbers where ldap service is available. For example,

  ```
  -p 33333
  ```

- '-k' - ldap search string (key)

  String that defines how a ldap map takes its input value and initiates an ldap search. For example,

  ```
  -k (&(ObjectClass=mailrecipient) (mail=%0))
  ```

- '-v' - ldap attribute

  Value that replaces the origin string in the map. In most cases, this will be the rfc822 email address. For example,

  ```
  -v mailroutingaddress
  ```

The ldap maps are defined in the configuration file as:

```
Kldap -1 -v mailHost -k (&(objectClass=inetLocalMailRecipient)
(mailLocalAddress=%0))
Kldapmra ldap -1 -v mailRoutingAddress -k (&(objectClass=inetLocalMailRecipient)
(mailLocalAddress=%0))
```

where

mailLocalAddress is the RFC 822 compliant email address of the recipient

mailHost is the fully-qualified host name of the MTA that is the final SMTP destination of the message to the recipient

mailRoutingAddress is the RFC 822 address to be used when routing messages to the SMTP MTA of the recipient.

New Option Value for QueueSortOrder - filename

This new option value can be used to sort the queue by opening each queue file to get the host and time information. The queue files need not be opened and read each time, when they are run. As a result, the queue processing becomes faster.

This option can be set in the sendmail.cf file as:

```
O QueueSortOrder=Filename
```

New Option Value for Privacy Options - nobodyreturn

This new option value instructs Sendmail to ignore the body of the original message, when notifying the delivery status of the message.

This option can be set in the sendmail.cf file as:

```
O PrivacyOptions=nobodyreturn
```

New or Enhanced Configuration Options

The following are the new or enhanced configuration options available in Sendmail 8.11.1:

- Timeout.*
  - The total time spent in satisfying a socket control request can be set using the 'Timeout.control' option. The default setting for this option is:

    ```
    #O Timeout.control=2m
    ```

  - The resolver's transmission time interval (in seconds) can be set using the 'Timeout.resolver.retrans' option. This option sets the 'Timeout.resolver.retrans.first', which sets the resolver's transmission time interval (in seconds) for the first attempt to deliver a message. It also sets the 'Timeout.resolver.retrans.normal' option. The default setting for this option is:

    ```
    #O Timeout.resolver.retrans=5s
    #O Timeout.resolver.retrans.first=5s
    #O Timeout.resolver.retrans.normal=5s
    ```

  - The frequency of resolver query retransmission can be set using the 'Timeout.resolver.retrans.normal' option. This option sets the 'Timeout.resolver.retry.first' option for the first attempt to deliver a message. It also sets the 'Timeout.resolver.retry.normal' option for all resolver lookups except for the first delivery attempt. The default setting for this option is:

    ```
    #O Timeout.resolver.retry=4
    #O Timeout.resolver.retry.first=4
    ```

```
#O Timeout.resolver.retry.normal=4
```

- DataFileBufferSize

  This option can be used to control the maximum size of a memory-buffered data (df) file before a disk-based file is used. The default setting for this option is:

  ```
  #O DataFileBufferSize=4096
  ```

- XscriptFileBufferSize

  This option can be used to control the maximum size of a memory-buffered (xf) transcript before a disk-based file is used. The default setting for this option is:

  ```
  #O XscriptFileBufferSize=4096
  ```

- MaxAliasRecursion

  The maximum depth of an alias recursion can be specified in the sendmail.cf file using this option. The default setting for this option is:

  ```
  #O MaxAliasRecursion=10
  ```

- PidFile

  The location of the ProcessId (Pid) file can be defined using this option. The default setting for this option is:

  ```
  #O PidFile=/etc/mail/sendmail.pid
  ```

  `/etc/mail/sendmail.pid` will be taken as the default file, if this option is not set. If you choose a directory other than `/etc/mail` for the pidfile, please ensure that the directory has proper write permissions as those of /etc/mail.

- ProcessTitlePrefix

  The prefix string for the process title shown in 'ps' listings can be specified using this option. By default, this option is commented. For example, if you set this option in the sendmail.cf file as:

  ```
  #O ProcessTitlePrefix=HPUX_Sendmail-8.11.1
  ```

  The command 'ps -ef | grep sendmail | grep -v grep' might show 'sendmail: HPUX_Sendmail-8.11.1: accepting connections' in the output.

- TrustedUser

  This option can be used to specify a user, who can own important files instead of root. This option necessitates 'fchown'. The default setting for this option is:

  ```
  #O
  TrustedUser=root
  ```

- MaxMimeHeaderLength

  The size of the MIME headers and parameters within those headers can be set using this option. This can also be used to protect Mail User Agents (MUA) from buffer overflow attacks. The default setting for this option is:

  ```
  #O MaxMimeHeaderLength=10
  ```

- DeadLetterDrop

  This option can be used to specify the location of the system-wide dead.letter file, which was formerly hardcoded to `/var/tmp/dead.letter`. The default setting for this option in this version is:

  ```
  O DeadLetterDrop=/var/tmp/dead.letter
  ```

  Please note that Sendmail will not save mails anywhere if this option is not set.

New Command Line Options

Table 1-3 lists and describes the new or enhanced command line options available in Sendmail 8.11.1:

Table 3 Command Line Options

Option	Description
`-G`	This option indicates that the message being submitted by the command line is meant only for relaying and not for gateway submission.
`-L`	This option can be used to set the identifier in syslog messages to a supplied tag.
`-C`	This option in "praliases" can be used to specify an alternate sendmail configuration file used for finding the alias file.
`-p`	This command can be used to print the output information in program-readable mode and reset the statistics file.

Finer spam control by using tags for the LHS of the access map

You can now tag entries in the access map based on their type. Three tags are available. They are:

1. Connect: connection information (${client_addr}, ${client_name})
2. From: sender
3. To: recipient

From:spammer@some.dom REJECT

To:friend.domain RELAY

Connect:friend.domain OK

Connect.from.domain RELAY

From:good@another.dom OK

From:another.dom REJECT

If the required item is looked up in a map, it will be tried with the corresponding tag in front, then without any tag (as fallback to enable backward compatibility). For example,

Changed Features

The following changes have been made in version 8.11.1 of Sendmail:

- The error code returned for unrecognized parameters to the SMTP mail and `RCPT` commands is changed from 501 to 555 as per RFC 1869.
- The configuration file (Sendmail.cf) version number is incremented to 9.
- Aliases with no right hand side are provided with 'missing value' warnings, when 'newaliases' is run instead of making an attempt to deliver the mail messages to an alias.
- A new mailer flag, 'F=%' is included in this release. Mailers, which have this flag will not attempt to deliver the message to the initial recipient. Those mails will be queued up. The queued messages are selected using one of the -ql/-qR/-qS queue run modifiers or an ETRN request and then delivered to the recipient.
- The [hostname] is added to class 'w' for the names of all interfaces unless DontProbeInterfaces option is set. This is useful for sending mails to hosts, which have dynamically assigned names.
- All numbered rulesets have been named in this release. They can still be accessed by their numbers. For example, Scanonify=3 instead of S3; SRecurse =97 instead of S97.

- A '/Quit' command to address the test mode is added. This command can be used to exit from the address test mode.
- The `SMTP` commands are not processed when the SMTP connection drops. This prevents a remote system from flooding the connection with commands and also disconnecting. In the earlier releases, all buffered commands were processed by the server.
- Purgestat and sendmail '-bH' options delete only expired files in the host status database, which have exceeded the values set by Timeout.hoststatus.

Compatibility with Previous Versions

Customers currently using any 8.x version of Sendmail do not need to modify their configuration file. It is compatible with this release of Sendmail. However, it is highly recommended to use the Sendmail 8.11.1 configuration file (.cf) version9 delivered with this release in order to effectively use the new features and changes incorporated in this version.

