



VitalQIP[®]
Command Line Interface
Version 6.2
User's Guide

190-409-044R6.2
Issue 1
June 30, 2005

Lucent Technologies - Proprietary

This document contains proprietary information of Lucent Technologies and is not to be disclosed or used except in accordance with applicable agreements.

Copyright © 2005 Lucent Technologies
Unpublished and Not for Publication
All Rights Reserved

This material is protected by the copyright and trade secret laws of the United States and other countries. It may not be reproduced, distributed, or altered in any fashion by any entity (either internal or external to Lucent Technologies), except in accordance with applicable agreements, contracts, or licensing, without the express written consent of Lucent Technologies and the business management owner of the material.

Trademarks

All trademarks and service marks specified herein are owned by their respective companies.

Contents

About this document	xi
Purpose.....	xi
Reason for reissue.....	xi
Intended audience	xii
How to use this information product.....	xii
Conventions used.....	xiii
Information product support.....	xv
Technical support.....	xv
How to order.....	xvi
How to comment.....	xvi
1 VitalQIP database functions	1
Overview	1
Issuing CLI commands.....	3
Notation key.....	4
Common arguments/parameters.....	5
Using the CLI commands	7
qip-active.....	8
qip-admin.....	11
qip-altersubnet.....	21
qip-bootpgen.....	23
qip-changepassword.....	25
qip-check.....	26
qip-checkobjname.....	27
qip-clear.....	28
qip-crypt.....	29
qip-dbinit.....	30
qip-del.....	32
qip-delacldtemplate.....	35
qip-deladminrole.....	36
qip-delpolicy.....	37

qip-dhcpdebuglog	38
qip-dhcpdump.....	40
qip-dhcpgen.....	41
qip-dhcpsync.....	43
qip-dnsgen.....	45
qip-dnsupdate	47
qip-folder.....	49
qip-genddnsconfs	51
qip-getacltemplate.....	52
qip-getactiveobjectlst (for use with AutoDiscovery)	54
qip-getaddrangelst.....	58
qip-getadminrole	60
qip-getapplst.....	62
qip-getbillinfo.....	63
qip-getbtpsvrlst	65
qip-getclientclass.....	66
qip-getcontactlst	67
qip-getdebuglevel.....	69
qip-getdecnetaddr.....	71
qip-getdhcpscopes.....	72
qip-getdhcpsubnetlst	74
qip-getdhcpsvrlst.....	75
qip-getdnsrr	77
qip-getdnssvrlst.....	80
qip-getdomnlst	82
qip-getfolderlst.....	84
qip-getfreesubnetlst	86
qip-gethublst	88
qip-gethubport.....	89
qip-gethubslotlst	91
qip-getipaddr.....	92
qip-getlocalsvrlst.....	93
qip-getloclst.....	94
qip-getmaclst.....	96
qip-getmacmodellst	98
qip-getmacpools	100

qip-getnetlst.....	102
qip-getnissvrlst.....	103
qip-getnmdnsserver.....	104
qip-getobjectlst.....	106
qip-getobjectprof.....	108
qip-getobjname.....	115
qip-getorganization.....	117
qip-getospflst.....	118
qip-getospfprof.....	119
qip-getpolicy.....	121
qip-getprimdnssvrlst.....	123
qip-getrevzonest.....	125
qip-getrrlst.....	127
qip-getsecdnssvrlst.....	128
qip-getsnddr.....	129
qip-getsnglsl.....	130
qip-getsnglspof.....	131
qip-getsubnetlst.....	133
qip-getsubnetprof.....	135
qip-gettemplate.....	139
qip-gettemplst.....	141
qip-gettimesvrlst.....	143
qip-getudflst.....	144
qip-getudfamelst.....	147
qip-getuser.....	148
qip-getuseradrlst.....	150
qip-getzoneext.....	151
qip-getzoneprof.....	154
qip-globalmacpool.....	157
qip-hndbgen.....	159
qip-hostgen.....	160
qip-leasefilegen.....	161
qip-mcancel.....	162
qip-move.....	164
qip-msextract.....	166
qip-namingpolicy.....	167

qip-objectclass	170
qip-qdhcplease	173
qip-reclaim.....	174
qip-report.....	177
qip-rot13	181
qip-rptadminaudit.....	182
qip-rptadminrole	184
qip-rptaudithistory	186
qip-rptdhcp	188
qip-rptfreesubnet	192
qip-rptinquire	194
qip-rptmanaged.....	197
qip-rptobjectlst	199
qip-rptrole	203
qip-rptzonerr	205
qip-scope	207
qip-search.....	211
qip-searchacltemplates	216
qip-setacltemplate	218
qip-setadminrole	219
qip-setbillinfo	221
qip-setclientclass	223
qip-setcontact.....	225
qip-setdnsrr.....	227
qip-setdomainfolder	230
qip-setlocation	232
qip-setmacpools	234
qip-setnmdnsserver.....	236
qip-setobject.....	239
qip-setospfprof	248
qip-setpolicy.....	250
qip-setreclaimschedule.....	252
qip-setsnorgprof.....	255
qip-setsubnet	257
qip-setudf	262
qip-setuser.....	264

qip-setzonednsoptions	266
qip-setzoneext.....	271
qip-setzoneprof	274
qip-sitegen	277
qip-siteimport	279
qip-splitmergeenum	281
qip-splitrevzone	283
qip-subnetmacpool	284
qip-syncexternal	286
qip-template	289
qip-tombstonepurge	293
qip-ungetdechnetaddr.....	294
qip-ungethubport	295
qip-ungetipaddr.....	296
qip-unlock	297
qip-util	298
vercheck.....	303
2 Exporting and importing whole databases	305
Overview	305
qip-export.....	306
qip-import	308
3 Exporting and importing DNS and Bootptab files	311
Overview	311
Exporting DNS files.....	312
qip-dnscsv	315
Using the “Rules” file with qip-dnscsv.....	318
qip-dnscsv output files	324
Importing DNS files.....	329
Exporting Bootptab files	330
qip-bootptabcsv	331
Using qip-bootptabcsv.....	332
qip-bootptabcsv output files.....	333
Importing Bootptab files	336

4	Exporting and importing with CLIs	337
	Overview	337
	Importing files	338
	Order of importing	338
	enteraddrange	339
	enteralias	341
	entercontact	343
	enterdnsobj	345
	enterdnsrr	347
	enterdomain	350
	enterlocalobj	356
	enterlocation	358
	entermanufacturer	360
	enternetwork	362
	enterorganization	364
	enterospf	366
	enterreversezone	368
	enterserver.....	375
	entersimpleobj.....	379
	entersubnet.....	384
	entersubnetorg	387
	enterudf	390
	enteruser	392
	enterusergrp	395
	enterzoneext	397
	qipbulkload.....	399
	qiploadsndomn.....	404
	Exporting files with CLIs	406
	exportaddrange.....	407
	exportcontact.....	409
	exportdnsrr	410
	exportdomain	412
	exportlocation	414
	exportmanufacturer	415
	exportnetwork	416
	exportorganization	417

exportospf	418
exportreversezone	419
exportserver	420
exportsimpleobj	421
exportsubnet	423
exportsubnetorg	425
exportudf	426
exportuser	428
exportusergrp	429
exportzoneext	430

Index

433

About this document

Purpose

Welcome to VitalQIP® - a powerful IP name and address management tool. VitalQIP simplifies the assignment and allocation of IP addresses and services, such as DHCP and DNS. This product is a comprehensive collection of management tools and user interfaces. Each management tool and user interface provides the ability to plan, manage, and locally administer IP addresses and services across UNIX, Linux, and Windows platforms. VitalQIP works with directory services and RDBMS database configurations.

Reason for reissue

Table 1 lists the changes to the VitalQIP GUI in version 6.2 that required the *Command Line Interface User's Guide* to be reissued.

Table 1 Command Line Interface User's Guide changes

Feature Name	Description	Feature Impact
Multiple organizations per administrator	VitalQIP 6.2 supports the ability for an administrator to manage one, more than one, or all defined Organizations in VitalQIP.	
Object limits by Organization	VitalQIP 6.2 supports the ability to limit the number of defined objects per organization.	enterorganization, exportorganization, qip-getorganization
User defined object class and renaming	VitalQIP 6.2 supports the ability to name defined object classes and support multiple names of the same type. For example, customers may clone and rename "Wiring Hub" to "Switch" and "Host", and rename "PC" to "Modem" and also to "IAD".	

Feature Name	Description	Feature Impact
Allow DHCP template management on subnet org level	VitalQIP 6.2 provides the following additional fields on the Subnet Organization Profile: DHCP server and DHCP option template. On the Subnet Profile, a “Same as Subnet Org” option has been added to the drop-down for these fields.	
Add Dynamic Subnet Usage % to Subnet Profile	VitalQIP 6.2 provides an additional field on the Subnet Profile to indicate % utilization of dynamic objects defined on the subnet from the GUI or CLI.	
M-DHCP object name protection	VitalQIP 6.2 supports a global policy option which identifies whether M-DHCP/BOOTP object names may be overwritten/updated by DHCP clients.	
Push specified zones only	VitalQIP must support the ability for an administrator to specify a push for specified zones (selectable from list of relevant zones for the server in question) for Lucent DNS 4.0 servers only.	

Intended audience

This manual is intended for network administrators who are maintaining VitalQIP Version 6.2.

How to use this information product

This manual is organized as follows:

- Chapter 1 VitalQIP Command Line Interface This chapter outlines CLI command usage and explains the parameters that are common to all CLI commands. It also explains the notation used in explanations of the command parameters. Chapter 1 also explains all of the CLI commands (in alphabetic sequence) except those used for exporting and importing information.
- Chapter 2 Exporting and Importing Whole Databases This chapter explains how to use qip-export and qip-import to export and import entire VitalQIP databases. These CLI commands are used primarily for upgrades or backup/recovery purposes.

Chapter 3 Exporting and Importing
DNS and Bootptab Files

This chapter explains how to use the export/import utilities to extract current BIND 4.9.x, BIND 8.x, or bootp data, and import it into your VitalQIP system. These utilities can be used for upgrades or new installations

Chapter 4 Exporting and Importing
using the VitalQIP CLI

This chapter explains how to use the VitalQIP CLI commands to export and import data to/from the VitalQIP database. The commands allow you to export data for domains, OSPF areas, subnet organizations, subnets, MAC address pools, or objects. They also allow you to import data for domains, OSPF areas, subnet organizations, subnets, MAC address pools, or objects. These CLI commands can be used for upgrades or new installations.

Conventions used

Table 2 lists the typographical conventions used throughout this manual.

Table 2 Typographical conventions

Convention	Meaning	Example
boldface	Names of items on windows. Names of commands and routines. Names of buttons you should click.	Select the Client check box. The qip_getapplst routine returns the entire list of existing applications. Click OK .
Arial boldface	Names of keys on the keyboard to be pressed.	Press Enter to continue.
courier font	Input that you should enter from your keyboard.	Run the following command: <code>c:\setup.exe</code>
<angle brackets>	Variables for which you must substitute another value.	<code>http://<VitalQIP_server_IP_address_or_name></code>
[square brackets]	Variables in square brackets are optional. Variables that are not within square brackets are required.	<code>qip-active -n dhcp_svr [-g loginserver]</code>
italics	Names of manuals and emphasis.	Refer to the <i>VitalQIP Administrator Reference Manual</i> for more information.
Arial italic	Directories, paths, file names, e-mail addresses, and Uniform Resource Locators (URLs).	The VitalQIP web site is <code>http://qip.lucent.com</code> .
click	Click the left button on your mouse once.	To delete the object, click Delete .

Convention	Meaning	Example
right-click	Click the right button on your mouse.	Right-click on a service.
double-click	Double-click the left button on your mouse.	Double-click the book icon.

The VitalQIP product includes the following items:

- A set of VitalQIP CDs (part number: 190-409-041)
- *VitalQIP QuickStart Card* (part number: 190-409-039)
- *VitalQIP Administrator Reference Manual* (part number: 190-409-042)
This guide describes planning and configuring your network, information about the VitalQIP interface, advanced DNS and DHCP configurations, and troubleshooting.
- *VitalQIP Installation Guide* (part number: 190-409-043)
This guide describes how to install the VitalQIP product.
- *VitalQIP Command Line Interface User's Guide* (part number: 190-409-044)
This guide discusses and describes how to use the VitalQIP Command Line Interface.
- *VitalQIP User's Guide* (part number: 190-409-068)
This guide describes how to set up and use the VitalQIP user interface on Windows and UNIX platforms.
- *API Toolkit User's Guide* (part number: 190-409-033)
This guide describes how to install and use the VitalQIP Application Programming Interface (API) Toolkit. VitalQIP API Toolkit is a developer's toolkit comprised of the VitalQIP API routines and Lucent DHCP server API routines.
- *Audit Manager User's Guide* (part number: 190-409-034)
This guide describes how to install and use Audit Manager. Audit Manager tracks the historical activity of all IP addresses and leases given out by Lucent DHCP servers combined with static DNS addresses. The application is purchased separately and requires a license key.
- *Network Allocator User's Guide* (part number: 190-409-035)
This guide describes how to install and use Network Allocator. Network Allocator automates IP address space allocation and subnet creation. The application is purchased separately and requires a license key.
- *Registration Manager User's Guide* (part number: 190-409-036)
This guide describes how to install and use Registration Manager. Registration Manager provides the capability to track registered users, and the corresponding hardware address(es) associated with their personal computer(s). The application is purchased separately and requires a license key.

- *SNMP Module User's Guide* (part number: 190-409-038)
This guide describes how to install and use the Simple Network Management Protocol (SNMP) Module. The SNMP Module provides a standard way for management products to monitor network devices and services. The application is purchased separately and requires a license key.
- *Services Manager User's Guide* (part number: 190-409-037)
This guide describes how to install and use Services Manager. Services Manager monitors and controls the status of defined system services. The application is purchased separately and requires a license key.

Information product support

The Information Products & Training Group within Lucent offers cost-effective educational programs that cover the VitalQIP products and add-ons. Our offerings also include courses on the underlying technology for the VitalQIP products (for example, DNS and DHCP). Our classes blend presentation, discussion, and hands-on exercises to reinforce learning. Students acquire in-depth knowledge and gain expertise by practicing with our products in a controlled, instructor-facilitated setting. If you have any questions, please email us at vitaleducation@lucent.com.

Technical support

If you need assistance with VitalQIP, you can contact the Technical Assistance Center for your region. Contact information is provided in Table 3.

Table 3 Technical support information

Region	Address	Contact information
North, Central, and South America	Lucent Technologies 400 Lapp Road Malvern, PA 19355 USA	Phone: 1-866-LUCENT8 (582-3688) Option 5 Web: www.lucent.com/support
Europe, Middle East, Africa, and China	Lucent Technologies Chiltern House Sterling Court Broad Lane Bracknell, RG12 9GU UK	Phone: 00 800 00 LUCENT or +353 1 692 4579 E-mail: emeacallcenter@lucent.com Web: www.lucent.com/support

Region	Address	Contact information
Asia Pacific	Lucent Technologies Australia 68 Waterloo Rd North Ryde NSW 2113 Australia	Phone: 1800-458-236 (toll free from within Australia) (IDD) 800-5823-6888 (toll free from Asia Pacific - Hong Kong, Indonesia, South Korea, Malaysia, New Zealand, Philippines, Singapore, Taiwan, and Thailand) (613) 9614-8530 (toll call from any country) E-mail: apactss@lucent.com

How to order

Customers can order additional VitalQIP manuals online at
<http://www.cic.lucent.com/documents.html>.

How to comment

To comment on this information product, go to the Online Comment Form or email your comments to the Comments Hotline (comments@lucent.com).



1 VitalQIP database functions

Overview

The VitalQIP Command Line Interface (CLI) provides another alternative process to using the VitalQIP Graphical User Interface (GUI). Commands permit the use of a prompt to carry out functions. The CLI provides commands that allow you to do the following:

- Assign, reserve, dynamically allocate IP addresses
- Assign network objects to subnets
- Assign and maintain users and administrators
- Create and modify templates
- Manage objects
- Delete objects and de-allocate previously reserved addresses
- Move objects from one subnet to another, immediately or on a schedule
- Cancel scheduled moves
- Reclaim unused IP addresses
- Generate reports from VitalQIP
- Create DNS configuration and data files for a domain, and distribute them to the primary DNS server
- Generate the Bootptab file
- Generate DHCP configuration files
- Define a DHCP scope
- Show active leases for DHCP
- Synchronize VitalQIP with DHCP
- Export data from the database and import data into the database
- Clear entries in the audit table before a certain date

- Unlock selected and Planned Use addresses



Issuing CLI commands

The CLI commands are executable from various directories, depending on the platform. Refer to Table 4 to determine where you should execute the CLI commands.

Table 4 Location for execution of CLI commands

Platform	Executable location
Windows server	%QIPHOME%\cli
UNIX enterprise server	\$QIPHOME/usr/bin
Web Client on both Windows and UNIX	The cgi-bin directory where you installed VitalQIP cgi scripts.

A few things to keep in mind

- All data lines must end with a carriage return, or they are not imported.
- If you do not specify the [-s *servername*] [-u *username*] [-p *password*] parameters, the CLI looks in the *qip.pcy* file for the values. If you do not want to use these values taken from the *qip.pcy* file, you must specify them on the command line.
- The [-o *organization*] parameter is only applicable if you are running the CLI as a master administrator. The [-o *organization*] parameter is case sensitive for all CLI commands.
- For all CLI commands, the date and year format must be mm/dd/yyyy. If you use mm/dd/yy, the input data is rejected and an error message explaining that the date is invalid displays.
- All information passed on the command line must be surrounded with quotes if it contains a space. For example, passing the -i of *My File.txt* must be surrounded with quotes because it contains a space, therefore it would appear as -i "My File.txt".
- Field names in input files are case-insensitive. Values, however, continue to be case-sensitive. For example,
 - Name=user1
 - name=user1
 - NAME=user1
 are all the same. However,
 - Name=user1
 - Name=USER1
 - Name=User1
 are not the same.



Notation key

To make it easier for you to code the parameters, this manual uses the notations described in Table 5 when explaining the synopsis of each CLI command:

Table 5 Synopsis notation conventions

Notation	Description
Bold	Used for directories, filenames, commands, and parameters. Type the boldface term as it appears in the Synopsis. Example: Type qip-dbinit as: qip-dbinit
Italics	Used to show generic arguments and options; replace text in italics with your own values. Example: Type -i <i>input_filename</i> as: -i input1.txt
[]	Used to indicate optional elements in a description of syntax. Do not type the brackets themselves. Example: Type [-m] as: -m
	Used in syntax descriptions to separate items for which only one alternative can be chosen at a time. Example: Type -t active expired all as: -t active or -t expired or -t all
Constant width	Used to show the contents of files or the output from commands.



Common arguments/parameters

Table 6 lists parameters that are valid for many CLI commands.

Table 6 CLI command parameters

Parameter	Explanation	Important notes
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.	
-s <i>qip_dataserver</i>	Specifies the server on which to operate.	Optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>) – refer to note following table.
-u <i>username</i>	Specifies the user ID for the database.	Optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>) – refer to note following table.
-p <i>password</i>	Specifies the user password for the database.	Optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>) – refer to note following table.
-f <i>input_filename</i>	Specifies the name of the input file that contains the command parameters or the import information.	
-FO <i>Formatted Output</i>	Adds line feed and carriage return characters to raw XML output to improve legibility.	This parameter is not currently functional. Support for this feature will be added in a future VitalQIP release.
-? or -h	Displays the syntax of the command's parameters (to assist you in coding the command).	Enter -? or -h to display the syntax, without processing the command.
-v	Displays version information for the CLI command.	
-o <i>organization</i>	Specifies the user's organization. This parameter is ignored unless the user has system privileges (for example, qipman). The default is "VitalQIP Organization".	This parameter is only applicable if you are running the CLI as a master administrator. Also, this parameter is case sensitive for all CLI commands.
-ac <i>action</i>	Defines what action to take. You can add, modify, delete or retrieve information.	You can only perform one action at a time. You cannot add and modify in the same call, nor can you combine additions and modifications in one input file.

Important! A value must be provided. If the appropriate value is not defined in your policy file (*qip.pcy*), specify it in this parameter. If this parameter is omitted, the CLI looks for the value in the *qip.pcy* file.



Using the CLI commands

This section explains (in alphabetical sequence) the CLI commands used for functions other than exporting and importing information. The CLI commands for importing and exporting information are explained in other chapters of this document.

Important! In some tables in this chapter, the following notation is used to indicate whether input file values are required or optional.

[M] = Mandatory

[O] = Optional



qip-active

qip-active displays active leases for the specified DHCP server or deletes an active lease. The Active Lease Report lists IP addresses in ascending order. The Object and Domain Name that appear in the listing are determined, as follows:

- If the Object Name or Domain Name exists in the DHCP server's database, that name is displayed.
- If the Object Name or Domain Name does not exist in the DHCP server's database, the system uses the name stored within VitalQIP if the object is defined. However, the Object Name and/or Domain Name is prefixed with an asterisk (*) to indicate that the name was taken from VitalQIP.
- If the Object Name or Domain Name does not exist in the DHCP server's database and no Object Name or Domain Name exists within VitalQIP (for example, the object has not been defined), the Object Name and/or Domain Name is blank.

Synopsis

```
qip-active -n dhcp_svr [-g loginsvr] [-s servername] [-u username]  
[-p password] [-o organization] -b subnet_address|-j object_address  
[-r parsed_option82|unparsed_option82]  
[-t active|expired|all|delete]
```

Parameters

qip-active recognizes the following parameters:

-n <i>dhcp_svr</i>	Specifies the fully-qualified name of the DHCP server for which the lease information is to be retrieved.
-g <i>loginsvr</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies VitalQIP organization (corporation) name.

-b *subnet_address* | Specifies the Subnet or Object IP address for which lease information is retrieved.

-j *object_address*

-r *parsed_option82* | *Parsed_option82* - Specifies that the Option 82 information is output in user-friendly format, with 4 additional fields: Circuit ID, Remote ID, Device Class, and Subnet Selection.

unparsed_option82 | *Unparsed_option82* - Specifies that the Option 82 information is output in unparsed format.

-t *active|expired|all|delete* Specifies the type of retrieved lease information.

Command line input examples

- To view all active leases:

```
qip-active -u qipman -p passwd -n dhcpserver.quadritek.com -t all
```

- To view all active leases with a specific subnet address:

```
qip-active -u qipman -p passwd -n dhcpserver.quadritek.com -t all -b 198.200.138.0
```

- To delete an active lease:

```
qip-active -u qipman -p passwd -n dhcpserver.quadritek.com -t delete -b 198.200.138.3
```

Output example 1

A typical output file example follows. Refer to Table 7, “Output field definition”, on page 10 for a definition of the output fields.

```
IP_Address  MAC.Address  Expired
DHCP_Server Lease_Granted Lease_Expired
Last_Trans.Host_Name  Domain_Name
-----
-----
-----
144.144.144.013  00:60:97:40:ca:62 No
198.200.138.207 03/26/1997 02:22 05/14/2000 23:52
03/26/1997 02:22 MDLAPTOP quadritek.com
```

Output example 2

A typical output file example generated using the *-r* parameter follows. The Option 82 information is displayed with 4 additional fields: Circuit ID, Remote ID, Device Class, and Subnet Selection. Refer to Table 7, “Output field definition”, on page 10 for a definition of the output fields.

```
IP_Address  MAC_Address  Expired  DHCP_Server
Lease_Granted Lease_Expired Last_Trans.Host_Name
Domain_Name Circuit_ID Remote_ID Device_Class
```

Subnet_Selection

144.144.144.01 300:60:97:40:ac:62 No 198.200.138.207
03/25/2003 02:22 05/14/2003 23:52 03/25/2003 02:22 MDLAPTOP
quadritek.com 80010006 0ac1578dc3ab4e CCCM 198.200.138.0

Table 7 Output field definition

Field	Description
IP_Address	The IP address of the object.
MAC_Address	The MAC or hardware address of the object.
Expired	Yes -Lease has expired. No -Lease has not expired.
DHCP_Server	The IP address of the DHCP Server.
Lease_Granted	The date and time of when the lease was granted.
Lease_Expired	The date and time of when the lease expires.
Last_Trans	The date and time of when the last transaction occurred against this object.
Host_Name	The host name of the Object.
Domain_Name	The domain name of the Object.
Circuit ID	The agent-local identifier of the circuit from which a client-to-server packet was received. It is intended for use by relay agents in forwarding DHCP responses back to the proper circuit. For more information, see RFC 3046.
Remote ID	The identifier that identifies the remote host end of the circuit to a DHCP relay agent that terminates a switched or permanent circuit. For more information, see RFC 3046.
Device Class	Provides DOCSIS cable modem device attributes. For more information, see RFC 3256.
Subnet Selection	The subnet/link IP address requested by a DHCP relay agent. For more information, see RFC 3527.



qip-admin

qip-admin adds, modifies, deletes, and retrieves administrators and their managed organizations in VitalQIP.

Synopsis

```
qip-admin [-n admin_name] [-mo managed_organization] [-details]
          [-nc [add_new_contact] [-g loginserver] [-s dataserver] [-u username]
          [-p password] [-f output_file] [-i input_file] -ac action
          -df data_format
```

Parameters

qip-admin recognizes the following parameters:

- | | |
|--|---|
| <code>-n <i>admin_name</i></code> | Specifies the name of the administrator. |
| <code>-mo <i>managed_organization</i></code> | Specifies the name of the managed organization. |
| <code>-details</code> | Retrieves an entire profile, including administrator information, access information, and customization information. If an administrator is specified, customization options are also displayed.
For an explanation of profile details, see the Administrator section of “Chapter 3. Manage the VitalQIP Infrastructure” in the <i>VitalQIP User's Guide</i> .
Important! The <code>-df c</code> parameter (comma delimited data format) cannot be used with <code>-details</code> . |
| <code>-nc <i>add_new_contact</i></code> | Adds a new contact. |
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>dataserver</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |

- f *output_file* Specifies the directory and filename where the output data is to be placed.

- i *input_file* Specifies the directory and filename where the input data is to be placed.

-ac action

Defines what action to take. Only one action may be performed at a time. You cannot add and modify in the same call, nor can you combine additions and modifications in one input file.

The *action* variable can have the following values:

add - adds an administrator. Use the **-i** option to specify the input file that contains the data. Use the **-df** option to specify the format of the input file.

mod - modifies an administrator. Use the **-i** option to specify the input file that contains the data. Use the **-df** option to specify the format of the input file.

del - deletes an administrator. The **-n** parameter is also required. The administrator specified is removed from the database.

get - retrieves a specific administrator's profile or a list of all administrators:

- To retrieve a specific administrator's default profile, use the **-n** parameter with the administrator's name. The default profile contains the administrator's profile information, contact information, default system privileges, and GUI preferences. If the password field is included in the input file, it will not appear in the output.
- To retrieve a list of all administrators in the database, omit the **-n** parameter. The output for a list includes only the Login Name and Administrator Type.

Important! Managed organizations are not retrieved by default. If you need to retrieve an entire profile including managed organizations, also specify the **-details** parameter.

addorg - adds an organization. Use the **-i** option to specify the input file that contains the data. Use the **-df** option to specify the format of the input file.

modorg - modifies an organization. Use the **-i** option to specify the input file that contains the data. Use the **-df** option to specify the format of the input file.

getorg - retrieves a specific organization or a list of all organizations managed by an administrator:

- To retrieve a managed organization for an administrator, specify the administrator with the **-n** parameter and the managed organization with the **-mo** parameter.
- To retrieve a list of all managed organizations for an administrator, specify the administrator with the **-n** parameter and omit the **-mo** parameter. The output for a list includes only the Organization Name, Managed Type, and Read Only flag.

delorg - deletes an organization. Both the **-n** parameter and the **-mo** parameter are required. The organization specified in the **-mo** parameter is removed from the administrator specified in the **-n** parameter.

`-df data_format`

Determines the output format. The *data_format* variable has two values: *c* and *n*. Use the *c* variable to produce comma delimited output. Use the *n* variable to produce name value pairs output. If the `-df` parameter is omitted from the command line, the default is name value pairs output.

The output data format is identical to the input format shown in “Comma delimited file format” below, and “Name value pairs file format”, on page 15.

Important! The comma delimited data format cannot be used with the `-details` flag.

Comma delimited file format

The first line of a comma delimited list file contains a layout of the items that will be entered. The columns can be in any order and any optional columns can be excluded.

The next lines encountered will be the data lines. Multiple data lines may exist, each separated by a line feed.

Default system privileges

If any default system privileges are specified in the file, the ‘Default System Privileges’ and the ‘End Default System Privileges’ placeholder columns are required (data is never present in these fields).

The fields between these tags may vary based on the setting of other fields. For example, the ‘Create Billing Infrastructure’, ‘Create Infrastructure’, and ‘Create Resource Records’ fields are applicable if the ‘Highest Level GUI Mode’ is set to a value other than ‘Basic Mode’. If these fields are specified but the ‘Highest Level GUI Mode’ is set to ‘Basic Mode’, these fields are ignored. Consequently, if the ‘Highest Level GUI Mode’ is to be set, it should come before the children fields (or the children fields will be ignored). The best way to understand which fields are applicable is to check the GUI, where fields that are not applicable are not displayed.

Command line input examples

- To add an administrator in a comma delimited file, enter the following. The sample input file illustrates the contents of *c:\input.txt*.

```
qip-admin -u qipman -p passwd -ac add -i c:\input.txt -df c
```

Sample input file

```
Login Name, Type, Password, First Name, Last Name, Email, Phone, Pager,  
Business Unit ID, Default Printer, Default System Privileges, Highest  
Level GUI Mode, Create Billing Infrastructure, Create Infrastructure,  
Create Resource Records, Create/Update Administrator, Allow User  
Selection, Require User, Delete Confirmation Warning, Require Alias,  
Require Contact Name, Require Location, Require MAC Address, Require  
Manufacturer Information, Restrict CNAME, Restrict Subnet, Unique  
Name Warning, End Default System Privileges
```

```
admin1,Normal,password,,,,,,,,Advanced Mode, False, False, True,
True, True, False, True, True, False, False, False, False, False,
False, True,
```

- To add a Managed Organization to administrator Bob's Access List, enter the following. The details of the Managed Organization are contained in the comma delimited sample input file (*c:\input.txt*).

```
qip-admin -u qipman -p passwd -ac addorg -n Bob -i c:\input.txt -df c
```

Sample input file

```
Organization Name,Managed Type,Read Only, System Privileges,Use
Default,Highest Level GUI Mode,Create Billing Infrastructure,Create
Infrastructure,Create Resource Records,Create/Update
Administrator,Allow User Selection,Delete Confirmation
Warning,Network Allocator Admin,Require Alias,Require Contact
Name,Require Location,Require MAC Address,Require Manufacturer
Information,Restrict CNAME,Restrict Subnet,Unique Name Warning,End
Default System Privileges
Org2,Normal,False,,False,Advanced
Mode,False,False,False,False,False,False,False,False,False,
False,False,False,False,False,
```

Name value pairs file format

Each line of the file starts with a field name. The valid field names are identical to the column names for the comma delimited file format. The fields can be listed in order. More than one object may be specified in an input file. Objects should be separated by a blank line.

Important! The same logic regarding default system privileges applies to the name value pairs format. Refer to “Default system privileges”, on page 14.

Command line input examples

- To add an administrator in a name value pairs file (by default), enter the following. The input file example illustrates the contents of *c:\input.txt*.

```
qip-admin -u qipman -p passwd -ac add -i c:\input.txt
```

Sample input file

```
Login Name=admin1
Type=Normal
Password=password
First Name=
Last Name=
Email=
Phone=
Pager=
Business Unit ID=
Default Printer=
Default System Privileges
```

```

Network Allocator Admin=False
Highest Level GUI Mode=Advanced Mode
    Create Billing Infrastructure=False
    Create Infrastructure=False
    Create Resource Records=True
    Create/Update Administrator=True
Allow User Selection=True
    Require User=False
Delete Confirmation Warning=True
Require Alias=True
Require Contact Name=False
Require Location=False
Require MAC Address=False
Require Manufacturer Information=False
Restrict CNAME=False
Restrict Subnet=False
Unique Name Warning=True
End Default System Privileges

```

- To add a Managed Organization to administrator “Bob”’s Access List, enter the following. The details of the Managed Organization are contained in the comma delimited input file (*c:\input.txt*).

```
qip-admin -u qipman -p passwd -ac addorg -n Bob -i c:\input.txt -df n
```

Sample input file

```

Organization Name=Org2
    Managed Type=Normal
    Read Only=False
    Default System Privileges
        Use Default=False
            Highest Level GUI Mode=Advanced Mode
                Create Billing Infrastructure=False
                Create Infrastructure=False
                Create Resource Records=False
                Create/Update Administrator=False
            Allow User Selection=False
            Delete Confirmation Warning=False
            Network Allocator Admin=False
            Require Alias=False
            Require Contact Name=False
            Require Location=False
            Require MAC Address=False
            Require Manufacturer Information=False
            Restrict CNAME=False
            Restrict Subnet=False
            Unique Name Warning=False
        End Default System Privileges

```

- To retrieve information for administrator “Bob” into a name value pairs file. The content of *c:\output.txt* is shown in the output example below.

```
qip-admin -u qipman -p passwd -ac get -n Bob -f c:\output.txt -df n
```

- To delete the existing administrator “adminD”:

```
qip-admin -u qipman -p passwd -ac del -n adminD
```

- To modify the administrator “Bob” with the name value pairs file *c:\input.txt*.

```
qip-admin -u qipman -p passwd -ac mod -n Bob -i c:\input.txt
```

- To retrieve a list of all of Administrator “Bob”’s Managed Organizations. The list will contain each Managed Organization’s name, Managed Type, and Read/Write access.

```
qip-admin -u qipman -p passwd -ac getorg -n Bob
```

- To retrieve Administrator “Bob”’s Managed Organization “VitalQIP Organization”.

```
qip-admin -u qipman -p passwd -ac getorg -n Bob -mo "VitalQIP Organization"
```

Output Example

The following example shows sample output in *c:\output.txt* from the following example:

```
qip-admin -u qipman -p passwd -ac get -n Bob -f c:\output.txt -df n
```

For a description of the fields in this example, refer to “Administrators” in Chapter 3 of the *VitalQIP User’s Guide*.

```
Login Name=Bob
Type=Master
First Name=
Last Name=
Email=
Phone=
Pager=
Business Unit ID=
Default Printer=
Default System Privileges
    Delete Confirmation=False
    Require Alias=True
    Require Contact Name=False
    Require Location=False
    Require MAC Address=False
    Require Manufacturer Information=False
    Require User=False
    Unique Name Warning=True
End Default System Privileges
FilePref=1111
HelpPref=111
InfrastructurePref=111111111111
ManagementPref=111111111111
NetworkPref=1111111
PolicyPref=110100011100
```

```

ReportPref=111111111111
ViewPref=11111
ImportPref=1111111
UIQuickViewPref=000

```

Bit representation of menu preferences

In the preference fields above, such as InfrastructurePref, “1” and “0” are bit representations that indicate whether the menu option is on or off respectively. These menu settings correspond to the settings on the Customize menu in the Administrator Profile GUI. Refer to Table 8 to determine the menu preference by bit representation order.

Table 8 Bit representation of menu preferences

Preference field name	Order	Bit representation (1 equals on and 0 equals off)
FilePref	1	Administrator Password
	2	Change Organization
	3	Change Server/Administrator
	4	Exit
HelpPref	1	About VitalQIP
	2	Contents
	3	Technical Support
InfrastructurePref	1	Application Profile
	2	Domain Profile
	3	OSPF Profile
	4	User Group Profile
	5	Non-Managed DNS Server Profile
	6	Network/Reverse Zone Profile
	7	Organization Profile
	8	Server Profile
	9	Usage Billing
	10	Subnet Organization Profile
	11	Administrative Role Profile
	12	Administrator Profile

Preference field name	Order	Bit representation (1 equals on and 0 equals off)
ManagementPref	1 2 3 4 5 6 7 8 9 10 11	Go To Search Objects by Contact Go To Search Objects by Location Go To Search Objects Global MAC Address Pool Object Management All Subnets Object Management Used Subnets Object Management Unused Subnets QIP Hierarchy Reclaim Addresses User Management User Group User Management User Profile
NetworkPref	1 2 3 4 5 6 7	Bootptab File Generation DNS Generation DHCP Generation Local Host Generation NIS Generation View Active Leases Windows 2000 DC Generation
PolicyPref	1 2 3 4 5 6 7 8 9 10 11 12	ACL Template Contact Class/Option Setup Option Template Policy Template Global Policies Location Profile Manufacturer Profile Object Class Naming Policy User-Defined Fields Client Class

Preference field name	Order	Bit representation (1 equals on and 0 equals off)
ReportPref	1 2 3 4 5 6 7 8 9 10 11 12	Administrator Audit History Management Reports Administrator Profile Management Reports Free Subnet Management Reports DHCP Report Management Reports Inquire Report Management Reports Object by Location Report Management Reports Object by Address Range Report Management Reports Zone Resource Record Report Management Reports Object by Administrator Profile Report Management Reports Object by Administrative Role Report Management Reports Object by Application Report Object Audit History
ViewPref	1 2 3 4 5	Advanced Mode Basic Mode Hierarchy Legend Standard Mode Toolbar
ImportPref	1 2 3 4 5 6 7	Import Domain Import OSPF Import MAC Addresses Import Network Import Subnet Organizations Import Objects Import Subnets
UIQuickViewPref	1 2 3	Display Domain Folders Network Quick View Subnet Quick View



qip-altersubnet

qip-altersubnet provides the capability to split or join subnets.

Synopsis

```
qip-altersubnet [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] -f fromSubnets -t toSubnets
  -m newMask
```

Parameters

qip-altersubnet recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-f <i>fromSubnets</i>	Specifies an existing subnet address (to split) or addresses (to join), separated by a comma.
-t <i>toSubnets</i>	Specifies the resulting subnet address (after join) or addresses (after split), separated by a comma.
-m <i>newMask</i>	Specifies the new mask to be applied.

Command line input examples

- To join subnet addresses:

```
qip-altersubnet -u qipman -p passwd -f 144.144.32.0,144.144.48.0
,144.144.64.0 -t 144.144.32.0 -m 255.255.224.0
```

- To split subnet addresses:

```
qip-alternsubnet -u qipman -p passwd -f 135.135.0.0  
-t 135.135.0.0,135.135.0.128,135.135.1.0,135.135.1.128 -m 255.255.255.128
```



qip-bootpgen

qip-bootpgen generates the Bootptab file for the specified Bootp server. The file is then copied and output to the specified file.

Synopsis

```
qip-bootpgen -n server_name [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-bootpgen recognizes the following parameters:

- n *server_name* Specifies the fully-qualified Bootp server name for which the Bootptab file is generated.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the filename to which the Bootptab file is output.

Important! If the -f parameter is omitted, the Bootptab file will be generated based on the setting of parameter 'Bootptab Filename' in server profile. If the server profile 'Bootptab Filename' does not include the full path, the file will be generated in *QIPHOME/etc*.

Command line input example

```
qip-bootpgen -u qipman -p passwd-n bootpserver.quadritek.com -f /etc/bootptab
```

Output example

```
Defaults:\
  :ds=130.63.168.21 130.63.237.99 130.251.1.2:\
  :sm=255.255.255.0:\
  :to=18000:
```

```
Subnet243:\  
:tc=Global_Defaults\  
:ht=ethernet\  
:gw=130.63.243.1:
```



qip-changepassword

qip-changepassword changes the password of an administrator. For more information on passwords, refer to Chapter 1 of the *VitalQIP User's Guide*.

Synopsis

```
qip-changepassword -u loginname [-s servername] [-g loginserver]
  [-o org_name] -p password -n new_password
```

Parameters

qip-changepassword recognizes the following parameters:

- u *loginname* Specifies the VitalQIP login name for which you are changing the password.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- o *org_name* Specifies the VitalQIP organization (corporation) name.
- p *password* Specifies the password for the associated administrator account.
- n *new_password* Specifies the new password to replace the existing password.

Command line input example

```
qip-changepassword -u qipman -p passwd -n qipmanxyz
```



qip-check

qip-check checks the current date with the date of all reserved addresses. If the reserved addresses have expired, it sets the status of the object to “unused”.

Synopsis

```
qip-check [-g loginserver] [-s servername] [-u username] [-p password]  
          [-o organization]
```

Parameters

qip-check recognizes the following parameters:

- | | |
|-------------------------------------|---|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |

Command line input example

```
qip-check -u qipman -p passwd
```



qip-checkobjname

qip-checkobjname shows all addresses associated with the specified host name or alias name.

Synopsis

```
qip-checkobjname [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] -n object|alias_name
```

Parameters

qip-checkobjname recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- n *object|alias_name* Specifies the object's fully-qualified name or alias.

Command line input examples

```
qip-checkobjname -u qipman -p passwd -n myobject
qip-checkobjname -u qipman -p passwd -n www
```

Output examples

The following objects are using the same name:

```
10.98.4.12
100.97.4.12
```



qip-clear

qip-clear deletes entries in the audit table *before* the date specified. To clear all audit data, enter -d ALL.

Synopsis

```
qip-clear [-g loginserver] [-s servername] [-u username] [-p password]  
          [-o organization] -d mm/dd/yyyy|ALL
```

Parameters

qip-clear recognizes the following parameters:

- | | |
|---------------------------|---|
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -d <i>mm/dd/yyyy</i> ALL | Required. Specifies the date (in mm/dd/yyyy format) before which entries are cleared. To clear all audit data, enter -d ALL instead of a date. |

Command line input example

```
qip-clear -u qipman -g <loginservername> -s <servername> -p passwd  
-d 12/30/1996
```



qip-crypt

qip-crypt allows you to encrypt a password. **qip-crypt** takes the password as the first argument and sends a hex-string encrypted password to *STDOUT*. The **qip-crypt** CLI command must be run again and the new password placed in the *qip.pcy* file if the qipman's password, the qipadmin's password, the Schedule Password, or the Update Password is changed.

Important! **qip-crypt** is only intended for use with a password in the *qip.pcy* file. It should not be used to encrypt the database login. Database logins can be encrypted using third-party tools; refer to your system administrator for more information relevant to your database.

Synopsis

```
qip-crypt [password]
```

Parameters

qip-crypt recognizes the following parameters:

<i>password</i>	Specifies your current password.
-----------------	----------------------------------



qip-dbinit

qip-dbinit initializes the VitalQIP *or* Audit Manager database for the Sybase and Oracle database servers. For **qip-dbinit** details, refer to Chapter 9, “Database Administration” in the *VitalQIP Administrator Reference Manual*.

When using **qip-dbinit**, the full path must be specified (for example, enter `$QIPHOME/script/qip-dbinit`), or you must execute **qip-dbinit** from the `$QIPHOME/script` directory.

Important! Make sure that you back up your database and have shut down your VitalQIP services before running this CLI.

Synopsis

```
qip-dbinit [-t qip_dbase] [-g loginserver] [-s servername]
           [-u username] [-p password] [-l log_file] [-q output_file]
           [-i script_path] [-b db_name] [-a] [-k] [-d datapath]
           [-z encrypted_password]
```

Parameters

qip-dbinit recognizes the following parameters:

- | | |
|-----------------------------|--|
| <code>-t qip_dbase</code> | Specifies the database server type - either Oracle or Sybase . |
| <code>-g loginserver</code> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s servername</code> | Specifies the name of the database server. Note the following: <ul style="list-style-type: none">• The database server name must match the Sybase server name or the Oracle database alias name.• This parameter is optional if the QIPDATASERVER environment variable is set.• The command line argument overrides the environment variable. For Audit Manager, the Audit Manager database server name must be specified. |
| <code>-u username</code> | Specifies the VitalQIP or Audit Manager administrator account to be used in establishing the database connection. |
| <code>-p password</code> | Specifies the password for the associated administrator account. |
| <code>-l log_file</code> | Specifies the name of the log file. |

- q *output_file* Quiet Mode. If this parameter is omitted, the output is sent to *STDOUT*.
- i *script_path* Specifies the directory where the SQL scripts reside.
- b *db_name* Specifies the database name; QIP *or* LAM. (You cannot specify both databases at the same time.) The default is QIP.
- a Appends to the log file. The default is to overwrite the log file.
- k Skips the prompt.
- d *data_path* Specifies the directory name where the configuration and data files are stored. If this option is not specified, the files/data are written to the default directory where the database resides.
- z *encrypted_password* Allows commands to be executed with the encrypted password.
- c *config_action*



qip-del

qip-del removes objects and unallocates addresses that have been reserved. This function can also be used to delete objects of various “Owner Types” (refer to Table 9 following).

Important! An Object Range deleted by using **qip-del** does not delete objects associated with the Object Range. However, all ties with the Object Range are broken.

Synopsis

```
qip-del [-g loginserver] [-s servername] [-u username] [-p password]  
-n name|-a address [-o organization] [-r] [-l appl_name] [-m]  
[-t owner_type] [-c class]
```

Parameters

qip-del recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-n <i>name</i> -a <i>address</i>	Specifies the fully-qualified name of the object, domain, contact ID, location ID or subnet IP of the object where the reserved object resides (-n), or the address ranges (-a).
-r	Unallocates previously reserved addresses on the subnet specified in the -a option.
-l <i>appl_name</i>	The application name for which the addresses were reserved.
-m	Deletes all names if the fully-qualified name is used and more than one object/subnet exists with that name. This parameter is only applicable when you specify the -n parameter.
-t <i>owner_type</i>	Refer to Table 9 following.
-c <i>class</i>	Specified policy or user-defined field class. Only valid for -t Policy and -t Userfield.

Table 9 Owner types

Owner_Type	Description
Object	Deletes a used object or a reserved object.
Dhcp_server	Deletes the dhcp server by name
Dns_server	Deletes the dns server by name.
Object_range	Deletes the object range.
Domain	Deletes the domain and detaches all associations.
Reverse_zone	Deletes the zone and detaches all associations.
Network	Deletes the network.
Subnet	Deletes all objects in the subnet.
Subnet_organization	Deletes the subnet organization.
Address_range	Deletes the administrator address range.
User	Deletes the user via loginID.
Contact	Deletes a contact. Pass the ContactID in the -n parameter. You can determine the contact ID by using the qip-getcontactlst command.
Location	Deletes a location. Pass the LocationID in the -n parameter.
Policy	Deletes a Global Policy. Pass the Policy Class in the -c parameter (for example, General, Billing, DYNDNS, Oracle, Reports). Pass the Policy Name in the -n parameter.
Userfield	Deletes a User-Defined Field (UDF). Pass the UDF type in the -c parameter (for example, Domain, Object, Organization, Subnet, Reverse Zone, User). Pass the UDF Name in the -n parameter.

Command line input examples

- To delete all reserved objects within a subnet associated with application **appl2** (two examples):

```
qip-del -a 144.144.144.0 -r -l appl2 -s srv1 -u qipman -p passwd
qip-del -n qtek.com -r -l appl2 -m -s svr1 -u qipman -p passwd
```

- To delete domains:

```
qip-del -n qtek.com -t domain -s srv1 -u qipman -p passwd
```

- To delete object 198.200.138.123:

```
qip-del -u qipman -p passwd -a 198.200.138.123 -t object
```

- To delete all reserved objects in the subnet 198.200.138.192:

```
qip-del -u qipman -p passwd -a 198.200.138.192 -t subnet -r
```

- To delete an address range:

```
qip-del -u qipman -p passwd -t Address_Range -a 199.199.199.10-199.199.199.120
```



qip-delacldtemplate

qip-delacldtemplate deletes an ACL Template. Once called, the CLI queries the database for zone option associations. If there are any zones associated with the template, an error message tells the user that there are zones associated with the template. If the user still wants to delete the template, the `-sm` (suppress message) flag suppresses the message and turns off the checking in the database. It deletes the template and all associations automatically.

Synopsis

```
qip-delacldtemplate [-g loginserver] [-s servername] [-o organization]
                    [-u username] [-p password] -tn template_name -sm
```

Parameters

qip-delacldtemplate recognizes the following parameters:

- `-g loginserver` Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- `-s servername` Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-u username` Specifies the VitalQIP administrator account to be used in establishing the database connection.
- `-p password` Specifies the password for the associated administrator account.
- `-tn template_name` Specifies the name of the template being retrieved.
- `-sm` Specifies that messages are suppressed when this CLI runs.

qip-deladminrole

qip-deladminrole deletes an administrative role for the VitalQIP enterprise server.

Synopsis

```
qip-deladminrole -n role_name [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] -c [-f filename]
```

Parameters

qip-deladminrole recognizes the following parameters:

- n *role_name* Specifies the administrative role name you want to delete.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- c Request that the output is in CSV format.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-deladminrole -n "admin role" -u qipman -p passwd
```



qip-delpolicy

qip-delpolicy deletes a policy entry from the VitalQIP enterprise server.