Documentation

The following manpages are distributed with this release of Sendmail:

- killsm.1m
- mailq.1
- mailstats.1
- makemap.1m
- praliases.1
- sendmail.1m

The following RFCs have been implemented in Sendmail 8.11.1:

- RFC 821
- RFC 822
- RFC 2821
- RFC 2033
- RFC 2034
- RFC 2222
- RFC 2476
- RFC 2487
- RFC 2505
- RFC 2553
- RFC 2554

Known Problems and Workarounds

The following are the known problems in this release of Sendmail:

- Sendmail uses identd, an optional authentication tool to find the user id for a given connection established with a remote machine. identd invokes some kernel services which hold the system resources for a long time. This affects the performance of Sendmail when there are large number of active TCP connections. The system appears hung during this timeframe.

  In order to resolve this problem, do the following:

  1. Disable identd by modifying the following entry in the sendmail.cf file:

     ```
     #O Timeout.ident=5s
     ```

as

```
O Timeout.ident=0s
```

Now you need to kill and restart Sendmail.

2. To disable identd, perform the following steps:
   a. Edit the `/etc/inetd.conf file` and comment out the ident line by placing a '#' in the first column as follows:

      ```
      #auth stream tcp wait bin /usr/lbin/identd identd
      ```

   b. Force inetd to re-read the inetd.conf file by executing '`/usr/sbin/inetd -c`' in the command line.

NOTE: identd is not distributed with this release of Sendmail. However if the system already contains an identd, you may encounter the above problem.

- If you specify a new location/file for the PidFile option in the sendmail.cf file and try killing Sendmail, the new file which does not contain any entries will be read by Sendmail instead of the default one. Therefore, Sendmail will not be killed. When you try restarting Sendmail, an error message is displayed.

  In order to resolve this problem, kill Sendmail before changing the PidFile option in the sendmail.cf file. You can then start Sendmail after making the changes in the sendmail.cf file. By doing this, the Pids will be written to the new file.

- Sendmail used to add other user's address in "Diagnostic-code" in warning messages under abnormal conditions, when the server connections are reset in the data phase. The email queue warning is returned to the sender with the Diagnostic-Code including an incorrect address.

  A workaround for this problem has been provided with this release. In such cases, no address will be added in the Diagnostic-Code line in the warning messages.

Limitations

The following are the limitations and fixes in Sendmail 8.11.1:

- identd, the tool used by Sendmail to authenticate and find the user id for a given connection established with a remote machine will be available in the future releases with appropriate defect fixes.

2 WU-FTPD 2.6.1

Availability of the new version of WU-FTPD 2.6.1 on various HP-UX platforms is summarized in the table below.

Table 4 Available WU-FTPD Versions

HU-UX	Software
11.00	Available as web upgrade
11.i	Available as web upgrade
11i Version 1.6	Shipped with WU-FTPD 2.6.1

Chapter Overview

This chapter contains the following sections:

- WU-FTPD 2.6.1 Features
- Compatibility with Previous Versions
- Documentation

The above listed items are discussed in the following sections of this document for your reference.

WU-FTPD 2.6.1 Features

The features incorporated in WU-FTPD on HP-UX 11i Version 1.6 are:

- Virtual hosts support
- privatepw
- New Clauses for ftpaccess
- Enabling RFC 1413
- New Features related to data transfer
- Field added to xferlog
- New Command line options
- HP-specific features
- Other Features

The above features are discussed in the subsequent sections for your reference.

Virtual Hosts Support

Virtual Hosts are now fully supported in WU-FTPD 2.6.1. A new configuration file named, "/etc/ftpd/ftpservers" has been added. This configuration file contains a set of virtual domain configuration file names that the WU-FTPD server can use. Thus, WU-FTPD has the ability to use separate configuration files for each virtual domain.

The new configuration file is placed in the directory "/etc/ftpd/". A sample configuration file can be found in directory "/usr/newconfig/etc/ftpd/".

WU-FTPD 2.4 partially supported virtual servers, that is it supported setting the root ftp directory, the log file and the banner for each virtual domain. All other directives in the ftpaccess file had to be shared globally across all virtual servers.

With WU-FTPD 2.6.1, the ftpaccess, ftpusers, ftpgroups, ftphosts and ftpconversions files can now be specified on a per domain basis. The Master WU-FTPD configuration files present under the

directory "/etc/ftpd" can now be overridden with a local copy specific to that domain. If you do not wish to place a copy of one or all the files listed above in the virtual host directory for any specific host, then the master copy can be used.

The following example illustrates a possible entry in the ftpservers configuration file:

```
123.123.123.123 /etc/ftpd/somedomain
```

In this example, when a ftp client connects to the server, using the IP Address 123.123.123.123, the WU-FTPD server searches for the configuration files ftpaccess, ftphosts, ftpusers, ftpgroups and ftpconversions under the directory "/etc/ftpd/somedomain". If a match is not found or an invalid directory path is encountered, the default master configuration files in directory "/etc/ftpd" are used instead.

privatepw

The admin utility "`privatepw`" allows modification of the WU-FTPD group access file information (`/etc/ftpd/ftpgroups`). A site currently supports the ftp commands SITE GROUP and SITE GPASS. When necessary, the administrator should be able to add, delete and list enhanced access group information needed for these two commands. The privatepw utility is used to update this information in the group access file (`/etc/ftpd/ftpgroups`). This command requires read/write permission for the appropriate ftpgroups file.

NOTE: For more details on the different options available in `privatepw`, refer to the `privatepw(8)` manpage.

New Clauses for ftpaccess

Several new ftpaccess clauses have been added to the file "/etc/ftpd/ftpaccess". The `/etc/ftpd/ftpaccess` file is used to configure the operation of ftpd. The new clauses have been listed below:

- Email-on-load:

  Using this feature, email addresses can be specified for anonymous upload notifications. The sender's email address can also be specified. By default, the sender's address is specified as 'wu-ftpd'. This can also be specified for virtual hosts. To avoid any problems if the recipient attempts to reply to a notification, or if downstream mail problems generate bounces, ensure that the mailfrom address is deliverable.

  The general syntax for this is:

  ```
  mailfrom <hostname>
  incmail <emailaddress>
  virtual <address> incmail <emailaddress>
  defaultserver incmail <email address>
  mailfrom <emailaddress>
  virtual <address> mailfrom <emailaddress>
  defaultserver incmail <emailaddress>
  deny-email <case-insensitive-email-address>
  ```

  NOTE: For more details on the email-on-load feature, refer to the `ftpaccess` (4) manpage.

  ```
  mailserver <abc.com>
  ```

  Specifying the name of a mail server that will accept upload notifications for the WU-FTP daemon. This option is relevant only if someone has to be notified of anonymous uploads.

  ```
  incmail <def@abc.com>
  ```

  Specifying the email addresses to be notified of anonymous uploads.

```
mailfrom <ghi@abc.com>
```

Specifying the sender's email address for anonymous upload notifications.

- timeouts:

  This feature is used to control the various timeouts used within the daemon. The following daemon timeout values are now available:

  1. accept - The time period the daemon waits for an incoming (PASV) data connection. The default value is 120 seconds.
  2. connect - The time period the daemon waits before attempting to establish an outgoing (PORT) data connection. The default value is 120 seconds. This affects the actual connection attempt. The daemon makes several attempts, sleeping a while between each, before completely giving up. During this 120 minutes time frame, the daemon keeps on trying to establish a connection. If the daemon fails to establish a connection during this period, it gives up.
  3. data - The time period the daemon will wait for some activity on the data connection. The default value is 1200 seconds.
  4. idle - The time period the daemon will wait for the next command. The default value is 900 seconds.
  5. RFC931 - The maximum time the daemon allows for the entire RFC931 (AUTH/ident) conversation. The default value is 10 seconds.
  6. maxidle - The SITE IDLE command allows the remote client to establish a higher value for the idle timeout. With a new option in ftpaccess (MaxIdle) this can be overridden. The default value is 1200 seconds.