Important! When you delete a policy, you are resetting the policy back to its default value. The policy still displays in the graphical user interface in the Global Policies. (For more information on the policies, refer to the “Global Policies” section in Chapter 2 of the *VitalQIP User’s Guide*.)

Synopsis

```
qip-delpolicy -c class -n policy_name [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization]
```

Parameters

qip-delpolicy recognizes the following parameters:

-c <i>class</i>	Specifies the class name associated with the policy you want to delete (for example, DNS).
-n <i>policy_name</i>	Specifies the policy name you want to delete (for example, PING_DELAY). For details about class lists and policy, refer to the “Define a DHCP/Bootp Template option” section in Chapter 2 of the <i>VitalQIP User’s Guide</i> .
-g <i>loginserver</i>	Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.

Command line input example

- To delete the **GENERAL PING_DELAY** policy from the VitalQIP enterprise server:
qip-delpolicy -s QIPSYBASE -u qipman -p passwd -c GENERAL -n PING_DELAY



qip-dhcpdebuglog

qip-dhcpdebuglog clears the debug log, changes the debug level, or stops the debug logging for a specified server, without restarting the server.

Synopsis

```
qip-dhcpdebuglog -n DHCP Server -c|-d new debug level |  
-t stop debug level [-g login_server] [-s dataserver]  
[-u username] [-p password] [-o organization] [-f output_file]
```

Parameters

qip-dhcpdebuglog recognizes the following parameters:

-n *DHCP Server* Specifies the DHCP server being debugged.

Important! One of the following three parameters, -c, -d or -t must be specified along with the -n parameter above.

-c Clears the debug log

-d *new debug level* Specifies the debug level for the server. Allowable values are:

- LevelCritical
- LevelError/
- LevelWarning
- LevelInfo
- LevelDebug

-t *stop debug level* Stops debugging the server.

-g *login_server* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

-s *dataserver* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

-u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

-p *password* Specifies the password for the associated administrator account.

- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-f output_file` Specifies the directory and filename where the output data is to be placed. The default filename is the path you are running the process from and is named *dhcpcdump.dat* in the current directory.



qip-dhcpdump

qip-dhcpdump reads the VitalQIP Lease Disk and outputs all active lease information to a “|” delimited output file.

Synopsis

```
qip-dhcpdump -i input_filename -f output_filename
[-t active|expired|all]
```

Parameters

qip-dhcpdump recognizes the following parameters:

- i *input_filename*** Specifies the VitalQIP Active Lease disk file path and name (for example, */\$QIPHOME/usr/bin/dhcp.db*).
On UNIX the default path for the *dhcp.db* file is in the directory of the *\$QDHCPCONFIG* environment variable.
On Windows the default path for the *dhcp.db* file is in *%QIPHOME%/dhcp*.
- Important!** The *dhcp.db* file can be placed in a directory other than the default directory.
For information on converting Active Lease files from previous versions of VitalQIP, refer to **qip-qdhcplease**.
- f *output_filename*** Specifies the directory and filename where the output data is to be placed. The default filename is the path you are running the process from and is named *dhcpdump.dat* in the current directory.
- t active|expired|all** Specifies whether you want “active” lease information, “expired” lease information, or “all” lease information.

Command line input example

```
qip-dhcpdump -i /QIPHOME/dhcp/dhcp.db -f /mydhcpdumplist.txt -t expired
```

Output example

```
144.144.144.14|00:80:5f:01:84:f4|ws_support_3|quadritek.com|19970307 13:12|19970307
13:42|Expired
144.144.144.15|00:80:5f:61:c4:91|AGRABIL|quadritek.com|19970311 05:34|19970311
06:04|Expired
135.16.53.13|00:40:97:40:ca:62|LAPTOP23|quadritek.com|19970314 12:36|19970314
13:06|Expired
135.16.53.14|00:40:4f:01:84:f4|ws_support_6|quadritek.com|19970307 13:12|19970307
13:42|Expired
135.16.53.15|00:40:4f:61:c4:91|ws_support_5|quadritek.com|19970311 05:34|19970311
06:04|Expired
```



qip-dhcpgen

qip-dhcpgen generates a DHCP configuration file for the specified DHCP server.

Synopsis

```
qip-dhcpgen -n dhcp_svr [-d directory] [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization] [-l]
[-f]
```

Parameters

qip-dhcpgen recognizes the following parameters:

-n <i>dhcp_svr</i>	Specifies the fully-qualified DHCP server name for which the DHCP configuration file is generated.
-d <i>directory</i>	Specifies the directory name to which the DHCP configuration file is generated. If this parameter is omitted, it is written to the directory defined for the DHCP server.
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-l	Generates the DHCP files on the local server.
-f	Forces a DHCP push to occur.

Command line input example

```
qip-dhcpgen -u qipman -p passwd -n dhcpserver.qtek.com -d c:\qip\dhcp
```

Output example

```
Server-identifier quad.qtek.com;
```

```
# Name: sol2
shared-network 198_200_130_0 {
  subnet 198.200.138.0 netmask 255.255.255.0 {
    manual-dhcp range 02-03-04-05-06-07 198.200.138.9 {
      option subnet-mask 255.255.255.0;
      option domain-name "qtek.com";
      option domain-name-servers
198.200.138.174;
      option routers 198.200.138.1;
      option dhcp-lease-time 4294967295;
    }
  }
}
```



qip-dhcpsync

qip-dhcpsync synchronizes VitalQIP with the true client-suggested names and their IP address bindings that are located in the DHCP server's Active Lease database. The process is illustrated in Figure 1. This command checks the DHCP server and obtains all active *and* expired leases known to the server.

Important! **qip-dhcpsync** is only used for recovery purposes and does not need to be run on a regular basis.

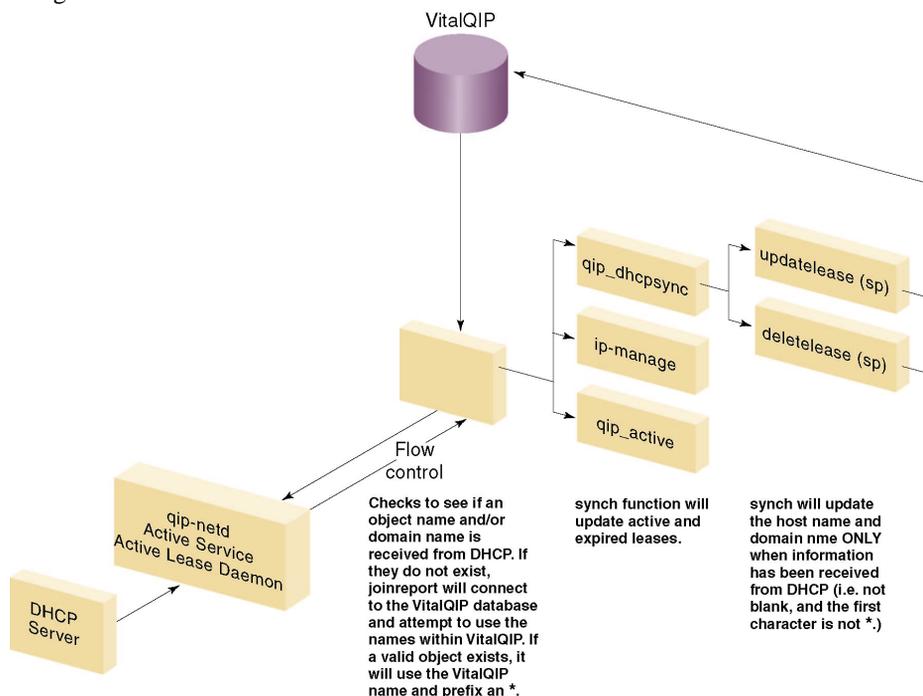


Figure 1 Name synchronization

When an Expired Object is updated in the VitalQIP database, the MAC address is removed and the host name reverts to the prior name. Likewise, when an Active Object is updated in the VitalQIP database, the MAC address, Object Name and Domain Name are updated according to the following rules:

1. If the Object Name or Domain Name exists in the DHCP database, and the "Accept Client Names" parameter in the Server Profile is set to True, the Object Name or Domain Name from the DHCP is used to update VitalQIP. The MAC address is also modified.
2. The Object *must* exist as a Dynamic Object; otherwise, no changes are made to VitalQIP.

If the Object Name or Domain Name do not exist in the DHCP database the Object Name or Domain Name is not changed within VitalQIP, and the “Accept Client Name” parameter in the Server Profile is set to False, only the MAC address is modified.

Synopsis

```
qip-dhcpsync -n dhcp_svr [-a subnet_address][-g loginserver]  
[-s servername] [-u username] [-p password] [-o organization]  
[-r reject_file]
```

Parameters

qip-dhcpsync recognizes the following parameters:

- | | |
|--------------------------|---|
| -n <i>dhcp_svr</i> | Specifies the fully-qualified name of the DHCP server you want to synchronize. |
| -a <i>subnet_address</i> | Specifies a subnet address; omit the parameter to synchronize all subnets. |
| -g <i>loginserver</i> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -r <i>reject_file</i> | Specifies the records that were not synchronized. |

Command line input example

```
qip-dhcpsync -u qipman -p passwd -n dhcpserver.qtek.com
```



qip-dnsgen

qip-dnsgen creates and distributes DNS configuration and data files for a DNS server. For Windows 2000 DNS servers, this CLI creates and distributes DNS files of “All Records.” This CLI will either accept an additional argument that specifies a comma separated list of zones or a file containing the zone names for Lucent DNS 4.x servers.

Synopsis

```
qip-dnsgen -n dns_svr [-g loginserver] [-s servername]
  [-o organization] [-u username] [-p password] [-d directory] [-l]
  [-z] [-f] [-t zone_filename]
```

Parameters

qip-dnsgen recognizes the following parameters:

- | | |
|------------------------|--|
| -n <i>dns_svr</i> | Specifies the fully-qualified primary DNS server name. |
| -g <i>loginserver</i> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -d <i>directory</i> | Specifies the directory name where the configuration and data files are stored. If this option is not specified, the files/data are written to the default directory for the DNS server specified.

Important! UNIX users. By default, the nameservers look for the DNS files under <i>/etc</i> . Therefore, if you wish to generate the DNS files to a directory other than <i>/etc</i> , you must set up a symbolic link from <i>/etc</i> to the working DNS directory. |
| -l | Generates files locally. |

- z Generates files for the changed zones only. If the `-z` parameter is not specified, all zones are generated.
- f Forces a push to the DNS server if the DNS server was locked previously. If it is still locked, an error message is returned, indicating it is “push locked”.
- t *zone_filename* This file contains the zones that *qip-dnsgen* uses to generate DNS files. This parameter is only valid for Lucent DNS 4.x.

Command line input examples

```
qip-dnsgen -u qipman -p passwd -n dnssvr.qtek.com -d /usr/etc  
qip-dnsgen -u qipman -p passwd -n dnssvr.qtek.com -d /usr/etc -z
```

Zone file example

The following is an example of a forward zone and a reverse zone as they would appear in a zone file:

```
abc.lucent.com  
200.200.200.0/24
```



qip-dnsupdate

qip-dnsupdate creates the DNS configuration and data files, increases the serial number in the SOA record, and restarts the server. For Lucent DNS 4.x servers, this CLI creates and distributes DNS files of “Changed Records Only.”

Synopsis

```
qip-dnsupdate -n dns_svr [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-d directory] [-l]
  [-z] [-f] [-m] [-t zone_filename]
```

Parameters

qip-dnsupdate recognizes the following parameters:

- n *dns_svr* Specifies the fully-qualified name of the primary DNS server.

- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- d *directory* Specifies the directory name where the configuration and data files are stored. If this option is not specified, the files/data are written to the default directory for the DNS server specified.

Important! UNIX Users. By default, the nameserver looks for the DNS files under */etc*. Therefore, if you wish to generate the DNS files to a directory other than */etc*, you must set up a symbolic link from */etc* to the working DNS directory.

- l Generates files locally.

- z Generates files for the changed zones only. If the **-z** parameter is not specified, all zones are generated.

- f Forces a push to the DNS server if the DNS server was locked previously. If it is still locked, an error message is returned, indicating it is “push locked”.
- m When doing a local push, updates the database as if a server push was specified.
- t *zone_filename* This file contains the zones that *qip-dnsgen* uses to generate DNS files. This parameter is only valid for Lucent DNS 4.x.

Command line input example

```
qip-dnsupdate -u qipman -p passwd -n dnsserver.quadritek.com
```

Zone file example

The following is an example of a forward zone and a reverse zone as they would appear in a zone file:

```
abc.lucent.com  
200.200.200.0/24
```



qip-folder

qip-folder adds, renames, or deletes folders used for grouping the domains in the Hierarchy of the GUI.

Synopsis

```
qip-folder -ac add|rename|del -n full path folder name
  [-r new folder name] [-g loginserver] [-s servername]
  [-o organization] [-u username] [-p password] [-h] [-v]
```

Parameters

qip-folder recognizes the following parameters:

-ac add rename del	The standard command line parameter, -ac, can have the following values: add, rename and del. Only one action is allowed per execution. Adding and modifying cannot be performed in the same call. When the value for -ac is add or del, the full path folder name must be provided. When the value for -ac is rename, both the -n full path folder name and the -r new folder name must be provided.
-n <i>full path folder name</i>	Enter the full path name of the folder to be added, renamed, or deleted.
-r <i>new folder name</i>	Only use this parameter if the -ac rename action is to be entered. This parameter establishes the new name of the folder.
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.

Command line input examples

- To create a new folder:

```
qip-folder -ac add -s server1 -u qipman -p passwd -o QIPOrg  
-n topfolder/nextfolder/bottomfolder
```

- To rename the “bottomfolder” to “newbottomfolder”:

```
qip-folder -ac rename -s server1 -u qipman -p passwd -o QIPOrg  
-n topfolder/nextfolder/bottomfolder -r newbottomfolder
```

- To delete an existing folder (the folder must be empty):

```
qip-folder -ac del -s server1 -u qipman -p passwd -o QIPOrg  
-n topfolder/nextfolder/bottomfolder
```



qip-genddnsconfs

qip-genddnsconfs generates the *<org>.DDNS.conf* files for organizations. It is called periodically by the DNS Update Service when the service is on the enterprise server. It accepts the command line parameter **-R** to restart the DNS Update Service when it has completed generating the files. It is installed on the enterprise server.

Synopsis

```
qip-genddnsconfs [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] [-R]
```

Parameters

qip-genddnsconfs recognizes the following parameters:

- | | |
|-------------------------------|---|
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -R | Restarts the DNS Update Service when the service is on the enterprise server. |

Command line input example

- To generate *ddns.conf* files for organizations:
qip-genddnsconfs
- To generate the *ddns.conf* files for organization 1:
qip-genddnsconfs -o 1
- To restart the DNS Update Service after generating *ddns.conf* files for organizations:
qip-genddnsconfs -R



qip-getacltemplate

qip-getacltemplate lists the values for a specific template.

Synopsis

```
qip-getacltemplate [-g loginserver] [-s servername] [-o organization]  
                  [-u username] [-p password] [-f output_file] -tn template_name
```

Parameters

qip-getacltemplate recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *output_file* Specifies the file where the report is stored. The default is *STDOUT*.
- tn *template_name* Specifies the name of the template being retrieved.

Input file example

This file is the input file for **qip-setacltemplate** on page 218. It contains the following elements:

Field	Description
Name	Name of the ACL template
Template Components	Template components include any combination of the following: <ul style="list-style-type: none">- IP Addresses- Network Addresses- Localhost

Example

```
testname  
135.119.106.0  
145.0.0.0/10  
localhost
```



qip-getactiveobjectlst (for use with AutoDiscovery)

qip-getactiveobjectlst constructs a list of all objects matching the Request Profile's list of subnets, and extracts object data for the specified organization and object list as follows:

- Unused objects will be excluded.
- Reserved and Selected objects will be excluded.
- Tombstoned objects will be excluded.
- All other objects with status = Static will be included.
- All other objects with status = Dynamic that are associated with a DHCP server (MAC Address is populated and object has a non-expired lease) will be included.
- Manual BOOTP objects will be included.

Synopsis

```
qip-getactiveobjectlst [-g loginserver] [-s servername]  
-o organization [-u username] [-p password] -i input_file  
[-f output_file]
```

Parameters

qip-getactiveobjectlst recognizes the following parameters:

- | | |
|------------------------|---|
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. If not provided, the value of the VitalQIP LOGIN environment variable will be used. |
| -s <i>servername</i> | Specifies the VitalQIP database server. If not provided, the value of the VitalQIP QIPDATASERVER environment variable is used. |
| -u <i>username</i> | Specifies the VitalQIP administrator login. This parameter is required, but is optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>) |
| -p <i>password</i> | Specifies the VitalQIP administrator login password. This parameter is required, but is optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>). |
| -o <i>organization</i> | Required. Specifies the VitalQIP organization (corporation) name. |
| -i <i>input_file</i> | Required. A directory and filename of the input data. The input data is a list of one or more subnets, one per line, that the CLI will extract from the VitalQIP database. Entries represent network address blocks and bits, for example: 135.222.146.0/24. |

`-f output_file` A directory and file name for the CLI output. If no filename is provided, the output will be to *STDOUT*.

The following information is included for each object in the CLI output:

- ObjectAddress
- MACAddress (formatted with colons)
- ObjectName
- DomainName
- ContactLastName
- ContactFirstName
- ContactEmail
- ContactPhone
- ContactPager
- RoomID
- Street1
- Street2
- City
- State
- Zip
- Country
- Objectclass
- ObjectType-a combination of Allocation Type and Dynamic Type, as follows:
 - Static
 - Dynamic_none
 - Manual_Bootp with non-expired lease
 - Manual_DHCP
 - Automatic_Bootp
 - Automatic_DHCP
 - Dynamic_DHCP
- ObjectDescription

Important! The output file is in XML format.

Command line input example

`qip-getactiveobjectlst -u qipman -p passwd -i infile.txt -f outfile.txt`

Input File Example

An input file consists of a list of one or more subnets, one per line, for example:

```
172.16.7.0/26
172.16.7.64/26
172.16.8.0/26
172.16.8.64/26
172.16.8.128/26
```

Output example

The following is an output file example for

```
qip-getactiveobjectlst -o "Active Objects Organization" -u qipman
-p passwd -i /import/infile.txt:
<qapi-active-obj-list>
<qapi-active-obj ip-address="172.16.7.65" name="udp000070uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Dynamic_DHCP">
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.66" name="udp000071uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Dynamic_DHCP">
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.79" name="udp000084uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Automatic_DHCP">
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.80" name="udp000085uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Automatic_DHCP">
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.81" name="udp000086uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Manual_DHCP">
<mac>87:de:0b:26:00:00</mac>
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.91" name="udp000094uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Automatic_Bootp">
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.1" name="qipsfv1204"
  domain="activeobj.net" object-class="Server" object-type="Static">
<mac>00:03:ba:7a:c9:b7</mac>
<description>Server</description>
<room>2C-335</room>
<contact-first-name>Admin</contact-first-name>
<contact-last-name>Master</contact-last-name>
<contact-email>masteradmin@example.com</contact-email>
<contact-phone>555-8153</contact-phone>
<contact-pager>12345@example.com</contact-pager>
```

```
<street1>600 Somewhere Lane</street1>
<street2>Room A</street2>
<city>Somewhereville</city>
<state>PA</state>
<zip>19355</zip>
<country>USA</country>
</qapi-active-obj>
<qapi-active-obj ip-address="172.16.7.90" name="udp000093uds"
  domain="activeobj.net" object-class="Undefined" object-
  type="Manual_Bootp">
<mac>87:de:0b:26:00:13</mac>
<description>Manual B00TP Object</description>
<contact-first-name>Admin</contact-first-name>
<contact-last-name>Master</contact-last-name>
<contact-email>masteradmin@example.com</contact-email>
<contact-phone>555-8153</contact-phone>
<contact-pager>12345@example.com </contact-pager>
</qapi-active-obj>
</qapi-active-obj-list>
```



qip-getaddrangelst

qip-getaddrangelst retrieves and lists the defined network address ranges in the VitalQIP database. This CLI command is part of the **Infrastructure|Network** option, and can be executed through the **Address Range** tab of the Network Profile. (For details, refer to Chapter 4 of the *VitalQIP User's Guide*.) The output is stored in the specified (or default) file.