  The general syntax for timeout is:

  ```
  timeout accept <seconds>
  timeout connect <seconds>
  timeout data <seconds>
  timeout idle <seconds>
  timeout maxidle <seconds>
  timeout RFC931 <seconds>
  ```

  ### Example 1

  ```
  timeout idle 200
  ```

  Displays the following message "Current IDLE time limit is 200 seconds; max 7200"

  ```
  timeout maxidle 6200
  ```

  Displays the following message "Current IDLE time limit is 200 seconds; max 6200"

  ```
  timeout RFC931 0
  ```

  Disables RFC931 based authentication since 0 has been specified.

- Enhanced DNS extensions:

  This feature is used for refusing (or overriding) an FTP session when a reverse DNS lookup fails.

  The general syntax for this is:

  ```
  dns refuse_mismatch <filename> [ override ]
  dns refuse_no_reverse <filename> [ override ]
  dns resolveroptions <options>
  ```

- Control of the address reported:

  This feature allows control of the address reported in response to a `PASV` command and the TCP port numbers, which may be used for a passive data connection.

  The general syntax for this is:

  ```
  passive address <externalip> <cidr>
  passive ports <cidr> <min> <max>
  ```

  Example 2

  ```
  passive address 10.0.1.15
    10.0.0.0/8
  ```

  In this example, clients connecting from the class-A network 10 will be notified that the passive connection is listening on the IP-address 10.0.1.15

  ```
  passive ports 10.0.0.0/8
  90 100
  ```

  In this example, if there is a control connection from the class-A network 10, the port range within 90 and 100 will be randomly selected for the daemon to listen on.

- Selectively allow PORT and PASV data connections:

  This feature enables the site admin to selectively allow PORT and PASV data connections. Usually a connection is not established if the remote IP address of the data connection does not match the remote IP address of the control connection data. Multiple passive addresses may be specified to handle complex, or multi-gatewayed, networks.

  The general syntax for this is:

  ```
  pasv-allow <class> [ addrglob ...]
  port-allow <class> [ addrglob ...]
  ```

- SO_KEEPALIVE:

  This feature sets the TCP option SO_KEEPALIVE for data sockets. This can be used to control network disconnect. You could specify "Yes" to set this option or "No" to use the system default, which is usually off.

  The general syntax for this is:

  ```
  keepalive yes|no
  ```

- ftpaccess log:

  The feature ftpaccess log clause has been changed to allow logging transfers to both the syslog and xferlog. This option enables you to redirect the logging messages for incoming and outgoing transfers to syslog. If this option has not been specified, the messages are written to xferlog.

  The general syntax for this is:

  ```
  log syslog
  log syslog+xferlog
  ```

- Clauses to control access to areas on the FTP site:

  The following clauses control whether a real or guest user is allowed access to areas on the FTP site outside their home directories. These clauses are not meant to replace the use of guestgroup and guestuser. Instead, you can use these clauses to supplement the operation of guests. The unrestricted-uid and unrestricted-gid clauses may be used to allow users to use their home directories who would otherwise be restricted.

  The general syntax for this is:

```
restricted-uid <uid-range>[...]
restricted-gid <gid-range>[...]
unrestricted-uid <uid-range>[...]
unrestricted-gid <gid-range>[...]
```

Example 3

```
restricted-uid abtalt abtuser
restricted-gid users abt
```

- Retrieval of files:

  This feature allows retrieval of files which would otherwise be denied by the 'noretrieve' clause. This clause overrides the noretrieve clause.

  The general syntax for this is:

  ```
   allow-retrieve [ absolute|relative ]  [ class=
  classname ] ...  [-] filename
  ```

- Virtual Server:

  New virtual server clauses have been added. Using these clauses, the access for different users to both the virtual and non-virtual domain can be restricted. Also, the virtual hostname can be specified for printing by using one of the options below.

  The general syntax for this is:

  ```
  virtual <address> allow <username> [ username ...]
  virtual <address> deny <username> [ username ...]
  virtual <address> private
  virtual <address> hostname|email string
  defaultserver deny <username> [ username ...]
  defaultserver allow <username> [ username ...]
  defaultserver private
  ```

Example 4

```
 virtual xx.xx.xx.xx allow
root
```

- Adding this entry will ensure that user root is allowed to start the ftp session in the machine. By default, all real and guest users will be denied service. This is applicable only for virtual ftpservers.

```
virtual xx.xx.xx.xx allow *
```

```
virtual xx.xx.xx.xx deny root
```

- Adding this entry will deny root users and allow other users to start ftp.

```
virtual xx.xx.xx.xx private
```

- Adding this entry will deny service for anonymous ftp.

```
virtual xx.xx.xx.xx hostname
telnet2.abc
```

- Adding this entry will print the greeting (telnet2.abc) in place of the actual hostname at the beginning.

```
defaultserver deny root
```

- Adding this entry denies ftp on the default ftp server for the root user. The message "FTP LOGIN REFUSED" is displayed in syslog.

```
defaultserver private
```

- Adding this entry anonymous ftp is denied on default server. The message "FTP LOGIN REFUSED" is logged in the syslog.

- Default host name:

  This feature defines the default host name of the ftp server which will be displayed in the greeting message. If this clause is not specified, the default host name of the local machine is used.

  The general syntax for this is:

  ```
  hostname <some.host.name>
  ```

Example 5

```
hostname telnet2.123.com
```

- Displays the default hostname specified (telnet2.123.com) in place of the actual hostname in the greeting message.

- Control information:

  This feature allows you to control the information given out in the greeting message before a remote user logs in. The information specified can be either hostname and daemon version, only hostname or just the message "FTP server ready". By default, "greeting full" is set as the default greeting clause.

  The general syntax for this is:

  ```
  greeting full|brief|terse
  ```

  ```
  greeting text <message>
  ```

  - Using the "greeting text" clause a different text message from the standard greeting can be printed.

Example 6

```
greeting text Hi!!! Welcome
to FTP Server
```

- Displays the message "Hi!!! Welcome to FTP server" as the greeting message.

- Limit the total time of a session:

  This feature allows you to limit the total time a session can take. By default, there is no limit set. Real users are never limited.

  The general syntax for this is:

  ```
  limit-time {*|anonymous|guest} <minutes>
  ```

- Forcing all UID/GID's:

  This feature has the ability to force all UID/GID in a range to be treated as guests. This is a new feature that has been added.

  The general syntax for this is:

  ```
  guestuser <username> [ username ... ]
  realgroup <groupname> [ groupname ... ]
  realuser <username> [ username ... ]
  ```

- Specification of UID and GID values:

  These clauses allow specification of UID and GID values, which will be denied access to the ftp server. By default, allow access is set.

  The general syntax for this is:

  deny-uid <uid-range>[...]

  ```
  deny-gid <gid-range>[...]
  allow-uid <uid-range>[...]
  allow-gid <gid-range>[...]
  ```

- access clauses:

  Detail counters and ftpaccess clauses allow to set limitation of the user's ability to upload/download files.

  The general syntax for this is:

  ```
  ul-dl-rate <rate> [ class ...]
  dl-free <filename> [ class ...]
  dl-free-dir <dirname> [ class ...]
  ```

Example 7

```
ul-dl-rate 2
```

- For every 1 byte of data that is uploaded, the ftpserver will allow 2 bytes of data to be downloaded.

- nice clause:

  The 'nice' clause allows modification of the nice value of the WU-FTPD server process for certain users. The process nice value is adjusted by the indicated nice-delta value, if the remote user is a member of the named class. If class is not specified, then use nice-delta as the default adjustment to the WU-FTPD server process nice value. This default nice value adjustment is

used to adjust the nice value of the server process only for those users who do not belong to any class for which a class-specific nice directive exists in the ftpaccess file.

The general syntax for this is:

```
nice <nice-delta> [ class ]
```

NOTE: Only negative values are considered. Positive values or 0 are ignored for the nice-delta.

- defumask clause:

  The 'defumask' clause allows to set the umask for a file created by the daemon, if the remote user is a member of the named class. There can be multiple entries for defumask. For classes which do not have a defumask entry, the system unmask will be used as the default.