Synopsis

```
qip-getaddrangelst -t address_type [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization]
[-n login_name|-a ip_address] [-f filename]
```

Parameters

qip-getaddrangelst recognizes the following parameters:

-t <i>address_type</i>	Specifies the address range type (network or subnet).
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-n <i>login_name</i> -a <i>ip_address</i>	Specifies either -n (to limit the retrieval to what is managed by a specific administrator) or -a (to limit the retrieval to a specific network).
-f <i>filename</i>	Specifies the file where the report is stored. The default is STDOUT .

Command line input examples

- To obtain a list of address ranges an administrator manages:

```
qip-getaddrangelst -u qipman -p passwd -s qipserver -n admin1 -t network
```

- To list the existing address ranges on network “10.10.0.0”:

```
qip-getaddrangelst -u qipman -p passwd -s qipserver -a 10.10.0.0 -t network
```

- To list all address ranges in VitalQIP database and write it to “*addrRange.dat*”:

```
qip-getaddrangelst -u qipman -p passwd -s qipserver -f addrRange.dat -t subnet
```

Output example

```
10.10.0.0,10.10.1.10,10.10.2.200
```

200,201,202,0,203,204,205,206,207,208,209,210,211,212



qip-getadminrole

qip-getadminrole displays an administrative role.

Synopsis

```
qip-getadminrole [-g loginserver] [-s servername] [-u username]  
                 [-p password] [-o organization] -n role_name [-f filename]
```

Parameters

qip-getadminrole recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- n *role_name* Specifies the administrative role name you want to display.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getadminrole -u qipman -p passwd -n adminrole -f c:\temp\output.txt
```

Output example

```
RoleName=test role  
RoleDescription=This is a test role  
Managed List:  
Application,Sample Application,True  
Subnet Organization,psl_suborg1,True  
Domain,usa.world.com,True  
Network,net_73,73.0.0.0,True  
Server,DHCP-DHCP1,name.com,0.0.0.0,True  
Server,DOMAIN CONTROLLER-dc.domain.com,0.0.0.0,True
```

AddressRange,170.10.0.0-170.10.20.0,False
ObjectRange,14.14.14.1-14.14.14.100,False



qip-getapplst

qip-getapplst retrieves the entire list of existing application(s) from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getapplst [-g loginserver] [-s servername] [-u username]  
              [-p password] [-o organization] [-f filename]
```

Parameters

qip-getapplst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getapplst -u qipman -p passwd -f c:\temp\output.txt
```

Output example

```
"Engineering"  
"Computer Club"
```



qip-getbillinfo

qip-getbillinfo returns a list of all the billing locations, billing user groups, or billing object classes (based on the **-t** parameter) for that organization. The output is stored in the specified (or default) file.

Synopsis

```
qip-getbillinfo [-g loginserver] [-s servername] [-u username]
                [-p password] [-o organization] -t Location|ObjectClass|UserGroup
                [-f filename]
```

Parameters

qip-getbillinfo recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-t <i>Location ObjectClass UserGroup</i>	Specifies the billing location(s), billing object class(es), or billing user group(s) associated with that organization.
-f <i>filename</i>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .

Command line input example

```
qip-getbillinfo -u qipman -p passwd -t ObjectClass -f c:\temp\output.txt
```

Output example

```
Bridge
Dynamic_Allocation
Gateway
Legacy_System
Others
```

PC
Printer
Router
Server
Switch
Terminal_Server
Test_Equipment
Undefined
Wiring_HUB
Workstation
X-terminal



qip-getbtpsvrlst

qip-getbtpsvrlst retrieves the entire list of fully-qualified Bootp server(s) from the VitalQIP database for the organization specified. The output is stored in the specified (or default) file.

Synopsis

```
qip-getbtpsvrlst [-g loginserver] [-s servername] [-u username]
                 [-p password] [-o organization] [-f filename]
```

Parameters

qip-getbtpsvrlst recognizes the following parameters:

- | | |
|-------------------------------------|---|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>filename</i></code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input example

```
qip-getbtpsvrlst -u qipman -p passwd -f c:\temp\output.txt
```

Output example

```
qbootp.qtek.com
ibootp.qtek.com
```



qip-getclientclass

Use the **qip-getclientclass** CLI to obtain a list of client classes and their associated vendor classes or user classes.

Synopsis

```
qip-getclientclass [-g loginserver] [-s servername] [-u username]  
                  [-p password] [-o organization]  
                  [-n client class] -f filename
```

Parameters

qip-getclientclass recognizes the following parameters:

- | | |
|-------------------------------------|--|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used for establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>filename</i></code> | Specifies the directory and filename of the input data. If the name contains spaces, it must be enclosed by quotation marks. |
| <code>-n <i>client class</i></code> | Specifies the client class information. |

Output file content example

For a description of the fields in the output file example, refer to Table 32, "Input file field definitions", on page 224.

```
ClientClassName=new client class  
ClassType=USER  
AttachedClass=qa support  
OptionTemplate=general  
ClientClassPolicyTemplate=policyTemp
```



qip-getcontactlst

qip-getcontactlst retrieves a list of the contact information for all contacts within the VitalQIP database, or for the contacts that match the search criteria specified in the *search_tokens* file. The output is stored in the specified (or default) file.

Synopsis

```
qip-getcontactlst [-g loginserver] [-s servername] [-u username]
                  [-p password] [-o organization] [-f filename]
                  [-t search_tokens_filename]
```

Parameters

qip-getcontactlst recognizes the following parameters:

<code>-g <i>loginserver</i></code>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
<code>-s <i>servername</i></code>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
<code>-u <i>username</i></code>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
<code>-p <i>password</i></code>	Specifies the password for the associated administrator account.
<code>-o <i>organization</i></code>	Specifies the VitalQIP organization (corporation) name.
<code>-f <i>filename</i></code>	Specifies the file to contain the retrieved information. The default is <i>STDOUT</i> .
<code>-t <i>search_tokens_filename</i></code>	Specifies the name of the file that contains the search criteria to match. If this file is present, the system returns contact information for the contacts that match the search criteria. Otherwise, contact information for <i>all</i> contacts are returned. Refer to Table 10 for sample search criteria.

Table 10 Search token criteria

Search Tokens file fields	Description
FirstName	The first name of the contact. Note: An implied “Begins with” search is performed. For example, enter Jo to return contact information for all contacts having first names that start with “Jo” (such as Joseph, Joanne, Joan).
LastName	The last name of the contact. Note that an implied “Begins with” search is performed. For example, enter Sm to return contact information for all contacts having last names that start with “Sm” (such as Small, Smedley, Smith).

The Input and Output examples following are produced by using the **-t allcontact.txt** parameter specification. The *allcontact.txt* file contains the following search criteria:

```
LastName=pa1
```

Command line input example

```
qip-getcontactlst -u qipman -p passwd -t allcontact.txt
```

Output example

For a description of the fields in the following output file, refer to Table 11 following.

```
"2" "pa1" "pa1" "pa1@qtek.com" "111-0033" "1-800-111-3300"
"3" "pa11" "pa1,jr I" "pa11@qtek.com" "111-3333" "1-800-111-3311"
"4" "pa12" "pa1,jr II" "pa12@qtek.com" "222-3333" "1-800-111-3322"
"5" "pa13" "pa13" "pa13@qtek.com" "333-3333" "1-800-111-3333"
"7" "pa15" "pa15" "pa15@qtek.com" "111-3555" "1-800-111-3355"
"8" "pa16" "pa16" "pa16@qtek.com" "111-3366" "1-800-111-3366"
"9" "pa17" "pa1" "pa17@qtek.com" "111-3377" "-800-111-3377"
```

Table 11 Output field description

Field	Description
ContactID	The identification of the contact.
LastName	The contact’s last name.
FirstName	The contact’s first name.
EmailAddress	The contact’s email address.
PhoneNumber	The contact’s phone number.
PagerNumber	The contact’s pager number.



qip-getdebuglevel

qip-getdebuglevel retrieves the current debug level, if one is set. If the debug level is set, the CLI returns one of the following values:

- levelCritical
- levelDebug
- levelError
- levelInfo
- levelWarning

Synopsis

```
qip-getdebuglevel -n dhcp_servername [-g loginserver] [-s dataserver]
[-u username] [-p password] [-o organization] [-f output_file]
```

Parameters

qip-getdebuglevel recognizes the following parameter

- | | |
|--|---|
| <code>-n <i>dhcp_servername</i></code> | Specifies the DHCP server name for which you want to obtain the debug level. |
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>dataserver</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>output_file</i></code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input example

```
qip-getdebuglevel -n qipserver
```

Output example
levelInfo



qip-getdecnetaddr

qip-getdecnetaddr retrieves the next available DECNet address from a specified DECNet area of the VitalQIP database.

Synopsis

```
qip-getdecnetaddr -a decnet_area [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization]
```

Parameters

qip-getdecnetaddr recognizes the following parameters:

- a *decnet_area* Specifies the DECNet area where you want to retrieve the DECNet address, from a DECNet node value of 1-63.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.

Command line input example

```
qip-getdecnetaddr -u qipman -p passwd -a 10
```

Output example

```
DECNetNode=6
DECNetAddress=AA00040000604
```



qip-getdhcpscopes

qip-getdhcpscopes provides a list of all configure DHCP address ranges for the specified organization.

Synopsis

```
qip-getdhcpscopes [-g loginserver] [-s servername] -o organization
  [-u username] [-p password] [-f output_file]
```

Parameters

qip-getdhcpscopes recognizes the following parameters:

- | | |
|-------------------------------------|---|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. If not provided, the value of the VitalQIP LOGIN environment variable will be used. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server name. If not provided, the value of the VitalQIP QIPDATASERVER environment variable shall be used. |
| <code>-o <i>organization</i></code> | Required. Specifies the VitalQIP organization name. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator login. This parameter is required but is optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>). |
| <code>-p <i>password</i></code> | Specifies the VitalQIP administrator login password. This parameter is required but is optional if the appropriate value is defined in your policy file (<i>qip.pcy</i>). |
| <code>-f <i>output_file</i></code> | A directory and file name for the CLI output. If no filename is provided, the output will be to <i>STDOUT</i> . |

Command line input example

```
qip-getdhcpscopes -u qipman -p passwd -f extfile.txt
```

Output example

The following is an example of an output file for `qip-getdhcpscopes -o "Active Objects Organization" -u qipman -p passwd`.

```
<qapi-range-list>
  <qapi-range start-addr="111.222.11.1" end-addr="111.222.11.1">
  </qapi-range>
  <qapi-range start-addr="172.16.7.90" end-addr="172.16.7.90">
```

```
<qapi-range start-addr="172.16.7.65" end-addr="172.16.7.75">
</qapi-range>
<qapi-range start-addr="172.16.7.76" end-addr="172.16.7.80">
</qapi-range>
<qapi-range start-addr="172.16.7.81" end-addr="172.16.7.85">
</qapi-range>
<qapi-range start-addr="172.16.7.91" end-addr="172.16.7.91">
</qapi-range>
</qapi-range-list>
```



qip-getdhcpsubnetlst

qip-getdhcpsubnetlst makes a list of the subnets that are serviced by a specific DHCP server. The output is stored in the specified (or default) file.

Synopsis

```
qip-getdhcpsubnetlst -n dhcp_server_name [-g loginserver]  
[-s servername] [-u username] [-p password] [-o organization]  
[-f filename]
```

Parameters

qip-getdhcpsubnetlst recognizes the following parameters:

- n *dhcp_server_name* Specifies the fully-qualified DHCP server name from which you are retrieving subnets.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getdhcpsubnetlst -u qipman -p passwd -n dhcpserver1.com -f dhcpsubnetlst.txt
```

Output example

```
144.144.144.0  
198.200.234.0
```



qip-getdhcpsvr1st

qip-getdhcpsvr1st retrieves the entire list of existing DHCP server(s) from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getdhcpsvr1st [-g loginserver] [-s servername] [-u username]
  [-p password] [-n subnet_address] [-o organization] [-f filename]
  [-all] [-b show_server_type(Y/N)]
```

Parameters

qip-getdhcpsvr1st recognizes the following parameters:

- g *loginserver*** Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*** Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*** Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*** Specifies the password for the associated administrator account.
- n *subnet_address*** Subnet address for the DHCP servers that are to be viewed.
- o *organization*** Specifies the VitalQIP organization (corporation) name.
- f *filename*** Specifies the file where the information is stored. The default is *STDOUT*.
- all** Displays all primary and secondary servers. If this parameter is omitted, only primary servers are displayed.
- b *show_server_type*** Specifies if the server type should appear in the output. Allowable values are
 - y - show the server type
 - n - do not show the server type

Command line input example

```
qip-getdhcpsvr1st -u qipman -p password -b y -f c:\temp\output.txt
```

Output example

```
dhcp1.qtek.com,190.200.138.24,Lucent DHCP 5.4  
dhcp2.quadritek.com,190.201.123.01
```



qip-getdnsrr

qip-getdnsrr retrieves the resource record(s) associated with an object, domain, or reverse zone. The output is stored in the specified (or default) file.

Synopsis

```
qip-getdnsrr -t owner_type -n owner_name [-a address/mask
  [-g loginserver] [-s servername] [-u username] [-p password]
  [-o organization] [-f filename]
```

Parameters

qip-getdnsrr recognizes the following parameters:

- t *owner_type* Specifies the owner type of the resource records, for example, object, domain, or reverse zone.
- n *owner_name* Specifies either the fully-qualified domain name for which you want to retrieve resource records *or* IP address of the object or reverse zone for which you want to retrieve resource records.
- a *address/mask* Specifies either the fully-qualified domain name for which you want to retrieve resource records *or* IP address of the object or reverse zone for which you want to retrieve resource records.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getdnsrr -t object -u qipman -p passwd -n quadritek.com -a 10.98.2.10 -f rr.lst
```

Output example

```
ResourceRecOwner=hostxxx.qtek.com
ResourceRecType=A
ResourceClass=IN
```

```

MinimumTTL=23343
ResourceRecText=198.200.138.16
ApplyToZone=1
ResourceRecOwner=www
ResourceRecType=CNAME
ResourceClass=IN
MinimumTTL=1
ResourceRecText=hostxxx.qtek.com
ApplyToZone=1
ChangeFlag=
ExternalFlag=1
Tombstoned=1
ExternalComment="Comment"
ExternalTimestamp=3/16/2005 19:59:36

```

Table 12 Output field definitions

Field Name/Label	Description
Note: The following fields appear once per resource record.	
ResourceRecOwner	The owner of the resource record as defined in RFC1035.
ResourceRecType	The type of resource record (for example, A, PTR) as defined in RFC 1035.
ResourceClass	The resource record class (for example, IN, CS, CH, HS) as defined in RFC 1035.
MinimumTTL	Specifies the time interval (time to live) that the resource record can be cached before the source of the information should again be consulted. A zero value is interpreted to mean that the resource record can only be used for the transaction in progress, and should not be cached. For example, SOA records are always distributed with a zero TTL to prohibit caching. Zero values can also be used for extremely volatile data. If no TTL is desired (for example, it never expires), leave this field blank. Note: A value of -1 indicates no TTL has been specified.
ResourceRecText	The data area of the resource record as defined in RFC 1035.
ApplyToZone	The zone where the record should be added: 1 - forward zone 0 - reverse zone
ChangeFlag	Always returns a null value, so if a file generated by qip-getdnsrr is used as input to qip-setdnsrr , all records are skipped.
External Flag	Indicates how the record was updated 0 - Updated by VitalQIP 1 - External update
Tombstoned	0 indicates the object was not tombstoned. 1 indicates the object was tombstoned.
External Comment	A comment indicating the history of this external add.

Field Name/Label	Description
External Timestamp	Specifies the actual date and time.



qip-getdnssvrlst

qip-getdnssvrlst retrieves the entire list of existing fully-qualified DNS server(s) under the specified domain from the VitalQIP database. IP addresses are also listed once the server objects have been defined. The output is stored in the specified (or default) file.

Synopsis

```
qip-getdnssvrlst [-n domain_name|-a reverse_zone_address] [-g  
  loginserver] [-s servername] [-u username] [-p password]  
  [-o organization] [-f filename] [-b show_server_type]
```

Parameters

qip-getdnssvrlst recognizes the following parameters:

- | | |
|--------------------------------|---|
| -n <i>domain_name</i> | Specifies the fully-qualified domain name of the primary DNS server |
| -a <i>reverse_zone_address</i> | or the IP address of a reverse zone for the primary servers. |
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -f <i>filename</i> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |
| -b <i>show_server_type</i> | Specifies if the server type should appear in the output. Allowable values are
y - show the server type
n - do not show the server type |

Command line input examples

```
qip-getdnssvrlst -u qipman -p passwd -n quadritek.com -f c:\temp\output.txt
```

```
qip-getdnssvrlst -u qipman -p passwd -a 211.211.211.0
```

```
qip-getdnssvr1st -u qipman -p passwd
```

Output example

```
DNS1.quadritek.com,192.168.20.25  
DNS2.quadritek.com
```



qip-getdomnlst

qip-getdomnlst retrieves a domain list or an associated domain(s) for a particular subnet or folder from the VitalQIP database. If the subnet address is provided, only the domain(s) associated with that subnet or folder is retrieved. If the folder name is provided, the domains and subfolders associated with that folder are retrieved. The output is stored in the specified (or default) file.

Synopsis

```
qip-getdomnlst [-g loginserver] [-s servername] [-u username]  
               [-p password] [-o organization] [-a subnet_address] [-n folder_name]  
               [-f filename]
```

Parameters

qip-getdomnlst recognizes the following parameters:

- | | |
|---------------------------------------|--|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-a <i>subnet_address</i></code> | Specifies the subnet address for the domains for which you want to retrieve domain lists. |
| <code>-n <i>folder_name</i></code> | Specify the foldername.
Important! In the output for this CLI, the folders are displayed with a “f” following the name to differentiate them from domains (for example, foldername/subfolder,f). |
| <code>-f <i>filename</i></code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input examples

```
qip-getdomnlst -u qipman -p passwd -a 188.200.138.211 -f c:\temp\output.txt
```

```
qip-getdomn1st -u qipman -p passwd -f c:\output.txt
```

Output example

```
qip.quadritek.com  
qa.quadritek.com  
ra.qtek.com
```



qip-getfolderlst

The use of grouping domains into folders allows for easier, more efficient management of large numbers of domains in VitalQIP. Domain folders are shown in a hierarchical tree in a number of places in the GUI, if you have opted to use them (handled through a policy option for Administrators). **qip-getfolderlst** retrieves *all* existing domain folders in the specified organization, or only the top level folder in the hierarchy from the VitalQIP database. The output is stored in the specified (or default) file.

For more information on domain folders and their use, refer to Chapter 4 of the *VitalQIP User's Guide*.

Synopsis

```
qip-getfolderlst [-g loginserver] [-s servername] [-u username]
                 [-p password] [-o organization] [-f filename]
```

Parameters

qip-getfolderlst recognizes the following parameters:

- | | |
|------------------------------|---|
| <code>-g loginserver</code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s servername</code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u username</code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p password</code> | Specifies the password for the associated administrator account. |
| <code>-o organization</code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f filename</code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input example

- To retrieve all folders in the VitalQIP Organization and store it in *folder.lst*:

```
dip-getfolderlst -u qipman -p passwd -o "VitalQIP Organization" -f folder.lst
```

Output example

```
Folder1  
TopFolder/nextFolder  
TopFolder/nextFolder/bottomFolder
```



qip-getfreesubnetlst

qip-getfreesubnetlst retrieves all available existing or new subnets in the specified network address with a specified subnet mask from the VitalQIP database. The default is to retrieve the existing available subnets. The output is stored in the specified (or default) file.

Synopsis

```
qip-getfreesubnetlst [-g loginserver] [-s servername] [-u username]  
[-p password] [-o organization] -a network_address -m subnet_mask  
[-t existing|new] [-f filename]
```

Parameters

qip-getfreesubnetlst recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-a <i>network_address</i>	Specifies the IP address of the network from which you want to retrieve available subnets.
-m <i>subnet_mask</i>	Specifies the subnet mask to be applied.
-t <i>existing new</i>	Specifies whether to retrieve the available existing or new subnets. Type either e (or <i>existing</i>) or n (or <i>new</i>) to retrieve the available existing or new subnets. The default is "existing".
-f <i>filename</i>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .

Command line input example 1

- To retrieve all NEW free subnets from a specified network with a specified subnet mask:

```
qip-getfreesubnetlst -u qipman -p passwd -t new -a 144.144.0.0 -m 255.255.255.0
```

Output example 1

This output lists all the possible free subnets, and contains only the addresses because subnets have not been used/named.

```
144.144.234.0
144.144.235.0
144.144.236.0
```

Command line input example 2

- To retrieve all free subnets from a specified network with a specified subnet mask to a file named *subnet.txt*.

```
qip-getfreesubnetlst -u qipman -p passwd -a 144.144.0.0 -m 255.255.255.0 -f subnet.txt
```

Output example 2

```
"144.144.1.0" "sn144 1" "suborg 1"
"144.144.10.0" "sn144 10" "suborg 1"
```

Output field definition

In Command line input example 2, the `-f` parameter specifies *subnet.txt*. Therefore, the output is listed in the datafile *subnet.txt*. The fields are described further in Table 13:

Table 13 Subnet field definitions

Field	Description
Subnet Address	The free Subnet Address.
Subnet Name	The name of the free Subnet Address.
Subnet Organization	Subnet Organization that this free subnet belongs to.



qip-gethublst

qip-gethublst retrieves a list of all Wiring_HUBs of the specified subnet from the VitalQIP database for the subnet address specified. The output is stored in the specified (or default) file.

Synopsis

```
qip-gethublst -a subnet_address [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-gethublst recognizes the following parameters:

- a *subnet_address* Specifies the subnet address from which you want to retrieve a list of Wiring_HUBs.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-gethublst -u qipman -p passwd -o "VitalQIP Organization" -a 101.200.111.0 -f gethub.lst
```

Output example

```
Ny_hub.qtek.com  
Jersey_hub.qtek.com
```

qip-gethubport

qip-gethubport retrieves available Hub ports and numbers, and marks the port number as “selected”. The hub name must be a fully-qualified name. The output is stored in the specified (or default) file.

Synopsis

```
qip-gethubport -n hub_name -l slot_name [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
[-f filename]
```

Parameters

qip-gethubport recognizes the following parameters:

- n *hub_name* Specifies the fully-qualified wiring hub name for which you want to retrieve port numbers.
- l *slot_name* Specifies the wiring hub slot name for which you want to retrieve hub port numbers.
- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-gethubport -u qipman -p passwd -n hub1.qtek.com -l slotthree
```

Output example

HubPort=345
HubPort=113



qip-gethubslotlst

qip-gethubslotlst retrieves the entire slot list of the specified Wiring_HUB from the VitalQIP database. The hub name must be a fully-qualified name. The output is stored in the specified (or default) file.

Synopsis

```
qip-gethubslotlst -n hub_name [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f output_filename]
```

Parameters

qip-gethubslotlst recognizes the following parameters:

- n *hub_name* Specifies the fully-qualified Wiring_HUB name from which you want to retrieve Wiring_HUB slot information.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-gethubslotlst -u qipman -p passwd -n hub1.qtek.com -f hubslot.lst
```

Output example

```
"Slot1" 111
"Slot3" 333
```



qip-getipaddr

qip-getipaddr retrieves the next free IP address within a subnet, and marks the address as “selected”. The output is stored in the specified (or default) file.

Synopsis

```
qip-getipaddr -a subnet_address [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getipaddr recognizes the following parameters:

- a *subnet_address* Specifies the subnet address from which you want to retrieve a free address.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getipaddr -u qipman -p passwd -a 101.200.100.0
```

Output example

```
IPAddress=101.200.100.21
```



qip-getlocalsvrlist

qip-getlocalsvrlist retrieves the entire list of existing fully-qualified Local Host Servers from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getlocalsvrlist [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] [-f filename]
```

Parameters

qip-getlocalsvrlist recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getlocalsvrlist -u qipman -p passwd -f localsvr.lst
```

Output example

```
Local.qtek.com
LHost.qtek.com
```



qip-getloclst

qip-getloclst retrieves a list of all existing locations from the VitalQIP database; or (if the `-t` parameter is specified), it retrieves a list of locations that match the search criteria file. The output is stored in the specified (or default) output file.

Synopsis

```
qip-getloclst [-g loginserver] [-s servername] [-u username]  
              [-p password] [-o organization] [-f filename]  
              [-t search_tokens_filename]
```

Parameters

qip-getloclst recognizes the following parameters:

- | | |
|---|---|
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the <code>LOGIN</code> environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the <code>QIPDATASERVER</code> environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>filename</i></code> | Specifies the directory and filename for the output data. The default is <code>STDOUT</code> . |
| <code>-t <i>search_tokens_filename</i></code> | Specifies the name of the input file that contains the search criteria to match. If this file is present, the system returns a list of all locations that match the search criteria. Refer to below for sample search criteria. |

Specify the information on which you want to search for locations. If you do not specify search criteria, all locations are returned. For example, your `Search_Tokens_File` may consist of the following fields, as listed in Table 14:

Table 14 Search token file fields

Search_Tokens_File fields	Description
LocationID	Unique ID representing the Location.
Street1	The Street1 field of the location. Note that an implied “Begins with” search is performed. For example, if you enter 10 , all the streets that start with “10” are returned.
Street2	The Street2 field of the location. Note that an implied “Begins with” search is performed. For example, if you enter Su , all the streets that start with “Su” are returned.
City	The City of the location. Note that an implied “Begins with” search is performed. For example, if you enter Ma , all the names that start with “Ma” are returned.
State	The State of the location. Note that an implied “Begins with” search is performed. For example, if you enter P , all the names that start with “P” are returned.
Zip	Zip of the location. Note that an implied “Begins with” search is performed. For example, if you enter 19 , all the names that start with “19” are returned.
Country	Country of the location. Note that an implied “Begins with” search is performed. For example, if you enter US , all the names that start with “US” are returned.

The *location.lst* file, as shown in the Input and Output examples shown below, may contain:

```
Country=usa
```

Command line input example

```
qip-getloc1st -u qipman -p passwd -f location.lst
```

Output example

```
"9" "20" "Old Still Rd" "Woodbridge" "CONN" "02644" "usa"
"8" "5" "Whitney Ave" "New Haven" "Ct" "02488" "usa"
"2" "400" "Lapp Road" "Malvern" "PA" "19355" "usa"
"10" "1103" "Clover Hill Drive" "East Bradford" "PA" "19382" "usa"
"1" "20" "Lapp Road" "Malvern" "Penn" "19355" "usa"
"4" "5" "Great Valley Parkway" "Malvern" "Penn" "19355" "usa"
"3" "5" "Lapp Road" "Malvern" "Penn" "19355" "usa"
"6" "5" "Valley Road" "Christiana" "Penn" "19855" "usa"
```



qip-getmaclst

qip-getmaclst retrieves the entire list of manufacturers for a particular object class from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getmaclst -c object_class [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getmaclst recognizes the following parameters:

- c *object_class* Specifies the object class for which you want to retrieve the list of manufacturers. The available options are:
- | | |
|---------------|---------------------------------|
| - Workstation | - Terminal_Server |
| - X-terminal | - Switch |
| - PC | - Legacy_System |
| - Printer | - Gateway |
| - Server | - Test_Equipment |
| - Wiring_HUB | - Any user-defined object class |
| - Router | - Undefined |
| - Bridge | - Others |
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getmac1st -u qipman -p passwd -c PC -f mac.lst
```

Output example

```
Gateway-2000  
compac
```



qip-getmacmodellst

qip-getmacmodellst retrieves all model lists of the specified manufacturer from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getmacmodellst -m manufacturer -c object_class [-g loginserver]  
[-s servername] [-u username] [-p password] [-o organization]  
[-f filename]
```

Parameters

qip-getmacmodellst recognizes the following parameters:

- m *manufacturer* Specifies the manufacturer for which you want to retrieve model lists.

- c *object_class* Specifies the object class for which you want to retrieve the list of manufacturers. The available options are:
 - Workstation
 - X-terminal
 - PC
 - Printer
 - Server
 - Wiring_HUB
 - Router
 - Bridge
 - Terminal_Server
 - Switch
 - Legacy_System
 - Gateway
 - Test_Equipment
 - Any user-defined object class
 - Undefined
 - Others

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

`-f filename` Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getmacmodellst -u qipman -p passwd -c PC -m compaq -f modellist.txt
```

Output example

```
Compaq presario  
PrefixMACAddress=00008E  
Compaq prosignia  
PrefixMACAddress=00805F
```



qip-getmacpools

qip-getmacpools configures the global and subnet DHCP Server MAC pools, for both the inclusion and exclusion pools.

Synopsis

```
qip-getmacpools -a subnet_address | -n dhcp_server_name  
  [-g loginserver] [-s servername][-u username] [-p password]  
  [-o organization][-f filename]
```

Parameters

qip-getmacpools recognizes the following parameters:

- a *subnet_address* Specifies the subnet address where MAC Pools are being configured.

- n *dhcp_servername* Specifies the DHCP server name where MAC Pools are being configured.

- g *loginserver* Specifies the login server associated with the MAC Pools.

- s *servername* Specifies the server name where the MAC Pool is located.

- u *username* Specifies the username of the MAC Pool administrator.

- p *password* Specifies the password of the MAC Pool administrator

- o *organization* Specifies the organization with which the MAC Pool is associated.

- f *filename* Specifies the filename that contains the output from this CLI.

Output example 1

```
DHCP Server Name=testdhcp.qtek.com  
I, Ethernet, 000001243fa7  
E, IEEE802, 00000c1703ea  
I, Token Ring, 100*
```

Output example 2

```
Subnet Address=200.100.10.1  
I,Chaos,321321*  
E,Pronet,aabbccddeeff  
I,Ethernet,1233211233211234  
E,Token Ring,222*
```

Important! In the above output examples, I indicates that the MAC address is Included, E indicates that the MAC address is Excluded.



qip-getnetlst

qip-getnetlst retrieves a list of all networks from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getnetlst [-g loginserver] [-s servername] [-u username]  
              [-p password] [-o organization] [-f filename]
```

Parameters

qip-getnetlst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getnetlst -u qipman -p passwd -f netlst.ext
```

Output example

The first column in the output file contains the network address, the second column contains the network name.

```
"144.143.0.0"      "Malvern"  
"150.50.0.0"      "New Jersey"  
"198.200.138.0"   "Atlanta"
```



qip-getnissvrlst

qip-getnissvrlst retrieves a list of existing NIS servers for the specified organization from the VitalQIP database. If no organization is specified, a list of NIS servers for the VitalQIP organization is returned. The output is stored in the specified (or default) file.

Synopsis

```
qip-getnissvrlst [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] [-f filename]
```

Parameters

qip-getnissvrlst recognizes the following parameters:

<code>-g <i>loginserver</i></code>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
<code>-s <i>servername</i></code>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
<code>-u <i>username</i></code>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
<code>-p <i>password</i></code>	Specifies the password for the associated administrator account.
<code>-o <i>organization</i></code>	Specifies the VitalQIP organization (corporation) name.
<code>-f <i>filename</i></code>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .

Command line input example

```
qip-getnissvrlst -u qipman -p passwd -f nissvr.lst
```

Output example

```
nis1.qtek.com
nis2.qtek.com
```

□

qip-getnmdnsserver

qip-getnmdnsserver retrieves the attributes (profile) of a non-managed DNS server from the VitalQIP database. The fully-qualified QIP managed DNS server associated with this server must be specified. The output is stored in the specified (or default) file.

Important! When **-m** is used without **-a** and **-n** options, the attributes are returned for all non-managed servers that are associated with the server specified in **-m**. When **-n** and **-a** options are used, only the attributes for the non-manged server that is specified in **-n -a** are returned. If options **-n -a** are used, both options must be used together. When **-m, -n,** and **-a** options are used together, the same results are returned as are returned when using the **-n -a** options.

Synopsis

```
qip-getnmdnsserver -m qip_managed_svr_name [-n nm_svr_name  
-a nm_svr_ip] [-g loginserver] [-s servername] [-u username]  
[-p password] [-o organization] [-f filename]
```

Parameters

qip-getnmdnsserver recognizes the following parameters:

- | | |
|--|--|
| -m <i>qip_managed_svr_name</i> | Specifies the fully-qualified VitalQIP managed DNS server that is associated with the non-managed DNS servers. |
| -n <i>nm_svr_name</i>
-a <i>nm_svr_ip</i> | Specifies the fully-qualified non-managed DNS server name for the selected attributes when -n is used. Specifies the IP address of the non-managed DNS server for the selected attributes when -a is used. |
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |

`-f filename` Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getnmdnsserver -u qipman -p passwd -n dns1.qtek.com -a 100.100.100.100  
qip-getnmdnsserver -u qipman -p passwd -m qip.quadritek.com
```

Output example

Important! For a description of the fields in this output example and more information on non-managed DNS servers, refer to “Non-managed DNS servers” in Chapter 3 of the *VitalQIP User’s Guide*.

```
ServerName=DNS1.qtek.com  
NMServerName=dns1Sec.qtek.com  
NMServerAddress=150.11.22.33  
DomainName=pa.qtek.com  
ReverseZoneAddress=100.100.100.0/24
```



qip-getobjectlst

qip-getobjectlst retrieves a list of all objects (whether their status is “used” or “unused”) of the specified subnet from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getobjectlst -a subnet_address [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getobjectlst recognizes the following parameters:

- a *subnet_address* Specifies the subnet address from which you want to retrieve objects.

- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getobjectlst -u qipman -p passwd -a 199.201.139.0 -f objectlst.txt
```

Output example

For a description of the fields in the output example, refer to Table 15.

```
"10.98.4.10" "BHM010" "Workstation" "Static" "qtek.com"  
"10.98.4.11" "BHM020" "Workstation" "Static" "qtek.com"  
"10.98.4.12" "BHM030" "Workstation" "Static" "qtek.com"  
"10.98.4.13" "BHM040" "Workstation" "Static" "qtek.com"  
"10.98.4.14" "BHM050" "Undefined" "Static" "qtek.com"
```

Table 15 Object list field definitions

Field	Description
IP Address	IP Address of Object
Host name	Host Name of object
Object Class	Object Class of object: Workstation, X-terminal, PC, Printer, Server, Wiring_HUB, Router, Bridge, Terminal_Server, Switch, Legacy_System, Gateway, Test_Equipment, User-defined, Undefined, Others
Object Status	Static, Unused, Dynamic, M-DHCP, M-BOOTP, D-DHCP, A-DHCP, A-BOOTP
Domain	Default Domain of object, if assigned.



qip-getobjectprof

qip-getobjectprof retrieves the Object Profile of a specified object from the VitalQIP database. The `-a` parameter is required for the command to retrieve data for the object. Table 16 describes the possible field values. The output is stored in the specified (or default) file.

Synopsis

```
qip-getobjectprof -a object_address [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getobjectprof recognizes the following parameters:

- | | |
|---------------------------------------|---|
| <code>-a <i>object_address</i></code> | Specifies the object address from which you want to retrieve the Object Profile. |
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>filename</i></code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input example

```
qip-getobjectprof -a 199.200.139.27 -u qipman -p passwd -f object prof.txt
```

Output example

For a description of the fields in the output example, refer to Table 16, "Object profile field definitions", on page 110.

```
ObjectAddress=144.144.144.2  
SubnetAddress=144.144.144.0  
ObjectName=ws2
```

DomainName=quadritek.com
ObjectClass=PC
ExpiredDate=
ServerType=None
Application=
MACAddress=
ObjectTag=
LocationID=2
RoomID=
Street1=400
Street2=Lapp Road
City=Malvern
State=PA
Zip=19355
Country=usa
Manufacturer=
ModelType=
SerialNo=
AssetNo=
HostID=
PurchaseDate=
ObjectDescription=this is my pc
HubName=whp000002whs.quadritek.com
SlotName=hub-5
PortNum=1
ContactID=10
ContactLastName=newPerson
ContactFirstName=personF
ContactEmail=
ContactPhone=
ContactPager=
RouterGroup=
DynamicConfig=
TftpServer=
BootFileName=
HardwareType=
Aliases=Workstation-ws2 Wws2 aliasws
MailForwarders=WS1.quadritek.com(2)
MailHosts=ws3.quadritek.com(1)
HubSlots=
DNSServers=
TimeServers=
DefaultRouters=144.144.144.100
UserClasses=
Users=pa13
NameService=A, PTR
DynamicDNSUpdate=A, PTR, CNAME, MX
DHCPServer=
DHCPOptionTemplate=

```

DHCPPolicyTemplate=
LeaseTime=0
VendorClass=
TTLtime=100
DECNetArea=10
DECNetNode=1
DECNetAddress=AA0004000128
DualProtocol=DECNet
Tombstoned=1
ExternalComment="Comment"
ExternalTimestamp=03/09/2005 20:37:42
UserClass=MS Vendor
UsageBillServices=On
UsageBillLocation=malvern
UsageBillUserGroup=qtek
UsageBillObjectClass=Server
AllowDHCPClientsModifyDynamicObjectResourceRecords=True

```

Table 16 Object profile field definitions

Field name/label	Description		
ObjectAddress	The IP Address of the object.		
SubnetAddress	The subnet address of the object.		
ObjectName	The hostname of the object. Not fully-qualified.		
DomainName	The domain name of the object.		
ObjectClass	The object class of the object can be:		
	<table border="0"> <tr> <td style="vertical-align: top;"> <ul style="list-style-type: none"> - Workstation - X-terminal - PC - Printer - Server - Wiring_Hub - Router - Bridge </td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> - Terminal Server - Switch - Legacy System - Gateway - Test_Equipment - Any user-defined object class - Undefined - Others </td> </tr> </table>	<ul style="list-style-type: none"> - Workstation - X-terminal - PC - Printer - Server - Wiring_Hub - Router - Bridge 	<ul style="list-style-type: none"> - Terminal Server - Switch - Legacy System - Gateway - Test_Equipment - Any user-defined object class - Undefined - Others
<ul style="list-style-type: none"> - Workstation - X-terminal - PC - Printer - Server - Wiring_Hub - Router - Bridge 	<ul style="list-style-type: none"> - Terminal Server - Switch - Legacy System - Gateway - Test_Equipment - Any user-defined object class - Undefined - Others 		
ExpiredDate	The expiration date of the lease.		
ServerType	Server type. Only applicable if ObjectClass is "server": <ul style="list-style-type: none"> - TimeServer – if object is a time server. - TftpServer – if object is a Tftp server. - TftpServer&TimeServer – if object is a TftpServer and TimeServer 		
Application	The Primary Application of the object.		
MACAddress	The MAC Address of the object.		

Field name/label	Description
ObjectTag	Tag field of the object.
LocationID	The location ID field of the object.
RoomID	Room ID field of the object.
Street1	Street1 field of the location of the object.
Street2	Street2 field of the location of the object.
City	City of the location of the object.
State	State of the location of the object.
Zip	Zip of the location of the object.
Country	Country of the location of the object.
Manufacturer	Manufacturer Name of the object.
ModelType	Model Name of the object.
SerialNo	Serial Number of the object.
AssetNo	Asset Number of the object.
HostID	Host ID of the object.
PurchaseDate	Purchase Date/Time of the object. Must be in mm/dd/yyyy HH:MM format. (for example, 09/22/1998 00:00)
ObjectDescription	Object Description of the object.
HubName	Hub Name(s) assigned of the object.
SlotName	Slot Name(s) assigned of the object.
PortNum	Port Numbers assigned of the object.
ContactID	The contact ID field of the object.
ContactLastName	Contact last name of the object.
ContactFirstName	Contact first name of the object.
ContactEmail	Contact email address of the object.
ContactPhone	Contact phone number of the object.
ContactPager	Contact pager number of the object.
RouterGroup	Router Group of the object. Only applicable if the object class is "router".

Field name/label	Description
DECNetArea	DECNet Area of the object. (Valid values: 1-63) Only applicable if Dual Protocol set to DECNet. The DECNet Address is returned based on the DECNet Area and DECNet Node.
DECNetNode	DECNet Node of the object. (Valid values: 1-1024) Only applicable if Dual Protocol is set to DECNet. The DECNet Address is returned based on the DECNet Area and DECNet Node.
DECNetAddress	DECNet address of the object, which is derived from the object's DECNet Area. This field is for display only.
IPXNetworkNumber	IPX Network number of the object. Only applicable if Dual Protocol is set to IPX.
IPXNode	IPX Node of the object. Only applicable if Dual Protocol set to IPX.
DynamicConfig	Dynamic Configuration of the Object: <ul style="list-style-type: none"> - Blank – Static object - None – dynamic (none) - M-DHCP – Manual DHCP - A-DHCP – Automatic DHCP - D-DHCP – Dynamic DHCP - M-BOOTP – Manual Bootp - A-BOOTP – Automatic Bootp
TftpServer	TFTP Server of object. Only applicable if object type is M-BOOTP.
BootFileName	Bootfile name of object. Only applicable if object type is M-BOOTP.
HardwareType	Hardware Type of object. Only applicable for M-BOOTP objects: <ul style="list-style-type: none"> - Ethernet - TokenRing - AX.25 - Pronet - Chaos - IEEE802 - Arcnet
Aliases	Alias names. If multiple, separate by space (for example, www www3).
MailForwarders	Fully-qualified host name with priority, for example, mailF.qtek.com(10)
MailHosts	Fully-qualified host name with priority, for example, mailH.qtek.com(100)
Users	User login names. To delete a user, place a (d) after the login name.
UserClass	User class, if assigned. Only applicable for QDHCP server objects.
HubSlots	List of slot names (port numbers) if the object is a Wiring_HUB.
DNSServers	DNS Servers attached to this object. If multiple, separate by space. Only applicable for M-BOOTP objects.

Field name/label	Description
TimeServers	Time Servers attached to this object. If multiple, separate by space. Only applicable for M-BOOTP objects.
DefaultRouters	IP Address(es) of Default routers. If multiple, separate by space.
NameService	Name Services selection: No – Name Services checked off. Yes – Name Services checked on. A – Name Services checked on, A record checked on. PTR - Name Services checked on, PTR record checked on. A,PTR – Name Services checked on, A record checked on, PTR record checked on.
DynamicDNSUpdate	Dynamic Update selection: - A – Dynamic updates for A records. - PTR – Dynamic updates for PTR records. - CNAME – Dynamic updates for CNAME records. - MX – Dynamic updates for MX records. Important! Multiple selections can be accomplished by specifying each component, separated by comma. (for example, A,PTR,CNAME)
DHCPServer	The DHCP Server assigned to this object. Only applicable for dynamic objects.
DHCPTemplate	The DHCP Template assigned to this object. Only applicable for dynamic objects.
LeaseTime	The lease time (in seconds) assigned to this object. Only applicable for dynamic objects.
VendorClass	The Vendor Class assigned to this object. Only applicable for dynamic objects.
TTLtime	Time to live (TTL) value of the object, in seconds. Important! Since 0 is a valid value, a value of –1 indicates no TTL set.
NetbiosName	NetBIOS Name. Only applicable when dual protocol set to NETBIOS.
NetbiosDomain	NetBIOS Domain. Only applicable when dual protocol set to NETBIOS.
DualProtocol	Dual Protocol of object: - None - DECNet - IPX - NETBIOS
UsageBillServices	Determines whether the Usage Billing Service is on or off.
UsageBillLocation	Location name of the Usage Billing Service.
UsageBillUserGroup	User group of the Usage Billing Service.
UsageBillObjectClass	Object class the Usage Billing Service.

Field name/label	Description	
Tombstoned	0 indicates the object was not tombstoned. 1 indicates the object was tombstoned.	
External Comment	A comment indicating the history of this external add.	
External Timestamp	Specifies the actual date and time.	
AllowDHCPclients ModifyDynamicObject ResourceRecords	True False Same as in Global Policies	



qip-getobjname

qip-getobjname retrieves the next free default object name of the given object class. For more information on the default object name, refer to the “Naming Policies” section in Chapter 2 of the *VitalQIP User’s Guide*. The output is stored in the specified filename.

Synopsis

```
qip-getobjname -c object_class [-g loginserver] [-s servername]
                [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getobjname recognizes the following parameters:

- c *object_class* Specifies the object class for which you want to retrieve the list of manufacturers. The following options are available (note that the first letter must be a capital letter, for example enter Router not router):

- Workstation	- Terminal_Server
- X-terminal	- Switch
- PC	- Legacy_System
- Printer	- Gateway
- Server	- Test_Equipment
- Wiring_HUB	- Any user-defined object class
- Router	- Undefined
- Bridge	- Others

- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getobjname -u qipman -p passwd -c Workstation
```

Output example

```
ObjectName=wsp000010wss
```



qip-getorganization

qip-getorganization retrieves the description of the organization, the maximum object count, and the number of objects that currently exist when the `-o` parameter is specified. When `-o` is specified, only one organization is retrieved. The output is stored in the specified (or default) file.

Synopsis

```
qip-getorganization [-g loginserver] [-s servername] [-u username]
  [-p password] [-f filename] [-i] [-o organization]
```

Parameters

qip-getorganization recognizes the following parameters:

- `-g loginserver` Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- `-s servername` Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- `-u username` Specifies the VitalQIP administrator account to be used in establishing the database connection.
- `-p password` Specifies the password for the associated administrator account.
- `-f filename` Specifies the file where the information is stored. The default is *STDOUT*.
- `-i` Displays all organization IDs. The ID is displayed in the form `<orgid>`, `<orgname>`.
- `-o organization` Specifies the VitalQIP organization (corporation) name. If the `-o` parameter is used, specific information about the organization is returned. If the `-o` parameter is not specified a list of organizations is returned, as shown in the output example below.