  The general syntax for this is:

  ```
  defumask <umask> [ class ]
  ```

  Example 8

  ```
  defumask 0177
  defumask 0133 ClassA
  ```

  - Specifying this will create files with the permission "-rw-r--r--" for a user of ClassA. For other users files will be created with the permission "-rw------".

- Controlling the maximum number of lines of output:

  A clause to set the maximum number of lines of output to be displayed on the screen has been added. By default, the limit is set to 20.

  The general syntax for this is:

  ```
  site-exec-max-lines <number> [ class ...]
  ```

- Setting the root directory:

  The clauses to set the root directory when the user logs in as anonymous or guest have been added.

  The general syntax for this is:

  ```
  anonymous-root <root-dir> [ class ]
  guest-root <root-dir> [ uid-range ]
  ```

- Enabling the server to listen:

  A clause has been added to enable the server to listen on any particular address. If this value is not set, then the server will listen for connections on every IP address. Use of this clause is discouraged as it will break virtual hosting.

  NOTE: This option will work only when WU-FTPD is running in a standalone mode. For more details, refer to `ftpd(1m)` manpage.

  The general syntax for this is:

  ```
  daemonaddress <address>
  ```

NOTE: For more details on the new clauses added to the ftpaccess utility, refer to the `ftpaccess(4)` manpage.

Enabling RFC 1413

The Identification Protocol (RFC 1413) provides a means to determine the identity of a user of a particular TCP connection. Given a TCP port number pair, it returns a character string which identifies the owner of that connection on the server's system. Use the "-I" daemon option to enable RFC 1413 based authentication. By default, this authentication is disabled.

New Data Transfer Features

Two new features related to data transfer have been introduced:

- For statistical purposes, you can keep track of the total bytes of data transferred. Also, you can limit the number of data bytes a user in any given class may transfer. A limit can be placed on the number of bytes in, out or total. This clause can be specified for a particular class. If no class is specified for a data limit entry, that limit entry is the default for all classes for which this clause is not specified. When logging off the ftp session, it will print the number of files and the number of bytes transferred.

  The general syntax for this is:

  ```
  data-limit [raw] in|out|total count [class]
  ```

- Limit the number of data files a user in the given class may transfer in a session. The limit can be placed on the files in, out or total. If no class is specified, the limit is the default for classes, which do not have a limit specified.

  The general syntax for this is:

  ```
  file-limit [raw] in|out|total count [class]
  ```

Field added to xferlog

A new field has been added in `xferlog` to indicate the completion status of the data transfer. "C" indicates complete transfer and "i" indicates incomplete transfer.

NOTE: For more details on the field that has been added to `xferlog`, refer to the `xferlog(5)` manpage.

New Command line Options

Table 2-1 lists and describes the new or enhanced command line options available in WU-FTPD 2.6.1:

Table 5 Command Line Options

Option	Description
`-q` & `-Q`	These options determine whether the WU-FTPD daemon uses the PID files.
`-r root dir`	This option instructs the daemon to chroot (see `chroot(2)`) to the specified rootdir immediately upon loading.
`-V`	This option causes the program to display copyright and version information, then terminate.
`-w` & `-W`	These options determine whether user logins are to be recorded in the wtmp file.
`-X`	If the -X option is specified, the output created by the -i and -o options is not saved to the xferlog file but written to syslog.

Table 5 Command Line Options *(continued)*

Option	Description
`-I`	This option enables the use of RFC1413 (AUTH/ident) to attempt to determine the username on the client.
`-s` & `-S`	These options run the daemon in standalone operation mode. The -S option runs the daemon in the background and is useful in start-up scripts during system initialization (i.e., in rc.local). The -s option leaves the daemon in foreground and is useful when running from init (see init(1M)).
`-c <ctrl port>` & `-C <data port>`	These options override the control and the data port numbers that is used by the daemon. These options override the control and the data port numbers that is used by the daemon.
`-U`	For 11.0By setting this option the user can switch from "sendfile()" system call to "send()" system call. By default, "sendfile()" is used. For 11i The option "sendfiletransfer" in the ftpaccess configuration file has been replaced with '-U' option in WU-FTPD. By setting this option the user can switch from "sendfile()" system call to "send()" system call. By default, "sendfile()" is used.
Other Utilities	In version 2.6.1, a new option "-V" prints the copyright and the version information information for all utilities (ftpcount, ftprestart, ckconfig, ftpwho, privatepw and ftpshut).

HP-specific Features

The following features have been incorporated in HP's port of WU-FTPD 2.6.1:

Command Line Options

Table 2-2 lists and describes the new or enhance command line options available in Sendmail 8.11.1: The options discussed below are present in WU-FTPD 2.4. These options are not present in the open-sourced version of WU-FTPD 2.6.1, but have been incorporated in HP's port for backward compatibility:

Table 6 Command Line Options

Option	Description
`-m` number_of_tries	Specifies the number of tries for a bind() socket call.
`-n` nice_value	This option can be used to set the identifier in syslog messages to a supplied tag.
`-B`	Sets the buffer size of the data socket to blocks of size of 1024 bytes. The valid range for size is from 1 to 64 (default is 56).
`-p` & `-P`	The -p option is used to allow private port access to the client. The -P option is used to allow Third Party Access as well as private port access.

Other Features

In addition to the above, WU-FTPD 2.6.1 on HP-UX also supports:

- Files greater than 2 GB;
- Large UIDs/GIDs;
- Trusted System features.

Compatibility with Previous Versions

Customers currently using WU-FTPD 2.4 do not need to modify their configuration file. It is compatible with this release of WU-FTPD. However, it is highly recommended to use the WU-FTPD 2.6.1 configuration file (.cf) delivered with this release in order to effectively use the new features and changes incorporated in this version.

Documentation

The README files for WU-FTPD are available in /usr/share/doc. You may also need to read WU-FTPD 2.4 Release Notes, posted on http://docs.hp.com for more information about the product.

The following are the manpages distributed with the FTP depot:

- ftp.1
- ftpd.1m
- ckconfig.1
- ftprestart.1
- ftpwho.1
- ftpcount.1
- ftpshut.1
- privatepw.a
- ftpaccess.4
- ftpgroups.4
- ftpservers.4
- ftpconversions.4
- ftpusers.4
- ftphosts.4
- xferlog.5

The following RFCs have been implemented in WU-FTPD 2.6.1:

RFC 959
RFC 1639
RFC 2428

3 BIND 9.2.0

Availability of the new version of BIND 9.2.0 on various HP-UX platforms is summarized in the table below.

Table 7 Available BIND versions

HU-UX	Software
11.00	Available as web upgrade.
11.i	Available as web upgrade
11i Version 1.6	Shipped with BIND 9.2.0

Chapter Overview

This chapter contains the following sections:

- Summary of BIND 8.1.2 features supported on HP-UX 11i Version 1.6
- Summary of BIND 9.1.3 features supported on HP-UX 11i Version 1.6
- BIND 9.2.0 Features
- Changed Features
- Compatibility with Previous Versions
- Documentation
- Known Problems and Workarounds
- Limitations

The above listed items are discussed in the subsequent sections of this document for your reference.

Summary of BIND 8.1.2 Features Supported on HP-UX 11i Version 1.6

This section lists the BIND 8.1.2 features that are supported on HP-UX 11i Version 1.6 platform:

- DNS Change Notification (DNS Notify) (RFC 1996)
- Support for Dynamic DNS Update
- Improved Logging System
- More Efficient Zone Transfers
- New Configuration Syntax in `/etc/named.conf`

NOTE: For information on the above features, refer to the BIND 8.1.2 Release Notes available at: http://www.docs.hp.com/hpux/netcom/index.html#Internet%20Services

Summary of BIND 9.1.3 Features Supported on HP-UX 11i Version 1.6

This section lists the BIND 9.1.3 features that are supported in BIND 9.2.0.

- Incremental Zone Transfer(RFC 1995)
- DNS Security (DNSSEC)
- Dynamic DNS Update
- TSIG-based Transaction Security
- Lightweight Resolver Library and Daemon

- Extended Configuration Syntax and Options
- Improved Logging Mechanism

NOTE: For information on the above features, refer to the BIND 9.1.3 Release Notes available at: http://www.docs.hp.com/hpux/netcom/index.html#Internet%20Services

BIND 9.2.0 Features

The features incorporated in BIND 9.2.0 on HP-UX 11i Version 1.6 are:

- Incremental Zone Transfer
- DNSSEC
- Dynamic DNS Update
- TSIG-based Security
- Lightweight Resolver Library and Daemon
- Improved Logging Mechanism
- Extended Configuration Syntax and Options
- New Options in "options" Statement
- `A` utility, named-checkconf to check the syntax of `named.conf` file
- A utility, `named-checkzone` to check the syntax and consistency of a zone's contents
- A utility, `rndc` to control the operation of a name server
- A utility rndc-confgen to generate the configuration file for rndc
- New command line options
- New options in "server" statement
- New options in "zone" statement

The above features are discussed in the following subsections for your reference.

Incremental Zone Transfer

The incremental zone transfer protocol is a mechanism for slave servers to transfer only the changed data, instead of transferring the entire zone every time the zone data changes. This is defined in RFC 1995.

When acting as a master, BIND 9.2.0 supports IXFR for those zones where the necessary change history information is available. These include master zones maintained by dynamic updates and slave zones whose data was obtained by IXFR. It does not support IXFR for manually maintained master zones and slave zones obtained by zone transfer (AXFR).

When acting as a slave, BIND 9.2.0 will attempt to use IXFR unless it is explicitly disabled.

The option statements used to disable IXFR are:

```
[request-ixfr yes_or_no;]
[provide-ixfr yes_or_no;]
```

These options are set manually by the administrator in the `/etc/named.conf` configuration file. These options can be set to 'yes' or 'no' or can be excluded from the `/etc/named.conf` configuration file.

The `provide-ixfr` clause determines whether the local server, acting as a master, will respond with incremental zone transfer when the given remote server, a slave, requests it.

If set to yes, incremental transfer will be provided whenever possible. If set to no, all transfers to the remote server will be non-incremental. If not set, the value of the `provide-ixfr` option in the global options block is used as the default.

The `request-ixfr` clause determines whether the local server acting as a slave will request incremental zone transfers from the given remote server, a master.

If set to yes, incremental transfer will be requested from the given remote server (master). If set to no, all transfers to the remote server will be non-incremental. If not set, the value of the `request-ixfr` option in the global options block is used as the default.

DNSSEC

Authentication of DNS information in a zone is possible through the DNS Security (DNSSEC) extensions defined in RFC 2535. In order to set up a DNSSEC secure zone, there are a series of steps, which need to be followed (explained below). BIND 9.2.0 ships with several tools that are used for this process.

There must be communication with administrators of the parent and/or child zone to transmit keys and signatures. The parent zone for a DNSSEC-capable resolver to trust its data must indicate a zone's security status. For other servers to trust data in this zone, they must either be statically configured with this zone's zone key or the zone key of another zone above this on in the DNS tree.

Validation for wild card records in secure zones is not fully supported. In particular, `"a name does not exist"` response will validate successfully even if it does not contain the NXT records to prove the existence of a matching wild card.

Generating Keys

The `/usr/bin/dnssec-keygen` program is used to generate keys.

A sample directive to invoke the dnssec-keygen program to generate a 768-bit DSA key for the domain example.com, is as shown below. The `"-a"` option is used to specify the encryption algorithm. The `"-b'` option is used to specify the key size and the "-n" option is used to specify the nametype which can be a ZONE, HOST, ENTITY or USER.

NOTE: Refer to the `dnssec-keygen(1)` man page for a detailed description of all supported functions.

```
# /usr/bin/dnssec-keygen -a DSA -b 768 -n ZONE example.com
```

The above command will generate the key identification string "Kexample.com.+003+26160" indicating a DSA key with identifier 26160.

Creating a Keyset

The `/usr/bin/dnssec-makekeyset` program is used to create a keyset from one or more keys.

A sample directive to invoke the `dnssec-makekeyset` for the key "Kexample.com.+003+26160" (generated by the `dnssec-keygen` program) is as shown below.

The option `"-t"` is used to specify the TTL value that will be assigned to the assembled KEY and SIG records in the output file. The options `"-s"` and `"-e"` are used to indicate the start-time and end-time or the expiry date for the SIG records respectively.

NOTE: Refer to the `dnssec-makekeyset (1)` man page for a detailed description of all supported options.

```
# /usr/bin/dnssec-makekeyset -t 86400 -s 20007011200000 -e +2592000
Kexample.com+003+26160
```

The output of this command is a file named example.com.keyset containing a SIG and KEY record for the ZONE example.com.

Signing the Child's Keyset

The `/usr/bin/dnssec-signkey` program is used to sign a keyset for a child zone.

```
# /usr/bin/dnssec-signkey example.com.keyset Kcom.+003+51944
```

The output of the above command is a file named `example.com.signedkey` which has the keys for example.com signed by the com zone's zone key.

Signing the Zone

The `/usr/bin/dnssec-signzone` program is used to sign a zone.

A sample directive to invoke the dnssec-signzone to sign the zone, example.com is as shown below.

Kexample.com.+003+26160 is the key identifier generated by the dnssec-keygen program.

```
# /usr/bin/dnssec-signzone example.com Kexample.com.+003+26160
```

dnssec-signzone will create a file named example.com.signed, the signed version of the example.com zone. This file can then be referenced in a zone statement{} in `/etc/named.conf` so that it can be loaded by the nameserver.

Configuring Servers

Unlike in BIND 8.1.2, data is not verified on load in BIND 9.2.0. Hence zone keys for authoritative zones do not need to be specified in the configuration file. The public key for any security root must be there in the configuration file's `trusted-keys` statement.

Dynamic DNS Update

Dynamic update is the ability to add, modify or delete records or RR sets in the master zone files under a specified zone. Dynamic update is based on RFC 2136. Dynamic update is enabled on a zone-by-zone basis, by including an allow-update or update-policy clause in the `zone` statement of the `/etc/named.conf` file.

NOTE: Zone files of dynamic zones must not be edited manually, as those changes could cause conflict with dynamic updates. Use the `nsupdate` utility to submit dynamic DNS update requests to a name server.

TSIG-based Security

To secure server-to-server communication, BIND 9.2.0 primarily uses TSIG. This includes zone transfer, notify, and recursive query messages. TSIG is most useful for dynamic updates. To secure dynamic updates to a primary server of a dynamic zone, key-based access control is more effective than IP-based access control. The `nsupdate` program with the `"-k"` and `"-y"` options is used to provide the shared secret needed to generate the TSIG record for authenticating dynamic DNS update request.

NOTE: Refer to the `nsupdate(1)` man page for more details.

Lightweight Resolver Library and Daemon

The applications that require address-to-name lookups have been linked with a stub resolver library that sends recursive DNS queries to a local caching name server.

BIND 9.2.0 provides resolution services to local clients using a combination of a lightweight resolver library and a resolver daemon process running on the local host. These communicate using a simple UDP-based protocol "lightweight resolver protocol", that is distinct from and simpler than the full DNS protocol.

To use the lightweight resolver interface, the system must run the resolver daemon `lwresd`. The daemon currently looks only in the DNS, but in the future it may use other sources such as `/etc/hosts`, `NIS`, etcetera.

NOTE: Refer to the `lwresd(1m)` man page for more information.

Improved Logging Mechanism

In BIND 9.2.0, the logging mechanism is established only when the entire configuration file has been parsed. In BIND 8.1.2, it was established as soon as the logging statement was parsed. When the server is starting up, all logging messages regarding syntax errors in the configuration file go to the default channels or to standard error if the "`-g`" option was specified. The log files are no longer dumped in the `/var/tmp` directory, they are put in the local directory.

Extended Configuration Syntax and Options

BIND 9.2.0 configuration is broadly similar to BIND 8.1.2, however, there are a few new areas of configuration such as views.