Command line input example

```
qip-getorganization -u qipman -p passwd -i
```

Output example

```
2, New Organization
1, VitalQIP Organization
```



qip-getospflst

qip-getospflst retrieves a list of all OSPF areas from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getospflst [-g loginserver] [-s servername] [-u username]  
               [-p password] [-o organization] [-f filename]
```

Parameters

qip-getospflst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getospflst -u qipman -p passwd -f ospflst.txt
```

Output example

```
BACKBONE  
new_ospf
```



qip-getospfprof

qip-getospfprof retrieves the profile of an OSPF area from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getospfprof -n ospf_area_name [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getospfprof recognizes the following parameters:

- n *ospf_area_name* Specifies the name of the OSPF area for which you are retrieving a profile.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getospfprof -u qipman -p passwd -n new_ospf -f ospf1st.txt
```

Output example

```
OspfAreaName=rew_ospf
OspfAreaID=0.1.226.64
WarningPercent=0
WarningType=0
SubnetAddress=180.200.0.1
SubnetName=subnet_name
NetworkAddress=180.200.0.0
NetworkName=network_180_200_new_name
SubnetOrgName=rew_subnet_org
```

OspfName=rew_ospf
SubnetAddress=180.200.0.1
SubnetName=subnet_name
NetworkAddress=180.200.0.0
NetworkName=network_180_200_new_name
SubnetOrgName=rew_subnet_org



qip-getpolicy

qip-getpolicy shows policy information on a policy for the enterprise server. For more information on policies, refer to Chapter 2, “Establish Policies and Profiles”, in the *VitalQIP User’s Guide*.

Synopsis

```
qip-getpolicy [-c class] [-n policyName] [-g loginserver]
              [-s servername] [-u username] [-p password] [-o organization]
              [-f filename]
```

Parameters

qip-getpolicy recognizes the following parameters:

- | | |
|------------------------------|--|
| <code>-c class</code> | Specifies the class of policy you want to show information on (for example, BILLING). If you do not specify a class, all classes appear. |
| <code>-n policy</code> | Name Specifies the policy you want to show information on (for example, PING_ATTEMPTS and ALWAYS_APPEND_ROUTER). If you do not specify a policy, you see all policies. |
| <code>-g loginserver</code> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s servername</code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u username</code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p password</code> | Specifies the password for the associated administrator account. |
| <code>-o organization</code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f filename</code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Command line input examples

- To show policy information for all classes:
`qip-getpolicy -s QIPSYBASE -U qipman -P passwd`
- To show policy information for all policies with the Billingclass:
`qip-getpolicy -s QIPSYBASE -u qipman -p passwd -c Billing`
- To show specific policy information for the user class policy in the Billing class:

```
qip-getpolicy -s QIPSYBASE -u qipman -p passwd -c Billing -n Required_Data
```

Output example

```
Class=Billing  
Policy=Required_Data, Value=USER GROUP
```



qip-getprimdnssvrlst

qip-getprimdnssvrlst makes a list of the primary DNS servers specified in the VitalQIP database based on domain name or the IP address of the reverse zone. The output is stored in the specified (or default) file.

Synopsis

```
qip-getprimdnssvrlst [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] [-f filename]
  -n domain_name|-a in-addr_IPaddress
```

Parameters

qip-getprimdnssvrlst recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-f <i>filename</i>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .
-n <i>domain_name</i> -a <i>in-addr_IPaddress</i>	Specifies the fully-qualified domain name of the primary DNS servers or the IP address of the reverse zone for which the servers are the primary.

Command line input example 1

- To get the list of primary DNS servers for the domain *example.com*:

```
qip-getprimdnssvrlst -n example.com
```

Output example 1

```
primdnsserver.example.com
```

Command line input example 2

- To get the list of primary DNS servers for the reverse zone 10.0.0.0:
`qip-getprimdnssvr1st -a 10.0.0.0`

Output example 2

`primdnserver.example.com`



qip-getrevzonest

qip-getrevzonest retrieves a list of reverse zones for an organization from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getrevzonest [-g loginserver] [-s servername]
                 [-o organization] [-u username] [-p password] [-f filename]
```

Parameters

qip-getrevzonest recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getrevzonest -u qipman -p passwd -f revzonest.txt
```

Output example

```
10.in-addr.arpa,10.0.0.0/8
0.0.10.in-addr.arpa,10.0.0.0/24
1.0.10.in-addr.arpa,10.0.1.0/24
2.0.10.in-addr.arpa,10.0.2.0/24
3.0.10.in-addr.arpa,10.0.3.0/24
4.0.10.in-addr.arpa,10.0.4.0/24
5.0.10.in-addr.arpa,10.0.5.0/24
0.0.11.in-addr.arpa,11.0.0.0/24
123.in-addr.arpa,123.0.0.0/8
1.1.192.in-addr.arpa,192.1.1.0/24
1.1.194.in-addr.arpa,194.1.1.0/24
138.200.198.in-addr.arpa,198.200.138.0/24
```

0-27.138.200.198.in-addr.arpa,198.200.138.0/27
64-27.138.200.198.in-addr.arpa,198.200.138.64/27



qip-gettrlst

qip-gettrlst retrieves a list of all routers in the specific subnet from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-gettrlst -a subnet_address [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-gettrlst recognizes the following parameters:

- a *subnet_address* Specifies the subnet address from which you want to retrieve routers.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-gettrlst -u qipman -p passwd -a 199.200.139.0 -f rtrlst.txt
```

Output example

```
199.200.139.1
199.200.139.161
```



qip-getsecdnssvrlst

qip-getsecdnssvrlst returns a list of fully-qualified secondary DNS server names based on the domain name or the IP address of the reverse zone. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsecdnssvrlst [-g loginserver] [-s servername] [-u username]  
[-p password] [-o organization] [-f filename]  
-n domain_name|-a in-addr_IPaddress
```

Parameters

qip-getsecdnssvrlst recognizes the following parameters:

- | | |
|--|--|
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -f <i>filename</i> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |
| -n <i>domain_name</i> -a <i>in-addr_IPaddress</i> | Specifies the fully-qualified domain name of the primary DNS servers or requests the IP address of the reverse zone for which the DNS servers are secondary. |

Command line input example

```
qip-getsecdnssvrlst -u qipman -p passwd -a 200.200.200.0 -f sdnssvrlst.txt
```

Output example

```
dns2.qtek.com  
dns3.qtek.com
```



qip-getsnaddr

qip-getsnaddr gets the subnet address, given the IP address of an object. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsnaddr -a object_address [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getsnaddr recognizes the following parameters:

- a *object_address* Specifies the IP address of an object.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getsnaddr -u qipman -p passwd -a 199.200.138.27 -f snaddr.txt
```

Output example

```
SubnetAddress=199.200.138.0
```



qip-getsnoorglst

qip-getsnoorglst retrieves a list of all subnet organizations from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsnoorglst [-g loginserver] [-s servername] [-u username]  
                 [-p password] [-o organization] [-f filename]
```

Parameters

qip-getsnoorglst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getsnoorglst -u qipman -p passwd -f subgrplst.txt
```

Output example

```
"qtek_suborg1"  
"new_subnet_org"
```



qip-getsnorgprof

qip-getsnorgprof returns the profile of a subnet organization. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsnorgprof [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] -n subnet_organization_name
  [-f filename]
```

Parameters

qip-getsnorgprof recognizes the following parameters:

-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-n <i>subnet_organization_name</i>	Specifies the name of the subnet organization.
-f <i>filename</i>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .

Command line input example

```
qip-getsnorgprof -u qipman -p passwd -n psl_suborg1 -f out.dat
```

Output example

For a description of the fields in the output example, refer to Table 17.

```
SubnetOrgName=snorg1
WarningPercent=95
WarningType=1
AssignGap=Workstation,1
AssignGap=PC,2
```

```

SubnetAddress(net,mask,addr)=144.144.0.0,255.255.255.0,144.144.1.0
SubnetAddress(net,mask,addr)=144.144.0.0,255.255.255.0,144.144.2.0
W2KSiteName=site1
UseSubnetOrgName=false
AssocControllers=dcl.lucent.com
DhcpServer=dhcp1.lucent.com
DhcpOptionTemplate=general

```

Table 17 Subnet organization profile field definitions

Field name/label	Description
SubnetOrgName	The Subnet Organization name.
WarningPercent	The Warning Managed Addresses full %. (for example, 10 equals 10 percent)
WarningType	Warning Type: 0 – no warning 1 – email only 2 – audible only 3 – both email and audible warning
AssignGap	The object class of the GAP object defined. The relative offset. Note that a positive number (>0) indicates “from the beginning” of the subnet. A negative number indicates “from the end” of the subnet.
SubnetAddress(net,mask,addr)	Specifies the subnet address, network address, and subnet mask.
W2KSiteName	The site name of the Windows 2000 site. If blank, the subnet organization name is used. (Optional)
AssocControllers	The list of Windows 2000 controllers.
DHCPServer	The default DHCP server for this subnet.
DHCPOptionTemplate	The default DHCP Option Template for this subnet.
UseSubnetOrgName	Specifies if the subnet organization name is used.



qip-getsubnetlst

qip-getsubnetlst retrieves a list of all subnets the specified domain ([name]) manages from the VitalQIP database. If name is provided, the list includes all subnets under that domain only. Otherwise, it includes all subnets in VitalQIP. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsubnetlst [-g loginserver] [-s servername] [-u username]
  [-p password] [-o organization] -t owner_type -n name|-a address
  [-f filename]
```

Parameters

qip-getsubnetlst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- t *owner_type* Specifies the owner type under which subnets belong. Refer to Table 18 following.

- n *name*|-a *address* Specifies the IP Address or fully-qualified domain, subnet organization, OSPF, or network name from where you want to retrieve subnets or the network address from where you want to retrieve subnets.

- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Table 18 Owner types

Owner type	Notes
Corp	Displays subnets for an Organization. Use -n to specify the organization name.
Subnetorg	Displays subnets for a Subnet Organization. Use -n to specify the subnet organization name.
Domain	Displays subnets for a domain. Use -n to specify the domain name.
OSPF	Displays subnets for an OSPF Area. Use -n to specify the OSPF area name.
Network	Displays subnets for a network. Use -a to specify the network address.

Command line input example

```
qip-getsubnetlst -u qipman -p passwd -t corp -f subnetlst.txt
```

Output example

For a description of the fields in the output example, refer to Table 19.

```
"sn1" "150.50.0.0" "N" "Network A" "150.50.0.0" "Building 1" "os1"
```

Table 19 Subnet list field definitions

Field name/label	Description
Subnet Name	The Subnet Name of the subnet.
Subnet Address	The Subnet Address.
Usage Flag	Usage Flag: Y–Used N–Not Used S–Scheduled Move P–Planned Move
Network Name	Network name.
Network Address	Network Address.
Subnet Organization	Subnet Organization to which this subnet belongs.
OSPF Area name	OSPF Area to which this subnet belongs.



qip-getsubnetprof

qip-getsubnetprof retrieves the Subnet Profile of a specific subnet from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getsubnetprof -a subnet_address [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-f filename]
```

Parameters

qip-getsubnetprof recognizes the following parameters:

- a *subnet_address* Specifies the subnet address from where you want to retrieve a Subnet Profile.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getsubnetprof -u qipman -p passwd -a 199.200.139.0 -f subnet_prof.txt
```

Output example

For a description of the fields in the output example, refer to Table 20.

```
SubnetAddress=144.144.144.0
SubnetName=SN144_144
SubnetMask=255.255.255.0
NetworkAddress=144.144.0.0
LocationID=2
Street1=400
```

Street2=Lapp Road
City=Malvern
State=PA
Zip=19355
Country=usa
ContactID=2
ContactLastName=pal
ContactFirstName=pal
ContactEmail=pal@qtek.com
ContactPhone=111-0033
ContactPager=1-800-111-3300
Application=
Domain=qtek.com
TftpServer=
ShowUsage=N
CheckUsage=N
SubnetDescription=
SharedNetwork=
HardwareType=
WarningType=2
WarningPercentage=60
DNSServers=dns1.qtek.com
TimeServers=pal.qtek.com
DefaultRouters=144.144.144.64 144.144.144.100 144.144.144.101
DHCPServer=pal.qtek.com
DHCPOptionTemplate=
DHCPPolicyTemplate=
PrimaryInterface=No
UsageBillLocation=malvern
UsageBillUserGroup=qtek
AllowDHCPClientsModifyDynamicObjectResourceRecords=True

Table 20 Subnet profile field definitions

Field	Description
SubnetAddress	The subnet Address.
SubnetName	The subnet name.
SubnetMask	The subnet mask.
NetworkAddress	The network address.
LocationID	The Location ID of the Subnet.
Street1	Street1 field of the location.
Street2	Street2 field of the location.
City	City of the location.
State	State of the location.
Zip	Zip of the location.
Country	Country of the location.
ContactID	The Contact ID of the subnet.
ContactLastName	Contact last name.
ContactFirstName	Contact first name.
ContactEmail	Contact email address.
ContactPhone	Contact phone number.
ContactPager	Contact pager number.
Application	Primary Application assigned to this subnet.
Domain	Domain assigned separated by spaces.
TftpServer	Default Tftp server assigned.
ShowUsage	Show Used Only Flag: 0 = No 1 = Yes
CheckUsage	Check before assign: Y=Ping N=None
SubnetDescription	Subnet “comment” text area.
SharedNetwork	List of the shared networks for this subnet.

Field	Description
HardwareType	Default hardware type: Ethernet TokenRing AX.25 Pronet Chaos -IEEE802 -Arcnet
WarningType	The Warning Type is issued with one of the numeric values: 0 = No warning is set 1 = Warning is set to E-mail 2 = Warning is set to Visual 3 = Warning is set to E-mail and Visual
WarningPercentage	Indicates the percentage of objects in service for a subnet at which a warning is issued.
DNSServers	DNS Servers for this subnet, separated by spaces.
TimeServers	Time Servers for this subnet, separated by spaces.
DefaultRouters	Default Routers for this subnet, separated by spaces.
DHCPsServer	The default DHCP server for this subnet.
DHCPOptionTemplate	The default DHCP Option Template for this subnet.
DHCPPolicyTemplate	The DHCP Policy Template for this subnet.
PrimaryInterface	Primary Interface – Yes / No
UsageBillLocation	Location of the Usage Billing Service.
UsageBillUserGroup	User group of the Usage Billing Service.
AllowDHCPclients ModifyDynamicObjectResourceRecords	True False Same as in Global Policies



qip-gettemplate

qip-gettemplate retrieves a specified DHCP template(s) and writes it to the specified (or default) output file.

Synopsis

```
qip-gettemplate -n template_name [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
[-f filename] -t option|policy_subnet|policy_clientclass|
policy_scope|policy_server
```

Parameters

qip-gettemplate recognizes the following parameters:

- | | |
|---|---|
| <code>-n <i>template_name</i></code> | Specifies the name of the DHCP template you want to retrieve. |
| <code>-g <i>loginserver</i></code> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| <code>-u <i>username</i></code> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| <code>-p <i>password</i></code> | Specifies the password for the associated administrator account. |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name. |
| <code>-f <i>filename</i></code> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |
| <code>-t <i>option </i>
<i>policy_subnet </i>
<i>policy_clientclass </i>
<i>policy_scope </i>
<i>policy_server</i></code> | <p>This parameter is used in conjunction with the <code>-n</code> parameter to specify the type of template for which to search. Valid options or values are:</p> <ul style="list-style-type: none"> • <code>option</code> - searches for a specific option template. • <code>policy_subnet</code> - searches for a specific subnet policy template. • <code>policy_clientclass</code> - searches for a specific client class policy template. • <code>policy_scope</code> - searches for a specific scope policy template. • <code>policy_server</code> - searches for a specific server policy template. |

Command line input example

```
qip-gettemplate -n dhcp_tempplate1 -s fred -u wilman -p wilma -f tpl.dat
```

Output example

For information on option descriptions, refer to the tables in “Defining DHCP/Bootp Template Options” section in Chapter 2 of the *VitalQIP User’s Guide*.

```
Template-name=MS_DHCP_Server_Template  
template-type=DHCP Option Template  
002=64157  
003=198.200.138.234  
013=31588  
015=quadritek.com  
019=33  
043=54321  
template-name=10.200.80.0_Template  
template-type=SUBNET  
003=198.200.138.0  
051=120
```



qip-gettemplst

qip-gettemplst retrieves a list of all DHCP templates from the VitalQIP database. The output is stored in the file name specified.

Synopsis

```
qip-gettemplst [-g loginserver] [-s servername] [-u username]
               [-p password] [-o organization] [-f filename] [-t
               option|policy_subnet|policy_clientclass|
               policy_scope|policy_server]
```

Parameters

qip-gettemplst recognizes the following parameters:

- | | |
|---|---|
| -g <i>loginserver</i> | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -f <i>filename</i> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |
| -t <i>option</i>
<i>policy_subnet</i>
<i>policy_clientclass</i>
<i>policy_scope</i>
<i>policy_server</i> | This parameter is used to specify the type of template for which to search. Valid options or values are: <ul style="list-style-type: none"> • <i>option</i> - searches for option templates • <i>policy_subnet</i> - searches for subnet policy templates • <i>policy_clientclass</i> - searches for client class policy templates • <i>policy_scope</i> - searches for scope policy templates • <i>policy_server</i> - searches for a server policy templates |

Command line input example

```
qip-gettemplst -u qipman -p passwd -f templates.txt
```

Output example

“general”

“microsoft_clients”



qip-gettimesvrlst

qip-gettimesvrlst retrieves a list of all time servers from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-gettimesvrlst [-g loginserver] [-s servername] [-u username]
                  [-p password] [-o organization] [-f filename]
```

Parameters

qip-gettimesvrlst recognizes the following parameters:

- g *loginserver*** Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*** Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*** Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*** Specifies the password for the associated administrator account.
- o *organization*** Specifies the VitalQIP organization (corporation) name.
- f *filename*** Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-gettimesvrlst -u qipman -p passwd -f timesvrs.txt
```

Output example

```
TimeSvr.qtek.com
TimeSvr2.qipman.com
```



qip-getudflst

qip-getudflst retrieves a list of all user-defined field names or values from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getudflst -t fields|values -c class_name [-a address|-n name]  
[-g loginserver] [-s servername] [-u username] [-p password]  
[-o organization] [-f filename]
```

Parameters

qip-getudflst recognizes the following parameters:

- | | |
|---------------------------------------|---|
| -t
fields values | Specifies either “fields” or “values” to retrieve a list of user-defined field names or user-defined field values. |
| -c <i>class_name</i> | Specifies the user-defined fields class name (type) associated with the user-defined field name. Refer to Table 21 for class name types and definitions. |
| -a <i>address</i>
-n <i>name</i> | Specifies the IP address of the class object, subnet, reverse zone, or the fully qualified object name with which the value is associated. The fully qualified object name is applicable when referring to the owner of the domain, organization, user, or object. When referring to the user, the login ID assigned to the user is applicable.

You can only use the -a parameter if the -t parameter is set to values.
You can only use the -n parameter if the -t parameter is set to fields. |
| -g <i>loginserver</i> | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i> | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable. |
| -u <i>username</i> | Specifies the VitalQIP administrator account to be used in establishing the database connection. |
| -p <i>password</i> | Specifies the password for the associated administrator account. |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name. |
| -f <i>filename</i> | Specifies the file where the information is stored. The default is <i>STDOUT</i> . |

Table 21 UDF class name types

UDF_Class_Name	Description
User	Used for user-defined fields for a user.
Domain	Used for user-defined fields for a domain.
ReverseZone	Used for user-defined fields for a reverse zone.
Organization	Used for user-defined fields for an organization.
Object	Used for user-defined fields for an object profile.
Subnet	Used for user-defined fields for a subnet profile.
Prov_block	Used for user-defined fields for a Network Allocator block.
Prov_Pool	Used for user-defined fields for a Network Allocator pool.

Command line input example 1

- To retrieve ALL user-defined fields of the type “User”:
qip-getudflst -u qipman -p passwd -t field -c User

Output example 1

“USER1”

Command line input example 2

- To retrieve ALL user-defined field values of the type “User”, for the user “jsmith”:
qip-getudflst -t value -c User -n jsmith

Output example 2

“USER1” “jsmith”

Command line input example 3

- To retrieve ALL user-defined fields of the type “Organization”:
qip-getudflst -t field -c Organization

Output example 3

“ORG1”
“ORG2”
“ORG3”
“ORG4”

Command line input example 4

- To retrieve ALL user-defined field values of the type “Organization” with the name “VitalQIP Organization”:

```
qip-getudflst -t value -c Organization -n "VitalQIP Organization"
```

Output example 4

```
"ORG1" "QA"  
"ORG2" "DEV"  
"ORG3" "MKTG"  
"ORG4" "QIP Org4"
```

Command line input example 5

- To retrieve ALL user-defined fields of the type "ReverseZone" for "Sales":

```
qip-getudflst -t field -c ReverseZone -o "Sales" -n reversezonels1
```

Output example 5

```
"REVERSEZONE1"
```

Command line input example 6

- To retrieve user-defined field values of the type "Reverse Zone" for a specific subnet, use the **-a** parameter and a network address/mask value:

```
qip-getudflst -t value -c ReverseZone -o "Mktg Organization" -a 10.0.0.0/8
```

Output example 6

```
"REVERSEZONE1" "SALESREVERSEZONE"
```

Command line input example 7

- To retrieve user-defined field names of the type "Domain":

```
qip-getudflst -t field -c Domain
```

Output example 7

```
D1  
D2  
D3
```

Command line input example 8

- To retrieve user-defined field values for a *specific* domain use the **-n** option:

```
qip-getudflst -t field -c Domain -n seg1.qa.quadritek.com
```

Output example 8

```
"D1" "Seg1"  
"D2" "qa"  
"D3" "quadritek"  
"D4" "com"
```



qip-getudfnamelst

qip-getudfnamelst retrieves a list of all user-defined field names from the VitalQIP database. The output is stored in the specified (or default) file.

Synopsis

```
qip-getudfnamelst [-g loginserver] [-s servername] [-u username]
                  [-p password] [-o organization] [-f filename]
```

Parameters

qip-getudfnamelst recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getudfnamelst -u qipman -p passwd -f nameLst
```

Output example

```
PSL-UDF, Object
SN-UDF, Subnet
DOMN-UDF, Domain
ORT-UDF, Organization
USER1, User
REVZ-UDF, ReverseZone
```



qip-getuser

qip-getuser retrieves VitalQIP User information. The output is stored in the specified (or default) file.

Synopsis

```
qip-getuser [-g loginserver] [-s servername] [-u username]  
            [-p password] [-o organization] -l login_name [-f filename]
```

Parameters

qip-getuser recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- l *login_name* Specifies the login ID of the VitalQIP user whose information is being retrieved.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getuser -u qipman -p password -l pal3
```

Output example

Important! For a description of the fields in this output example and more information on users, refer to “Define VitalQIP Users” section in Chapter 4 of the *VitalQIP User's Guide*.