BIND 8.1.2 configuration files should work with few alterations in BIND 9.2.0, although the more complex configurations should be reviewed to check that they are more efficiently implemented using the new features found in BIND 9.2.0.

BIND 4.9.7 configuration files need to be converted to the new format using the shell script `/usr/bin/named-bootconf.sh`.

New Options in "options" Statement

The following lists and describes the new options added in "`options`" statement:

- `allow-recursion`

  This option specifies which hosts are allowed to make recursive queries through the server. If not specified, recursive queries from all hosts are allowed.

- `max-transfer-idle-in`

  This option is used to terminate inbound zone transfers making no progress in the specified period. Default is 60 mins.

- `max-transfer-time-out`

  This option is used to terminate outbound zone transfers running longer than the specified time. Default is 120 mins.

- `max-transfer-idle-out`

  This option is used to terminate out bound zone transfers making no progress in the specified period. Default is 60 mins.

- `max-cache-ttl`

  This option is used to specify maximum time to cache positive answers. Default is 7 days.

- `max-ncache-ttl`

  This option is used to specify maximum time to cache negative answers. Default is 3 hrs. This value should not exceed 7 days and will be truncated to 7 days if a longer time period is specified.

- `transfer-source`

  This option specifies the IPv4 address to use for inbound zone updates, which is also the source address to use for refresh queries and forwarded dynamic updates. If not set, it defaults to a system-controlled value which will usually be the address of the interface "closest to" the remote end.

- `request-ixfr`

  This option is used to determine whether the local server, acting as a master, will respond with an incremental zone transfer when the given remote server, a slave, requests it. If set to yes, incremental transfer will be provided whenever possible. If set to no, all transfers to the remote server will be non-incremental. If not set, the value of the `provide-ixfr` option in the global options block is used as default.

- `provide-ixfr`

  This option is used to determine whether the local server, acting as a slave will request incremental zone transfers from the given remote server, a master. If not set, the value of the request-ixfr option in the global options block is used as default. This option is used to allow incremental zone transfers for the requests to the master server.

- `recursive-clients`

  This option is used to specify the maximum number of recursive lookups the servers will perform on behalf of clients. Default is 1000.

- `tcp-clients`

  This option is used to specify the maximum number of TCP connections a server will accept. Default is 100.

- `tkey-domain`

  This option is used to specify the domain name that is appended to the shared keys generated by TKEY. When a client requests a TKEY exchange, it may or may not specify the desired name for the key. If present, the name of the shared key will be "client specified part" + "tkey-domain". Otherwise, the name of the shared key will be "random hex digits" + "tkey-domain". In most cases, the domain name must be the server's domain name.

- `tkey-dhkey`

  This option is used to specify the Diffie-Hellman key used by the server to generate shared keys for clients using the Diffie-Hellman mode of TKEY. The server must be able to load the public and private keys from files in the working directory. In most cases, the keyname should be the server's host name.

- `port`

  This option is used to specify the port number to be used.

- `sig-validity-interval`

  This option is used to specify the expiry time of DNSSEC signature that is automatically generated as a result of dynamic updates. Default is 30 days.

- `dump-file`

  This option is used to specify the pathname of the file to which the server dumps the database with the `rndc dumpdb` command. Default is `named_dump.db`. The syntax of `dump-file` option in the "`options`" statement in the `/etc/named.conf` file is as shown below:

  ```
  dump-file "path_name";
  ```

  Where "path_name" is the path name of the file to which the server dumps the database.

- `statistics-file`

  This option is used to specify the pathname of the file in which the server appends statistics using the `rndc stats` command. Default is `named.stats` in the server's current directory. The syntax of `statistics-file` option in the "`options`" statement in the `/etc/named.conf` file is as shown below:

`statstics-file "path_name";`

The statistics file generated by BIND 9.2.0 is similar, but not identical, to that generated by BIND 8.1.2. For information on the format of the statistics file and the statistics counters, refer to the `named-conf(1)` man page distributed with this release.

- `blackhole`

  This option is used to specify a list of addresses from which the server will not accept queries or and does not use them to resolve a query. Default is `none`. The syntax of `blackhole` option in the "`options`" statement in the `/etc/named.conf` file is as shown below:

  ```
  [ blackhole {address_match_list {; ]
  ```

- `coresize`

  This option is used to specify the maximum size of a core dump. Default is `default`. The syntax of `coresize` option in the "`Options`" statement in the `/etc/named.conf` file is as shown below:

  ```
  [ blackhole {address_match_list {; ]
  ```

- `sortlist`

  The `sortlist` statement takes an `address_match_list` and interprets it. Each top level statement in `sortlist` must be an explicit `address_match_list` with one or two elements. The first element, which may be an IP address, IP prefix, `acl` name or a nested address_match_list is checked against the source address of the query until a match is found.

  Once the source address of the query has been matched, if the top level statement contains only one element, the actual element that matched the source address is used to select the address in the response to move to the beginning of the response. Each top level statement element is assigned a distance and the address in the response with the minimum distance is moved to the beginning of the response.

  A sample `sortlist` statement usage in the `Options` statement in the `/etc/named.conf` file is as shown below:

  ```
  [ sortlist { address_match_list }];
  ```

  NOTE: Refer to the `named.conf(4)` man page for more information on the usage of `sortlist` statement.

- `max-cache-size`

  `max-cache-size` is used to specify the maximum amount of memory to use for the server's cache, in bytes. When the amount of data in the cache reaches this limit, the server will cause records to expire prematurely so that the limit is not exceeded. In a server with multiple views, the limit applies separately to the cache of each view. The default is `unlimited`, meaning that records are purged from the cache only when their TTLs expire.

New Options in "zone" Statement

The following are the new zone options added in BIND 9.2.0:

- `update-policy`

  This is applicable only for master zones. When specified, one should ensure that allow-update is not present, else it is an error. A set of rules are specified, where each rule either grants or denies permissions for one or more names to be updated by one or more identities. If the dynamic update request message is signed (that is, it includes either a TSIG or SIG(0) record), the identity of the signer will be determined.

  A rule definition looks like this:

`(grant | deny ) identity nametype name [ types ]`

Each rule grants or denies privileges. Once a message has successfully matched a rule, the operation is immediately granted or denied and no further rules are examined.

The identity field specifies a name or a wildcard name. The nametype field has 4 values, name, subdomain, wildcard, and self.

If the nametype field is not specified, the rule matches all types except SIG, NS, SOA, and NXT Resource Records. Types may be specified by name, including "ANY" (ANY matches all types except NXT, which can never be updated).

- `max-transfer-time-out`

  This option is used to specify the time period for which Outbound zone transfers will run. Default is 2 hrs.

- `max-transfer-idle-out`

  This option is used to specify the time period for which Outbound zone transfers are idle. Default is 60 mins.

- `sig-validity-interval`

  This option is used to specify the expiry time of DNSSEC signature that is automatically generated as a result of updates. Default is 30 days.

- `match-clients`

  This option is used to specify the IP addresses of the namespace defined by each `view` statement.

- `zone`

  This option is used to specify the IP addresses of the namespace defined by each `view` statement.