```
LoginID=pal3  
LastName=test  
FirstName=pal3  
Phone=292-1312  
EmailAddress=pal3@qtek.com
```

```
Password=*****  
Pin=  
Description=another user with different id  
Street1=5  
Street2=Whitney Ave  
City=New Haven  
State=Ct  
Zip=02488  
Country=usa  
ActivationStatus=1  
GroupInformation="isp_qtek2","isp_qtek"  
DefaultSubnets=144.144.144.0  
ManagedRange=144.144.144.3,144.144.144.2  
UserDefinedFields="USER1: user1 udf value for pal3"  
UsageBillLocation=malvern  
UsageBillUserGroup=qtek
```



qip-getuseraddr1st

qip-getuseraddr1st retrieves a list of IP objects associated with the user. It retrieves the IP address, MAC Address, hostname, dynamic type, lease grant, and expiration values. The output is stored in the specified (or default) file.

Synopsis

```
qip-getuseraddr1st [-g loginserver] [-s servername] [-u username]  
[-p password] [-o organization] -l login_name [-f filename]
```

Parameters

qip-getuseraddr1st recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- l *login_name* Specifies the login name of the user to retrieve the address list for.
- f *filename* Specifies the file where the information is stored. The default is *STDOUT*.

Command line input example

```
qip-getuseraddr1st -u qipman -p password -l testuser -f usr_list.dat
```

Output example

The format of the file is as follows:

IP_address, hostname, MAC_address, dynamic_configuration, lease_granted, lease_expires

```
198.200.139.11, testuserpc, 223344556677, Dynamic_DHCP,,  
198.200.138.12, Device1, 112233445566, Manual_DHCP, 10/15/1999 09:00,
```

□

qip-getzoneext

qip-getzoneext retrieves the contents of the zone extension associated with the domain (-n parameter), or DNS server extensions provided in the -d parameter, or the reverse zone provided in the -a parameter. (The -d, -n, and -a parameters are mutually exclusive – only one can be used at a time.)

Synopsis

```
qip-getzoneext -n domain_name|-a reverse_zone_address|maskLen|-d
  dns_server_fqdn [-g loginserver] [-s dataserver] [-u username]
  [-p password] [-o organization] [-f output_filename] [-i input_file]
```

Parameters

qip-getzoneext recognizes the following parameters:

-n <i>domain_name</i>	Specifies the fully qualified domain name (-n) for which you are retrieving a profile, or the IP address (-a) of the reverse zone and mask for which you are retrieving a profile, or the DNS Server name (-d) for which you want to retrieve extensions.
-a <i>zone_address/maskLen</i>	
-d <i>dns_server_fqdn</i>	
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>dataserver</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-f <i>filename</i>	Specifies the filename of the output file.
-i <i>input_file</i>	Specifies the directory and filename where the input data is to be placed.

Command line input examples

- To get the Domain Extensions, execute the following:

```
qip-getzoneext -u qipman -p passwd -n qtek.com -f extfile.txt
```

- To get the DNS Server Extensions, execute the following:
qip-getzoneext -u qipman -p passwd -d dnserver1.qtek.com -f extfile.txt
- To get the Reverse Zone Extensions, execute the following:
qip-getzoneext -u qipman -p passwd -a 144.44.44.0/24 -f extfile.txt

Output examples

For a description of the fields in this output example and information on Domain Extensions, refer to “Define Domains” in Chapter 3 of the *VitalQIP User’s Guide*.

- Output for Domain Extensions:

```

DomainName=qtek.com
OptionType=Extensions
ParameterName=Prefix of zone db file
STTL 6400
ParameterName=Postfix of zone db file
candybar IN A 19.20.21.22
snickers IN CNAME candybar.quadritek.com
candybar IN HINFO "VAX" "UNIX"
quadritek.com. IN MX 10 candybar.quadritek.com.
OptionType=BIND-8.X Options
ParameterName=zone block of named.conf
// extra zone options for BIND 8.X server
qddns {
wins-servers {1,2,3,4:1.2.3.5;}
wins-ttl 0;
};
OptionType=BIND-9.X Options
ParameterName=zone block of named.conf
// extra zone options for BIND 9.X server
forward only;
zone-statistics yes;
OptionType=LUCENT DNS 3.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 3.X server
// ...
OptionType=LUCENT DNS 4.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 4.X Server
// ...
OptionType=WINDOWS 2000 DNS Options
ParameterName=zone-options
// extra zone options for WINDOWS 2000 DNS Server
// ...

```

- Output for DNS Server Extensions:

```

ServerName=dnserver.qtek.com
ParameterName= Corporate Extension

```

```
include "allowquery.conf"
include "blackhoe.conf"
ParameterName=db.cache file extension
Include "test.conf"
```

- **Output for Reverse Zone Prefix and Postfix Extensions:**

```
ReverseZone=170.200.0.0/16
DomainName=qtek.com
OptionType=Extensions
ParameterName=Prefix of zone db file
STTL 4600
ParameterName=Postfix of zone db file
68 IN PTR madeup.quadritek.com.
69 IN PTR madeup2.quadritek.com.
70 IN PTR madeup3.quadritek.com.
71 IN PTR madeup4.quadritek.com.
OptionType=BIND-8.X Options
ParameterName=zone block of named.conf
qddns {
wins-servers {1,2,3,4:1.2.3.5;}
wins-ttl 0;
};
OptionType=BIND-9.X Options
ParameterName=zone block of named.conf
// extra zone options for BIND 9.X server
forward only;
zone-statistics yes;
OptionType=LUCENT DNS 3.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 3.X server
// ...
OptionType=LUCENT DNS 4.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 4.X Server
// ...
OptionType=WINDOWS 2000 DNS Options
ParameterName=zone-options
// extra zone options for WINDOWS 2000 DNS Server
// ...
```

□

qip-getzoneprof

qip-getzoneprof retrieves the profile of a specified domain or reverse zone. The output is stored in the specified (or default) file.

Synopsis

```
qip-getzoneprof -n zone_name|-a zone_address/maskLen [-g loginserver]  
[-s servername] [-u username] [-p password] [-o organization]  
[-f filename]
```

Parameters

qip-getzoneprof recognizes the following parameters:

-n <i>zone_name</i>	Specifies the fully-qualified domain name (-n) for which you are retrieving a profile, or the IP address (-a) of the reverse zone and mask for which you are retrieving a profile.
-a <i>zone_address/maskLen</i>	
-g <i>loginserver</i>	Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
-s <i>servername</i>	Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
-u <i>username</i>	Specifies the VitalQIP administrator account to be used in establishing the database connection.
-p <i>password</i>	Specifies the password for the associated administrator account.
-o <i>organization</i>	Specifies the VitalQIP organization (corporation) name.
-f <i>filename</i>	Specifies the file where the information is stored. The default is <i>STDOUT</i> .

Command line input example 1

```
qip-getzoneprof -n rob.com -u qipman -p passwd -f domainprof.txt
```

Output example 1

For a description of the fields in the Output examples, refer to the section "Define Domains" in Chapter 3 of the *VitalQIP User's Guide*.

```
Zone=rob.com  
dnsServers=dns1.rob.com P 0,dns2.rob1.com S 0,dns2k.rob2k.com S 0  
RefreshTime=21600
```

```

ExpirationTime=604800
RetryPeriod=3600
MinimumTTL=86400
NegativeCacheTTL=86400
ZoneMail=rob@rob.com
Bind8AllowQuery=Use List
Bind8AllowTransfer=Use List
Bind8AllowUpdate=Use List
Bind8CheckNames=Fail
Bind8Notify=Use Server Value
Bind9AllowNotify=Use List
Bind9AllowQuery=Use List
Bind9AllowTransfer=None
Bind9AllowUpdate=None
Bind9Notify=Use Server Value
Lucent3ZoneEduP=False
Lucent3AllowQuery=None
Lucent3AllowTransfer=None
Lucent3AllowUpdate=None
Lucent3CheckNames=Fail
Lucent3Notify=Use Server Value
Lucent4ZoneEduP=False
Lucent4AllowNotify=None
Lucent4AllowQuery=None
Lucent4AllowTransfer=None
Lucent4AllowUpdate=None
Lucent4Notify=Use Server Value
Aging=True
MSAllowTransfer=None
MSAllowUpdate=Yes
AgeNoRefreshHours=1
MSNotify=Yes
AgeRefreshHours=1

```

Example 2 (reversezone)

```

Zone=200.200.200.0/24
ParentAddress=
NetworkAddress=
dnsServers=,revdns.rev.com P 0
RefreshTime=21600
ExpirationTime=604800
RetryPeriod=3600
MinimumTTL=86400
NegativeCacheTTL=86400
ZoneMail=rev1@rev.com
Bind8AllowQuery=Any
Bind8AllowTransfer=Any
Bind8AllowUpdate=Any
Bind8CheckNames=Warn

```

Bind8Notify=No
Bind9AllowNotify=Any
Bind9AllowQuery=Any
Bind9AllowTransfer=Any
Bind9AllowUpdate=Any
Bind9Notify=No
Lucent3ZoneEdup=False
Lucent3AllowQuery=Any
Lucent3AllowTransfer=Any
Lucent3AllowUpdate=Any
Lucent3CheckNames=Warn
Lucent3Notify=No
Lucent4ZoneEdup=False
Lucent4AllowNotify=Any
Lucent4AllowQuery=Any
Lucent4AllowTransfer=Any
Lucent4AllowUpdate=Any
Lucent4Notify=No
Aging=False
MSAllowTransfer=Any
MSAllowUpdate=No
AgeNoRefreshHours=0
MSNotify=No
AgeRefreshHours=0



qip-globalmacpool

qip-globalmacpool allows you to add or delete an individual MAC address from a DHCP server MAC pool or query the MAC pool available to a DHCP server.

Synopsis

```
qip-globalmacpool -n dhcp_server_name -a add|delete|query
[-m mac_address] [-t hardware_type] [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-x mac_address]
```

Parameters

qip-globalmacpool recognizes the following parameters:

- n *dhcp_server_name* Specifies the fully-qualified server name of the MAC pool owner that you want to add or delete a MAC pool address.

- a *add|delete|query* Specifies the process: to add or delete a MAC address, or to query the available MAC pool. The optional argument (-x) is available for -a *add* only, and indicates that specific addresses should be excluded. Using the -a displays the MAC address, hardware type, and excluded attribute specified with the -x (if applicable).

- m *mac_address* Specifies the MAC address to be added or deleted. Supports the trailing wildcard character (*).

- t *hardware_type* Specifies the hardware type of the object with the MAC address to be added. The following options are available:
 - Hardware_Type
 - Ethernet
 - TokenRing
 - AX.25
 - Pronet
 - Chaos
 - IEEE802
 - Arcnet

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- x *mac_address* Indicates that the specified address should be excluded – only applicable when -a is used.

Command line input examples

- To add a MAC address to the “dhcp1.qtek.com” server:

```
qip-globalmacpool -u qipman -p passwd -n dhcp1.qtek.com -a add -m 00008e334455 -h ethernet
```

- To enter a query of the “dhcp1.qtek.com” server for global MAC pool addresses:

```
qip-globalmacpool -u qipman -p passwd -n dhcp1.qtek.com -a query
```

- To delete a MAC address:

```
qip-globalmacpool -a delete -m 00008e334455
```

- To add a MAC address, excluding specific addresses:

```
qip-globalmacpool -a add -x -m 01028b* -t ethernet -n oneill-w2k.quadritek.com
```

Output example

```
MacAddr=111111116771, HardwareType=AX.25
MacAddr=121212121221, HardwareType=Ethernet
MacAddr=121212223333, HardwareType=Ethernet
MacAddr=333213321111, HardwareType=Pronet
MacAddr=333213775511, HardwareType=Chaos
MacAddr=333213776677, HardwareType=Arcnet
MacAddr=551212112212, HardwareType=ieee802
MacAddr=551212121111, HardwareType=Token Ring
MacAddr=553423121212, HardwareType=Ethernet
```



qip-hndbgen

qip-hndbgen is equivalent to performing the **Network Services|NIS Generation** option in the VitalQIP interface which generates NIS records into the ethers file, hosts file, and netmasks file. The output is stored in the NIS servers default directory (specified in the Server Profile) or the directory specified with the `-d` option.

Synopsis

```
qip-hndbgen -n NIS_ServerName [-g loginserver] [-s servername]
  [-u username] [-p password] [-o organization] [-d directory]
```

Parameters

qip-hndbgen recognizes the following parameters:

- `-n NIS_ServerName` Specifies the fully-qualified NIS server name.
- `-g loginserver` Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- `-s servername` Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- `-u username` Specifies the VitalQIP administrator account to be used in establishing the database connection.
- `-p password` Specifies the password for the associated administrator account.
- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-d directory` Specifies the directory name where the information is returned. If no directory is specified, the files are written to the NIS server's default directory specified in the Server Profile.

Command line input example

```
qip-hndbgen -u qipman -p passwd -n nis.qtek.com -d /opt/qip
```



qip-hostgen

qip-hostgen generates a Local Host configuration file to the Local Host Server defined. The output is stored in the specified (or default) directory.

Synopsis

```
qip-hostgen -n host_name [-g loginserver] [-s servername]  
[-u username] [-p password] [-o organization] [-d directory]
```

Parameters

qip-hostgen recognizes the following parameters:

- n *host_name* Specifies the fully-qualified name of the local host server.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- d *directory* Specifies the directory name where the information is returned. The default is the non-DNS server directory or */tmp* if not defined.

Command line input example

```
qip-hostgen -n local.qtek.com -u qipman -p passwd -d /tmp
```

Output example

```
161.251.83.58 blwp029 blwp029.qtek.com # qtek.com;  
161.251.83.59 blwp030 blwp030.qtek.com # qtek.com;  
161.251.83.60 blwp031 blwp031.qtek.com # qtek.com;
```



qip-leasefilegen

qip-leasefilegen creates a VitalQIP DHCP Active Lease file from information stored in the VitalQIP database. The generated Active Lease file can be used to re-create active lease files in case of DHCP server failure or a corrupted active lease database. After this file is generated, you can place this file on your VitalQIP DHCP server and restart the VitalQIP DHCP server so the system reads the new configuration.

Synopsis

```
qip-leasefilegen [-g loginserver] [-s servername] [-u username]
                 [-p password] [-o organization] [-n dhcpServerName] [-d directory]
```

Parameters

qip-leasefilegen recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- n *dhcpServerName* Specifies the fully-qualified DHCP server name.
- d *directory* Specifies the directory where the *dhcp.db* file should reside.

Command line input example

```
qip-leasefilegen -u qipman -p passwd -n dhsp0.qtek.com -d /qiphome/dhcp
```

Output example

```
144.144.144.013 00-60-97-40-ca-62 MDLAPTOP quadritek.com 2 0858360960 0958362760
144.144.144.014 00-80-5f-01-84-f4 ws_support_3 quadritek.com 2 0857758374 0857760174
```



qip-mcancel

qip-mcancel cancels a previously scheduled move.

Synopsis

```
qip-mcancel -t object|subnet -n name|-a address [-g loginserver]  
[-s servername] [-u username] [-p password] [-o organization]  
[-b begin_date] [-e end_date]
```

Parameters

qip-mcancel recognizes the following parameters:

- t *object|subnet* Specifies the type of move you want to cancel, object or subnet.

- n *name*|-a *address* Specifies the object or subnet name or the object or subnet IP address.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- b *begin_date* Begins the cancellation from this date. Use the format mm/dd/yyyy HH:MM (time is optional). If time is used, quotes must be placed around the date and time. (-t *subnet* only)

- e *end_date* Performs cancellation until this date. Use the format mm/dd/yyyy HH:MM (time is optional). If time is used, quotes must be placed around the date and time. (-t *subnet* only)

Command line input example

```
qip-mcancel -u qipman -p passwd -t object -n 198.200.138.10  
qip-mcancel -u qipman -p passwd -t object -a 198.200.138.211
```

Important! If you specify an “object” (using the `-t` option), the *begin_date* and *end_date* fields are ignored.



qip-move

qip-move either moves objects from one subnet to another on demand or pre-schedules the move for a future date. Individual objects, or entire subnets can be moved.

Synopsis

```
qip-move -t Object|Subnet -a from_address|-n from_Name
-l subnet|object
-e to_address [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-d mm/dd/yyyy] [-m hh:mm]
```

Parameters

qip-move recognizes the following parameters:

- t *Object|Subnet* Specifies whether you want to move the object or subnet.
- a *from_address* | -n *from _Name* Specifies the address of the object or subnet from which you want to move from, or the name of the object or subnet from which you want to move.
- l *object|subnet* Specifies whether you want this to be moved to an object or subnet.
- e *to_address* Specifies the address where you want to move the object or subnet to.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- d *mm/dd/yyyy* | -m *hh:mm* If the date is specified, move the objects on the specified date; otherwise move them immediately. Use the format mm/dd/yyyy hh:mm (time is optional).

Command line input examples

- To move an object from one subnet to another:

```
qip-move -a 101.114.21.30 -e 101.114.21.30 -d 12/09/2003
```

- To move an object to a subnet:

```
qip-move -a 101.114.21.30 -t object -l subnet -e 101.114.21.30 -d 12/09/2003
```

- To move an object to another object:

```
qip-move -u qipman -p passwd -a 199.200.13.27 -t object-e 199.200.139.64 -d 12/31/2003
```



qip-msextract

qip-msextract extracts the configuration of a Microsoft DHCP Server running on Windows, so it can be imported into VitalQIP. This configuration is read from the Windows registry and includes scope (subnet) ranges, server- and scope-level options, lease times for each scope, excluded (static) addresses for each scope, and reserved (manual DHCP) addresses for each scope. **qip-msextract** creates data files for three VitalQIP import CLI commands: **qip-template**, **qip-scope**, and **qip-setobject**. It also creates a batch file (called *mstoqip.bat*) that can be run from a DOS window that runs the three CLI commands.

Before you begin

There are several limitations with **qip-msextract**:

- The Microsoft DHCP Server must be running in order for the import file for **qip-setobject** to be created. The reason is that the DHCP Server stores the object names of its reserved (manual DHCP) addresses in its internal database. These are retrieved using a Microsoft-supplied CLI that requires the DHCP Server to be up and running.
- **qip-msextract** must be run on the machine that has both the Microsoft DHCP Server and at least the remote portion of VitalQIP installed.
- All subnets that exist within the Microsoft DHCP Server must be defined within VitalQIP. The Microsoft DHCP Server itself must also be defined. For more information on Microsoft servers, refer to Chapters 7 and 8 in the *VitalQIP Administrator Reference Manual*.

Synopsis

```
qip-msextract [-d directory]
```

Parameters

qip-msextract recognizes the following optional parameter:

- d *directory* Specifies the directory where the CLI data files and batch file is created; if no directory is specified, the files are created in the directory specified by the environment variable QIPHOME.

Command line input example

- To create a file in *c:\temp*:

```
qip-msextract -d c:\temp
```



qip-namingpolicy

qip-namingpolicy retrieves and modifies an organization's naming policies.

Synopsis

```
qip-namingpolicy -n object_class_name [-g loginserver] [-s dataserver]
[-u username] [-p password] [-o organization] [-f output_file]
[-i input_file] [-ac action] [-df data_format]
```

Parameters

qip-namingpolicy recognizes the following parameters:

- n *object_class_name* Specifies the object class name.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *dataserver* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *output_file* A directory and file name for the CLI output. Use the -df parameter to specify the format. If no filename is provided, the output will be to *STDOUT*.

- i *input_file* Specifies the file that contains the input data.

`-ac action` Defines what action to take. Only one action may be performed at a time. You cannot get and modify in the same call, nor can you combine retrievals and modifications in one input file. The *action* variable can have the following values:

`get` - retrieves a specific set of naming policies for an object class or a list of all object classes and naming policies:

- To retrieve a specific set of naming policies for an object class, use the `-n` parameter with the object class name. The naming policy contains the object class name, the Use Default flag, prefix, length, and suffix.
- To retrieve a list of all object classes and naming policies in the database, omit the `-n` parameter.

`mod` - modifies a naming policy for an object class. Use the `-i` option to specify the input file that contains the data. Use the `-df` option to specify the format of the input file.

`-df data_format` Determines the output format. The *data_format* variable has two values: `c` and `n`. Use the `c` variable to produce comma delimited output. Use the `n` variable to produce name value pairs output. If the `-df` parameter is omitted from the command line, the default is name value pairs output. The output data format is identical to the input format shown in “Comma delimited file format” or “Name value pairs file format”.

Comma delimited file format

The first line of a comma delimited list file contains a layout of the items to be entered. The columns can be in any order and any optional columns can be excluded.

The next lines encountered are the data lines. There can be multiple data lines, each separated by a line feed.

Sample comma delimited input file

```
Object Class Name, Use Default, Prefix, Length, Suffix
IBM Workstation, True,,,
Lucent Router, False,crp,6,cpp
```

Name value pairs file format

Each line of the file starts with a field name. The valid field names are identical to the column names for the comma delimited file format. The fields can be listed in any order. There can be more than one object specified in an input file. Objects should be separated by a blank line.

Sample name value pair input file

```
Object Class Name=object class 1
Use Default=False
```

Prefix=abc

Length=6

Suffix=xyz

Object Class Name=object class 2

Use Default=False

Prefix=def

Length=6

Suffix=uvw



qip-objectclass

qip-objectclass adds, modifies, retrieves, and deletes object classes.

Synopsis

```
qip-objectclass [-n object_class_name] [-g loginserver]  
               [-s dataserver] [-u username] [-p password] [-o organization]  
               [-f output_file] [-i input_file] [-ac action] [-df data format]
```

qip-objectclass recognizes the following parameters:

- n *object_class_name* Specifies the object class name.

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.

- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.

- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password* Specifies the password for the associated administrator account.

- o *organization* Specifies the VitalQIP organization (corporation) name.

- f *output_file* A directory and file name for the CLI output. Use the -df parameter to specify the format. If no filename is provided, the output will be to *STDOUT*.

- i *input_file* Specifies the file that contains the input data.

`-ac action` Defines what action to take. Only one action may be performed at a time. You cannot add and modify in the same call, nor can you combine additions and modifications in one input file.

The *action* variable can have the following values:

`add` - adds an object class. Use the `-i` option to specify the input file that contains the data. Use the `-df` option to specify the format of the input file.

`mod` - modifies an object class. Use the `-i` option to specify the input file that contains the data. Use the `-df` option to specify the format of the input file.

`del` - deletes an object class. The `-n` parameter is also required. The object class specified is removed from the database.

`get` - retrieves a specific object class or a list of all object classes:

- To retrieve a specific object class, use the `-n` parameter with the object class name. The object class profile contains the object class name, device type, prefix, length, and suffix.
- To retrieve a list of all object classes in the database, omit the `-n` parameter.

`-df data_format` Determines the output format. The *data_format* variable has two values: `c` and `n`. Use the `c` variable to produce comma delimited output. Use the `n` variable to produce name value pairs output. If the `-df` parameter is omitted from the command line, the default is name value pairs output.

The output data format is identical to the input format shown in “Comma delimited file format” or “Name value pairs file format”.

Comma delimited file format

The first line of a comma delimited list file contains a layout of the items to be entered. The columns can be in any order and any optional columns can be excluded.

The next lines encountered are the data lines. There can be multiple data lines, each separated by a line feed.

Sample comma delimited input file

```
Object Class Name, Device Type, Prefix, Length, Suffix
IBM Workstation, Host, iwp, 6, iws
Lucent Router, Router, crp, 6, cpp
```

Name value pairs file format

Each line of the file starts with a field name. The valid field names are identical to the column names for the comma delimited file format. The fields can be listed in any order. There can be more than one object specified in an input file. Objects should be separated by a blank line.

Sample name value pair input file

```
Object Class Name=object class 1
```

Device Type=host
Prefix=abc
Length=6
Suffix=xyz

Object Class Name=object class 2
Device Type=host
Prefix=def
Length=6
Suffix=uvw



qip-qdhcplease

qip-qdhcplease is used in the upgrade from the VitalQIP DHCP 3.1 and 4.0 servers to the current version of the VitalQIP DHCP server format. The command converts existing VitalQIP DHCP lease database files into the new VitalQIP DHCP lease database file format.

Synopsis