- `View`

  This is an option that lets a nameserver answer a DNS query differently, depending on whether it is an internal query or external query. This is used to setup split DNS. All the below options of `view` are similar to those that are defined in "`options`" statement:

  1. `auth-nxdomain`
  2. `notify`
  3. `recursion`
  4. `also-notify`
  5. `forward`
  6. `forwarders`
  7. `allow-query`
  8. `allow-transfer`
  9. `allow-recursion`
  10. `query-source`
  11. `max-transfer-time-out`
  12. `max-transfer-idle-out`
  13. `max-cache-ttl`
  14. `max-ncache-ttl`
  15. `transfer-format`
  16. `transfer-source`
  17. `request-ixfr`
  18. `provide-ixfr`

19. `cleaning-interval`
20. `key`
21. `server`
22. `trusted-keys`
23. `sig-validity-interval`

An example of `View` (split DNS set-up) is as shown below:

```
view "internal" {
// This should match our internal networks
match-clients {10.0.0.0/8:};
//Provide recursive service to internal clients only
recursion yes;
//Provide a complete view of the example.com zone
// including addresses of internal hosts.
type master;
file "example-internal.db";
};
};
view "external" {
match-clients { any; };
// Refuse recursive service to external clients.
recursion no;
// Provide a restricted view of the example.com zone
// Containing only publicly accessible hosts.
zone "example.com" {
type master;
file "example-internal.db";
};
};
```

- `forwarders`

  This option can be used to specify the IP addresses to be used for forwarding. The forwarding facility can be used to create a large site-wide cache on a few servers, reducing traffic over links to external nameservers. This facility also allows queries by servers that do not have direct access to the Internet, but wish to look up exterior names. Forwarding occurs only on those queries for which the server is not authoritative and does not have an answer in its cache.

  The `forwarders` option is specified in the `/etc/named.conf` file as:

  ```
  [ forwarders { ip_addr [port ip_port] ;
  [ ip_addr [port ip_port] ; ... ] }; ]
  ```

- `allow-update`

  This option can be used to specify which hosts are allowed to submit Dynamic DNS updates for master zones. By default, updates from all hosts are denied.

NOTE: `allow-update` option is not applicable for slave zones. Refer to the `named.conf(4)` man page for more information.

New Option in "server" Statement

The `bogus` option can be used to prevent queries to a remote server which is giving out invalid data. The default value of `bogus` is `no`. The syntax of `bogus` option in the "`server`" statement is as shown below:

```
[ bogus yes_or_no ; ]
```

named-checkconf

This utility is used to check the syntax of `named.conf` file. However, it does not check for the semantics of the configuration file.

`named-checkconf` is run on the command line as:

```
/usr/sbin/named-checkconf [filename]
```

Where `filename` specifies the configuration file to be checked. The default `filename` is `/etc/named.conf`.

NOTE: Refer to the `named-checkconf(1)` man page for more information.

named-checkzone

This utility is used to perform syntax and consistency checks on the contents of a zone.

`named-checkzone` is run on the command line as:

```
/usr/sbin/named-checkzone [-dq] [-c class] zone [filename]
```

Where

`-d` is used to enable debugging.

`-q` is used to enable quiet mode for exit code only.

`-c class` is used to specify the class of the zone.

`zone` specifies the zone whose contents need to be checked.

`filename` specifies the file that should be used for checking the contents of a zone.

NOTE: Refer to the `named-checkzone(1)` man page for more information.

rndc

The remote name daemon control (`rndc`) program allows the system administrator to control the operation of a name server.

`rndc` is run on the command line as:

```
rndc [-c config] [-s server] [-p port] [-y key] command [command...]
```

Where

`-c config file` is used to specify an alternate configuration file. The default configuration file is `/etc/rndc.conf`.

`-s server` is used to specify the server whose operation needs to be controlled.

`-p port` is used to instruct `rndc` that it should send commands to TCP port number `port` on the system running the name server instead of BIND 9.2.0's default control channel port, 953.

`-y key` identifies the key-id to use from the configuration file.

and `command` is one of the following:

Table 8 rndc commands

Command	Description
`reload`	reload configuration file and zones
`reload zone [class [view]]`	reload the given zone
`refresh zone [class [view]]`	schedule zone maintenance for the given zone
`stats`	write serve statistics to the statistics file
`querylog`	toggle query logging
`dumpdb`	dump the current contents of the cache into the file specified by the dump-file option in named.conf.
`stop`	stop the server after saving any recent changes into the master files of the updated zones.
`halt`	stop the server immediately without saving any recent changes into the master files.
`reconfig`	reload configuration file and new zones only.
`trace`	increment debugging level by 1
`trace level`	change the debugging level
`notrace`	set debugging level to 0
`flush`	flush all the server's caches
`flush [view]`	flush the server's cache for a view
`status`	display the status of the server

NOTE: Refer to the `rndc(1)` man page for more information.

A sample `rndc.conf` file is distributed with this release of BIND. This file can be generated automatically by the `rndc-confgen` utility, which is distributed with BIND 9.2.0. For more information on `rndc-confgen`, read the `rndc-confgen` section above.

`rndc` has its own configuration file `/etc/rndc.conf`. A sample minimal configuration file looks like:

```
key rndckey {
algorithm "hmac-md5";
secret "IbtRYdcP8k2mVtel6aYfbQ==";
};
options {
default-server localhost;
default-key rndckey;
};
```

This file, if installed as `/etc/rndc.conf`, would allow the `$ rndc reload` command to connect to `127.0.0.1 port 953` and cause the nameserver to reload if a nameserver on the local machine is running with the following controls statements:`};`

```
controls {
inet 127.0.0.1 allow { 127.0.0.1; } keys { rndckey;
```

```
};
};
```

and also if the `named.conf` has an identical `key` statement for `rndckey`.

NOTE: Refer to the `rndc.conf(4)` man page for more information on the rndc configuration file.

Generating rndc.conf File

`rndc-confgen` can be used to generate `rndc.conf`, the configuration file for `rndc`. Alternatively, it can also be run with the `-a` option to set up a `rndc.key` file thus avoiding the need for a `rndc.conf` file and a `control` statement.

`rndc-confgen` is run on the command line as:

```
rndc-confgen [-a] [-b keysize] [-c keyfile] [-h] [-k keyname] [-p port]
[-r randomfile] [-s address] [-t chrootdir] [-u user]
```

Where

`"-a"` option is used to configure `rndc` automatically. This creates a file `rndc.key` in `/etc` which is read by both `rndc` and `named` on start-up.

`"-b keysize"` is used to specify the size of the authentication key in bits. The value must range between 1 and 512. Default is 128 bits.

`"-c keyfile"` is used with the `-a` option to specify an alternate location for the `rndc.key` file.

`"-h"` is used to print a short summary of the options and arguments to `rndc-confgen` utility.

`"-k keyname"` is used to specify the key name of the rndc authentication key. This must be a valid domain name. Default is `rndc-key`.

`"-p port"` is used to specify the command channel port where `named` listens for connections from `rndc`. Default is 953.

`"-r random file"` is used to specify a source file of random data for generating the authorization. Default is keyboard input.

`"-s address"` is used to specify the IP address where `named` listens for command channel connections from `rndc`. Default is the loopback address 127.0.0.1.

`"-t chrootdir"` is used with the `-a` option to specify a directory where `named` will run `chrooted`. An additional copy of the `rndc.key` will be written relative to this directory so that it will be found by the `chrooted named`.

`"-u user"` is used with the `-a` option to set the owner of the generated `rndc.key` file. If `-t` is also specified, the owner of the file in `chroot` area will be changed.

NOTE: Refer to the `rndc-confgen(1)` man page for more information.

New Command Line Options

Table 3-3 lists the new command line options that have been added for the various binaries and tools in BIND 9.2.0.

Table 9 New Command Line Options

Binaries/Tools	Options	Usage
dig	-b	Set the source IP address of the query to address. This must be a valid address on one of the host's network interfaces.
dig	-k	Sign the DNS queries sent by dig and their responses using transaction signatures (TSIG).
dig	-y	Specify the TSIG key on the command line.
dnssec-makekeyset & dnssec-signkey	-a	Verify all generated signatures.
dnssec-signkey	-c class	Specify the DNS class of the key sets. Currently only IN class is supported.
dnssec-signkey	-e end-time	Specify the date and time when the generated SIG records become invalid. If no end-time is specified, 30 days from the start time will be used as a default.
dnssec-signkey	-s start-time	Specify the data and time when the generated SIG records become valid. This can be either an absolute or relative time. If no start-time is specified, the current time will be used.
dnssec-signzone	-d directory	Look for signedkey files in directory as the directory.
dnssec-signzone	-h	Print a short summary of the options and arguments to dnssec-signzone.
dnssec-signzone	-i interval	Specify the cycle interval as an offset from the current time (in seconds). If a SIG record expires after the cycle interval, it is retained. Else, it is considered to be expiring soon and will be replaced. The default cycle interval is one quarter of the difference between signature end and start times. If neither end-time nor start-time is specified, dnssec-signzone generates signatures that are valid for 30 days and with a cycle interval of 7.5 days. If any existing SIG record expires in less than 7.5 days, they would be replaced.
dnssec-signzone	-n ncpus	Specify the number of threads to use. By default, one thread is started for each CPU.
dnssec-signzone	-o origin	Specify the zone origin. If no zone origin is specified, the name of the zone file will be considered as the origin.

Table 9 New Command Line Options *(continued)*

Binaries/Tools	Options	Usage
dnssec-signzone	`-t`	Print the performance statistics at the time of completion.
named	`-v`	Report the version number and exit.
named-checkconf	`-t`	`chroot` to directory to process `include` directives in the configuration file as if it is run by a similarly chrooted named.
named-checkconf	`-v`	Print the version number of `named-checkconf` and exit.
named-checkzone	`-v`	Print the version number of `named-checkzone` and exit.
nsupdate	`key {name} [secret]`	Specify that all updates need to be TSIG signed using the keyname keysecret pair. The `key` command overrides any key specified on the command line via -y or -k.
nsupdate	`local {address} [port]`	Send all dynamic update requests using the local address. If no local statement is provided, nsupdate will send updates using an address and port chosen by the system. port can also be used to set a specific port from where requests are sent. If port number is not specified, the system will assign one.
`nsupdate`	`send`	Send the current message. This is equivalent to entering a blank line.
`nsupdate`	`show`	Display the current message, containing all the pre-requisites and updates specified since the last `send` operation.
`rndc`	`-k keyname`	This option is used to specify the key name of the rndc authentication key. This must be a valid domain name. Default is `rndc-key`.

Changed Features

This section describes the changed features in BIND 9.2.0.

HP-specific Options

The following lists the HP-specific options added in BIND 9.2.0:

- `noforward`

  This option cannot be specified in "`options`" statement in BIND 9.2.0. Instead, forwarding can be suppressed by including an empty `forwarders` sub-statement as shown in the following example:

```
options {
forwarders { 192.249.249.1; };
}
zone "hp.com" {
```

```
type slave;
masters { 192.249.249.4; };
file "db.hp";
forwarders { };
}
```

This will supress queries like "foo.india.hp.com" from being forwarded to nameservers at 192.249.249.1.

NOTE: Forwarding to the nameservers available in the delegation information cannot be suppressed using an empty `forwarders` sub-statement.

- `alias-ip`

  This option is no longer supported. Use the `"listen-on"` option of the `"Options"` statement to implement the `alias-ip` option.

- `auth-nxdomain yes/no`

  If this option is specified as `yes`, then the AA bit is always set on NX domain responses, even if the server is not actually authoritative. The default value for this option has been changed from `"yes"` to `"no"`.

Compatibility with Previous Versions of BIND

This section provides the BIND compatibility information.

BIND 4.9.7 Compatibility

This section discusses the BIND 9.2.0-BIND 4.9.7 compatibility.

- BIND 9.2.0 uses a system assigned port for the UDP queries it makes rather than port 53 that BIND 4.9.7 uses. This may conflict with some firewalls. To specify a port, edit the /etc/named.conf file as follows:

  ```
  query-source address * port 53;

  transfer-source * port 53

  notify-source * port 53;
  ```

- BIND 9.2.0 no longer uses the minimum field to specify the TTL of records without a explicit TTL.

  Use the $TTL directive to specify a default TTL before the first record without an explicit TTL. The `hosts_to_named` script will create TTL value in the db files.

- BIND 9.2.0 does not support multiple CNAMEs with the same owner name. For example:

  ```
  www.example.com. CNAME host1.example.com
  www.example.com. CNAME host2.example.com.
  ```

  In the above example, multiple records with the same owner name "www.example.com" are not supported.

  The `named-checkzone` program can be used to check the correctness of the database files.

- BIND 9.2.0 does not support "CNAMEs with other data" with the same owner name, ignoring the DNSSEC records (SIG, NXT, KEY) that BIND 4.9.7 did not support. For example:

  ```
  www.example.com. CNAME host1.example.com.
  www.example.com. MX 10 host2.example.com.
  ```

- BIND 9.2.0 is less tolerant of errors in master files, so check your logs and fix any errors reported. The `named-checkzone` program can also be used to check master files.
- Outgoing zone transfers now use the "`many-answers`" format by default.This format is not understood by certain old versions of BIND 4.9.7.This problem can be resolved by using the option "`transfer-format one-answer;`", but HP recommends upgrading the slave servers.

BIND 8.1.2 Compatibility

This section discusses the BIND 9.2.0-BIND 8.1.2 compatibility.

- Configuration file compatibility
  - BIND 9.2.0 supports most of the options in `named.conf` file of BIND 8.1.2. BIND 9.2.0 issues a log message if the specified option is not implemented. It also logs the information if the default value is changed.
  - In BIND 9.2.0, `named` refuses to start if it detects an error in `named.conf`. Earlier versions would start despite errors, causing the server to run with a partial configuration.
  - In BIND 9.2.0, the "`logging`" statement only takes effect after the entire `named.conf` file has been read. In BIND 8.1.2, the new logging configuration took effect immediately after a "`logging`" statement was read.
  - The source address and port for notify messages and refresh queries is now controlled by "`notify-source`" and "`transfer-source`", respectively, as against `query-source` in BIND 8.1.2.
- Zone file compatibility
  - BIND 9.2.0 does not support serial numbers of SOA record with an embedded period, like "3.002". Serial numbers should be integers.
  - TXT records with unbalanced quotes, like 'host TXT "foo', were not treated as errors in previous versions of BIND. If the zone files contain such records, then error messages like "unexpected end of file"will be displayed because BIND 9.2.0 will interpret everything up to the next quote character as a literal string.
  - Previous versions of BIND accept RRs containing line breaks that are not properly quoted with parentheses. This is not legal master file syntax and will be treated as an error by BIND 9.2.0.

Documentation

BIND 9.2.0 documentation is available through its man pages. Table 3-4 lists and describes the man pages distributed with BIND 9.2.0.

Table 10 Man Pages

Man Page	Description
named.1m	Internet domain name server
dnssec-keygen.1	Key generation tool for DNSSEC
dnssec-makekeyset.1	Program used to produce a set of DNS keys.
dnssec-signkey.1	DNSSEC keyset signing tool
dnssec-signzone.1	DNSSEC zone signing tool
host.1	DNS lookup utility
nslookup.1	Program used to query nameservers interactively.

Table 10 Man Pages *(continued)*

Man Page	Description
nsupdate.1	Dynamic DNS update utility
lwresd.1m	Lightweight resolver daemon
rndc.1	Name server control utility
rndc.conf.4	`rndc` configuration file
sig-named.1m	Program used to send signals to the nameserver.
named-checkconf.1	`named` configuration file syntax checking tool
named-checkzone.1	Zone validity checking tool
hosts_to_named.1m	Program used to translate host table to name server file format.
dig.1m	Domain information groper
rndc-confgen.1	`rndc` key generation tool
named-conf.4	Configuration file for name daemon

Known Problems and Workarounds

The following are the known problems in BIND 9.2.0:

- In BIND 9.2.0, if duplicate data is available for a query, the duplicate data will not be dropped.
- Use of wildcard address "*" in "query-source address * port 53;" may not work as expected. Instead of the wildcard address "*", you need to use an explicit source IP address.

NOTE: HP recommends using `dig` instead of `nslookup`, as it may be obsoleted in the future releases of BIND.

Refer to the `dig(1m)` man page for information on the `dig` utility.

Limitations

The following is the limitation in BIND 9.2.0:

- The `rndc dump.db` command dumps only the cache information. You can run `dig axfr <domain>` command to obtain the db file information.