```
qip-qdhcplease -d [directory]
```

Parameters

qip-qdhcplease recognizes the following optional parameter:

-d *directory* Specifies the directory where the existing lease database files reside. If the **-d** option is omitted, the lease database files are assumed to be in the current working directory. The conversion utility creates the new lease file in the same directory as the existing lease files.

Command line input example

To convert the existing lease database files in the *c:\temp* directory to the format of the current version of VitalQIP DHCP:

```
qip-qdhcplease -d c:\temp
```



qip-reclaim

qip-reclaim pings all addresses under the subnet, reclaims unused addresses and produces a report. The report is stored in *QIPHOME/report* directory with the name *SR<subnet_address>.yyyymmddHHMM* (for example, *SR135.111.104.0.200505111542*).

Important! *yyyymmddHHMM* represents the year (yyyy), month (mm), day (dd), hour (HH), and minute (MM).

Although this CLI can be run as a stand alone CLI, **qip-reclaim** is designed to run on scheduled intervals as specified by the administrator of the subnet. (The reclaim schedules must be setup through the GUI or by using the **qip-setreclaimschedule** CLI.) **qip-reclaim** gathers object statistics on a subnet and generates a report each time it runs. On the last run it actually does the reclaim of the objects, or lets the administrator select a set of objects based on the gathered statistics, and reclaims them through the GUI.

For more information on the Reclaim functions of the VitalQIP interface, refer to the “Reclaim Addresses” section in Chapter 4 of the *VitalQIP User’s Guide*.

Before you begin

- The Collection Frequency in the report heading indicates the frequency and the total number of times (as shown above) or the times during the day and number of days (that is, 8:00, 12:00, 16:00 For 5 Days).
- Statistics are stored and displayed for all addresses in the subnet, regardless of whether the address is assigned.
- Name, Object Class and Object Status are obtained from existing objects at the time the report is executed. The name may have changed several times over the reclaim cycle - only the current name is displayed.
- If the object is deleted, the Name and Object Class are blank in the report, the Object Status is Unused, but the other statistics cannot be reset – they continue to grow.
- The length of the object name is limited by the value of the OBJECT_NAME_LEN Global Report Policy.

Synopsis

```
qip-reclaim -a subnet_address -t report|reclaim|both  
[-e email_address] [-g loginserver] [-s servername] [-u username]  
[-p password] [-o organization] [-b]
```

Parameters

qip-reclaim recognizes the following parameters:

-a *subnet_address* Specifies the subnet whose names or IP addresses you want to reclaim.

- t *report|reclaim|both* Specifies the reclaim type. *report* only generates the report and does not reclaim the addresses in subnet. *reclaim* resets the addresses to be unused if applicable. *both* generates a report as well as resets the addresses.
- Important!** Reclaim and both are valid only if the global policy 'AUTOMATIC_RECLAIM' was set to True.
- e *email_address* Specifies a valid email address to which reclaim notices are sent.
- Important!** On Windows platforms, you must enter a valid SMTP Host IP address in the *QSendMail.ini* file located in the system directory (the default value is *localhost*). For example:
- ```
[SendMail]
SMTP Host=10.100.30.1
IIS header=Yes
Mime Encoding=Yes
Return Path=sample@sample.com
```
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- b If -t is set for *reclaim* or *both* on a subnet which is scheduled, an abort needs to be forced with a -b before a reclaim can take place.

### Command line input example

```
qip-reclaim -u qipman -p passwd -a 144.144.0.0
```

### Output example

```
<<< MANUAL SCHEDULED RECLAIM REPORT >>>
```

```
Report Date: 10/26/2001 08:24 Start Date: 10/24/2001 16:15
Collection Frequency: Collect: Every 20 Hours Total: 2 Times
SUBNET: 144.144.0.0
```

| Address | Name | Object Class | Object Status | # Try | # In DNS | # Reachable | Reclaim Status | Last Time Reachable |
|---------|------|--------------|---------------|-------|----------|-------------|----------------|---------------------|
|---------|------|--------------|---------------|-------|----------|-------------|----------------|---------------------|

-----  
144.144.0.1           Unused    Unused    2    0    0    Not In DNS; Unreachable  
144.144.0.2           Unused    Unused    2    0    0    Not In DNS; Unreachable



## qip-report

---

**qip-report** creates an SQL report based on the format file or parameters specified on the command line. The output is stored in the specified (or default) file.

### Synopsis

```
qip-report -r range_type [-g loginserver] [-s servername]
 [-u username] [-p password] [-o organization]
 [-a ip_address|-n name][-c column_name[,column_name...]]
 |[-m format_file] [-t obj_dynamic_type] [-d delimiter]
 [-f report_file]
```

### Parameters

**qip-report** recognizes the following parameters:

- r *range\_type*                    Specifies the type of report you want (for example, corporation, domain, subnet, subnet organization, network, OSPF area).
  
- g *loginserver*                    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*                    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*                      Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*                      Specifies the password for the associated administrator account.
  
- o *organization*                  Specifies the VitalQIP organization (corporation) name.
  
- a *ip\_address*|-n *name*          Specifies the address corresponding to the specified range type (if it is a subnet or network) or the name of a specified range type (domain, subnet organization, OSPF, or Corporation) to be reported on.
  
- c *column\_ name*  
[,*column\_ name*...]                Specifies the name(s) of the database table columns to be included in the report (to streamline your report). You can code multiple column names, separated by commas. Refer to Table 22 following. The default is *all* columns.

- m *format\_file*                    Specifies the file that defines the report requirements. It can include information on column values to be used to filter the report. The specifiable column names are marked with an asterisk (\*) in Table 22 following. (Refer to Note 2 for details.)
  
- t *obj\_dynamic\_type*            Specifies the type of object (for example, dynamic, static, or reserve).
  
- d *delimiter*                    Specifies the delimiter character to separate fields in the output file (such as -d |). The default delimiter is a comma.
  
- f *report\_file*                    Specifies the name of the file where the report is returned. The default is *STDOUT*.

**Table 22     Column names**

| <b>Column name</b> | <b>Additional notes</b>                                                |
|--------------------|------------------------------------------------------------------------|
| MacAddr            | Can be used in the format file to filter the report.                   |
| ObjIpAddr          | IP address of the object.                                              |
| Name*              | Can be used in the format file to filter the report (refer to Note 2). |
| SubnetName*        | Can be used in the format file to filter the report (refer to Note 2). |
| SubnetAddress*     | Can be used in the format file to filter the report (refer to Note 2). |
| FTPServerName*     | Can be used in the format file to filter the report (refer to Note 2). |
| BootfileName*      | Can be used in the format file to filter the report (refer to Note 2). |
| NameServices       |                                                                        |
| NameServiceRecords |                                                                        |
| AuthorizationName  |                                                                        |
| ObjectType         | Type of object that can be passed as the -t parameter.                 |
| ObjectClass*       | Can be used in the format file to filter the report (refer to Note 2). |
| ObjectDesc         |                                                                        |
| DynamicType*       | Can be used in the format file to filter the report (refer to Note 2). |
| HardwareType       |                                                                        |
| ModelType*         | Can be used in the format file to filter the report (refer to Note 2). |
| ClientClass        |                                                                        |

| Column name    | Additional notes                                                       |
|----------------|------------------------------------------------------------------------|
| UserClass      |                                                                        |
| ObjectTag      |                                                                        |
| DHCPServer*    | Can be used in the format file to filter the report (refer to Note 2). |
| DHCPTemplate*  | Can be used in the format file to filter the report (refer to Note 2). |
| LeaseTime*     | Can be used in the format file to filter the report (refer to Note 2). |
| TTLTime*       | Can be used in the format file to filter the report (refer to Note 2). |
| Application*   | Can be used in the format file to filter the report (refer to Note 2). |
| Manufacturer*  | Can be used in the format file to filter the report (refer to Note 2). |
| SerialNumber*  | Can be used in the format file to filter the report (refer to Note 2). |
| AssetNumber*   | Can be used in the format file to filter the report (refer to Note 2). |
| HostID*        | Can be used in the format file to filter the report (refer to Note 2). |
| PurchaseDate   |                                                                        |
| ExpirationDate |                                                                        |
| Contact        |                                                                        |
| Location       |                                                                        |
| RoomID         |                                                                        |
| BillStatus*    | Can be used in the format file to filter the report (refer to Note 2). |
| BillLocation*  | Can be used in the format file to filter the report (refer to Note 2). |
| BillGroup*     | Can be used in the format file to filter the report (refer to Note 2). |
| BillObject*    | Can be used in the format file to filter the report (refer to Note 2). |

**Important!** You can use a format file (specified in the `-m` parameter) that contains the `column=ColumnName[,ColumnName...]` statement to define the columns of data to be output (such as `column=Name,Type,SubnetAddress`). Each output column is displayed consecutively, from left to right, even if you define multiple `column=` statements. You can specify any of the column names shown in the table above.

**Important!** If the column names in the table above are marked with an asterisk (\*), you can also use them to filter the report by including them (in the format `ColumnName=value`) in the “format file” you specified in the `-m` parameter. For example, your format file can contain the record `SubnetAddress=150.1.0.0` to limit the report

to records in which the Subnet Address is **150.1.0.0**. Example 1 shows the use of the `-m` parameter.

### Command line input example 1

- The following example includes the `-m` parameter (to specify the format file that contains report format details):

```
qip-report -r domain -n qtek.com -m rpt.fmt
```

- The `rpt.fmt` format file specified in the `-m` parameter contains:

```
Column=Name,DynamicType,SubnetAddress
Column=ObjIpAddr
Application=qtek app, mci_app
SubnetAddress=150.1.0.0
```

### Output example 1

The resulting output is:

```
Name,DynamicType,SubnetAddress,ObjIpAddr,Application,Billing Location
"hub.au.com", "", 150.1.0.0, 150.1.0.1, qtek app, malvern
"dc.au.com", "", 150.1.0.0, 150.1.0.2
"wpenn.au.com", "", 150.1.0.0, 150.1.0.3, mci_app, boston
"philly.au.com", "", 150.1.0.0, 150.1.0.4, qtek app
"geneva.au.com", "", 150.1.0.0, 150.1.0.5, qtek app, malvern
"alps.au.com", "", 150.1.0.0, 150.1.0.6, mci_app
"bootp1.au.com", "", 150.1.0.0, 150.1.0.14, mci_app
```

### Command line input example 2

- The following example includes the `-c` parameter (to limit the report to specific columns):

```
qip-report -c ObjIpAddr,Name -r subnet -a 150.231.35.160
```

### Output example 2

The resulting output is:

```
ObjIpAddr,Name
150.231.35.169,pc1.lucent.com
150.231.35.168,pc2.lucent.com
150.231.35.164,pc3.lucent.com
150.231.35.163,pc4.lucent.com
150.231.35.161,pc5.lucent.com
```



## qip-rot13

---

**qip-rot13** encrypts a text string by using the ROT13 weak encryption algorithm. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rot13 -i input_string [-f filename]
```

### Parameters

**qip-rot13** recognizes the following parameters:

**-i** *input\_string* Specifies the string to be scrambled.

**-f** *filename* Specifies the file where the information is returned. The default is *STDOUT*.

### Command line input example

```
qip-rot13 -i atest
qip-rot13 -i "test rot"
```

### Output example

```
ngrfg
grfg ebg
```



## qip-rptadminaudit

---

**qip-rptadminaudit** creates an Administrator Audit Report. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptadminaudit [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] -n login_name [-c] [-d start_date]
[-e end_date] [-f filename]
```

### Parameters

**qip-rptadminaudit** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- n *login\_name* Specifies the VitalQIP administrator login name.
- c Specifies that the output is in CSV format.
- d *start\_date* Specifies the audit period start date (format: mm/dd/yyyy).
- e *end\_date* Specifies the audit period start and date (format: mm/dd/yyyy).
- f *filename* Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptadminaudit -u qipman -p passwd -n qipman
```

**Output example**

```
VitalQIP Administrator Audit Report
Date : 1998-07-28 13:47
Organization: VitalQIP Organization
User Name : qipman
Object Name
IP Address. : 161.251.45.79
Domain.....
Class.....: Undefined
Name services: N
TTL time: 0
```



## qip-rptadminrole

---

**qip-rptadminrole** creates an Administrative Role Report with assigned administrators to the role. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptadminrole [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -n role_name [-c] [-f filename]
```

### Parameters

**qip-rptadminrole** recognizes the following parameters:

- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*       Specifies the password for the associated administrator account.
  
- o *organization*   Specifies the VitalQIP organization (corporation) name.
  
- n *role\_name*       Specifies the VitalQIP administrative role name.
  
- c                   Specifies that the output is in CSV format.
  
- f *filename*       Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptadminrole -u qipman -p passwd -n "admin role name" -f adminrole.txt
```

**Output example**

VitalQIP Role/Administrator Report  
Date : 2000-02-09 13:42  
Organization: VitalQIP Organization  
User Name : qipman  
Role: testrole  
user1



## qip-rptaudithistory

---

**qip-rptaudithistory** creates an Object History Report. This report is retrieved one object at a time. If there is more than one object with that “name”, an error is sent stating that there are multiple objects with that name. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptaudithistory -a ip_address|-n name [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
[-d start_date] [-e end_date] [-c] [-f filename]
```

### Parameters

**qip-rptaudithistory** recognizes the following parameters:

- a *ip\_address*|-n *name* Specifies the IP address of the object to be audited or the name of the object to be audited.
- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- d *start\_date* Specifies the audit period start and date (format is mm/dd/yyyy).
- e *end\_date* Specifies the audit period start and date (format is mm/dd/yyyy).
- c Specifies that the output is in CSV format.
- f *filename* Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptaudithistory -u qipman -p passwd -a 10.200.80.10
```

**Output example**

```

VitalQIP Object Audit Report
Date : 2005-05-20 14:30
Organization: VitalQIP Organization
User Name : qipman

ADDED 02/11/2005 11:31:55 qipman

Object Name..... qahp1104
IP Address..... 10.200.80.10
Domain..... seg1.qa.quadritek.com
Class..... Server
Subnet Address..... 10.200.80.0
Subnet Mask..... 255.255.255.0
Subnet Name.....
Name services..... Y (A and PTR records)
Dynamic pushes..... A, PTR, CNAME, MX
TTL time..... Unlimited
 MISC. OBJECT INFORMATION:
 OBJECT 3:
 OBJECT 4:
 OBJECT 5:
 OBJECT INFORMATION:
 OBJECTMAXCHARACTERS1111111112:
 OBJECTMAXCHARACTERS111111111E:

```

| Date/Time | Administrator | Audit Activity |
|-----------|---------------|----------------|
|-----------|---------------|----------------|



## qip-rptdhcp

---

**qip-rptdhcp** creates a DHCP Server Profile Report. This report can be generated in a compressed format. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptdhcp -n dhcp_FQN [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-t compressed|expanded] [-c]
[-f filename]
```

### Parameters

**qip-rptdhcp** recognizes the following parameters:

|                               |                                                                                                                   |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------|
| -n <i>dhcp_FQN</i>            | Specifies the fully-qualified name of the DHCP server on which you are seeking a Server Profile Report.           |
| -g <i>loginserver</i>         | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i>          | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| -u <i>username</i>            | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| -p <i>password</i>            | Specifies the password for the associated administrator account.                                                  |
| -o <i>organization</i>        | Specifies the VitalQIP organization (corporation) name.                                                           |
| -t <i>compressed expanded</i> | Specifies the report type (compressed or expanded).                                                               |
| -c                            | Specifies that the output is in CSV format.                                                                       |
| -f <i>filename</i>            | Specifies the file where the report is stored. The default is <i>STDOUT</i> .                                     |

### Command line input example

```
qip-rptdhcp -u qipman -p passwd -t expanded -n srt08.seg5.qa.quadritek.com
```

**Output example**

```

VitalQIP DHCP Report
Date : 2005-05-20 14:37
Organization: VitalQIP Organization
User Name : qipman
DHCP Server : srt08.seg5.qa.quadritek.com
Report Type : Expanded

Server Name..... srt08.seg5.qa.quadritek.com
Managed Range..... Corporation
Default Directory..... /opt/qip/dhcp
DHCP Template..... general
Accept Client Names..... True
Additional Policies.....
Client Class.....
Debug Information.....
 Debug..... None
 DebugFile..... dhcpd.log
 MaxDebugFileSize..... -1
Failover Server Type..... Standalone/Primary
 CtlReqRetryMax..... 3
 PollDelay..... 60
 SyncBindRetryMax..... 3
 SyncBindingBufSize..... 1024
 Use Failover Server..... False
 WaitCtlRetSecs..... 5
 WaitSyncBindAckSecs..... 5
 WaitSyncBindUpdateSecs..... 15
Remote Server Proxy.....
 Description.....
Scheduled Automatic Updates..... None
Support Bootp..... True
Use Server Policy Template..... False
 DHCP Server Policies.....
 ActiveLeaseExpiration..... Off
 Bootfile..... Default
 CheckTransactionID..... False
 ClientHostNameProcessing..... Ignore
 CompressedLog..... False
 DefaultDescentThreshold..... 0
 DefaultLease..... 90 Days 0 Hours 0 Minutes
 DefaultUnavailableThreshold..... 0
 DropZeroMacAddressPackets..... True
 ExpireAllLeasesOnRestart..... True
 ForceClass..... None
 HonorRequestedLeaseTime..... True
 HonorUnqualifiedBootfile..... False
 InitRebootAddressShuffle..... Off
 IssueDropUnknownClientTrap..... False
 LeaseExpirationSleepTime..... 60000
 LeaveBootpParametersInOptions..... False
 LogLeaseGrantAndRenew..... True
 MacWarningsToEventLog..... False
 MaxOutgoingDhcpMessageSize..... 1024

```

```

MaxPendingSeconds..... 10
MaxUnavailableTime..... 86400
NackDhcpRequestsForDuplicates..... True
NakUnknownClients..... True
Option81Support..... Suppress
PadBootpReply..... True
PingAttempts..... 1
PingBeforeManualBootp..... False
PingBeforeManualDhcp..... True
PingDelay..... 0
PingRetention..... 0
PingSendDelay..... 0
RegisteredClientsOnly..... True
RenewAddressShuffle..... Off
RenewAddressShuffleMaxCount..... 0
SearchDynamicFirst..... False
SendRequestedParamsOnly..... False
SendServerIdLast..... False
ShareAutoBootpAndDynDhcp..... False
SupportAutoRelease..... True
SupportBootpAutoRelease..... True
SupportEncodingLongOptions..... False
SupportRelayAgentDeviceClass..... False
SupportRelayAgentOption..... True
SupportSubnetSelection..... False
ThresholdMonitorSleepTime..... 60
Update QIP Operations.....
 Autorelease..... True
 Bootp..... True
 Decline..... True
 Delete..... True
 Expiration..... True
 Grant..... True
 Release..... True
 Renew..... True
ZeroCiAddr..... False

```

```

Subnet Address: 10.58.208.0
Subnet Mask : 255.255.240.0
Default Router(s):

```

```

Subnet Name:
Effective Domain: seg5.qa.quadritek.com

```

MAC Pool:

```

10.58.208.4 --10.58.208.4 general
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com
10.58.208.6 --10.58.208.6 general
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com
10.58.208.8 --10.58.208.8 general
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com
10.58.208.10 --10.58.208.12 general
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com
10.58.208.14 --10.58.208.24 general
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com

```

10.58.208.26 --10.58.214.50 general  
D-DHCP --- 90 Days 0 Hours 0 Minutes seg5.qa.quadritek.com

Option Template general:

|                         |                           |
|-------------------------|---------------------------|
| Subnet Mask.....        | Same as in Subnet Profile |
| Router.....             | Same as in Subnet Profile |
| Domain Name Server..... | Same as in Subnet Profile |
| Domain Name.....        | Same as in Subnet Profile |

MAC Pool:



## qip-rptfreesubnet

---

**qip-rptfreesubnet** creates a report on free subnets for a given network and subnet mask. It only reports on “unallocated” subnets for a specific network using a specific mask. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptfreesubnet -a network_IP -m subnet_mask [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization] [-c]
[-f filename]
```

### Parameters

**qip-rptfreesubnet** recognizes the following parameters:

- a *network\_IP*      Specifies the network IP address where the free subnets are located.
- m *subnet\_mask*    Specifies the subnet mask.
- g *loginserver*    Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*       Specifies the password for the associated administrator account.
- o *organization*   Specifies the VitalQIP organization (corporation) name.
- c                   Specifies that the output is in CSV format.
- f *filename*       Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptfreesubnet -a 100.101.0.0 -m 255.255.255.0
```

### Output example

```
VitalQIP Free Subnet Report
Date : 1998-07-28 12:59
```

Organization: VitalQIP Organization  
User Name : qipman  
100.101.0.0  
100.101.1.0  
100.101.2.0  
100.101.3.0  
100.101.4.0  
100.101.5.0



## qip-rptinquire

---

**qip-rptinquire** generates an Inquiry Report on the object specified in the command parameter. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptinquire -n name|-a address [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
{-t search_type [[-b sub_searchType]
{[-r subRange <rrType|udf_fieldname>] [-d owner|data|both]}} [-c] [-
f filename]
```

### Parameters

**qip-rptinquire** recognizes the following parameters:

- n *name* | -a *address*    Specifies the name of the object *or* the IP address of the object.
- g *loginserver*            Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*            Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*              Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*              Specifies the password for the associated administrator account.
- o *organization*         Specifies the VitalQIP organization (corporation) name.
- t *search\_type*            The search type of the object. Refer to Table 23 following.
- b *sub\_searchType*        Applicable if the -t *search\_type* is User\_Field or Resource\_Record. Refer to Table 24 following.
- r *subRange*              Applicable if the -t *search\_type* is User\_Field or Resource\_Record. Specifies the resource record type or the user-defined field names listed in Table 24 following.
- d *owner* | *data* | *both*    Applicable only if the -t *search\_type* is Resource\_Record.

- c Specifies that the output is in CSV format.
- f *filename* Specifies the file where the report is stored. The default is *STDOUT*.

**Table 23 Search type fields**

| Search_Type     | Notes                                                                                                 |
|-----------------|-------------------------------------------------------------------------------------------------------|
| All             | Searches all supported fields.                                                                        |
| Corp            | Searches for an Organization, use -n to specify the organization name.                                |
| Subnetorg       | Searches for a Subnet Organization, use -n to specify the subnet organization name.                   |
| Domain          | Searches for a domain, use -n to specify the domain name.                                             |
| Ospf            | Searches for an OSPF Area name, use -n to specify the OSPF Area name.                                 |
| Network         | Searches for a network. Use -n to search for the network name. Use -a to specify the network address. |
| Object          | Searches for an object. Use -n to search for the object name. Use -a to specify the object address.   |
| Router_Group    | Searches for router group names. Use -n to specify the router group name.                             |
| Resource_Record | Searches for the match of the string passed on the <b>Resource Record Owner</b> field.                |
| User_Field      | Searches for the match of the string passed on the <b>User Defined</b> field.                         |

**Table 24 SubSearch type fields**

| Search_Type     | SubSearch_Type                                                |
|-----------------|---------------------------------------------------------------|
| Resource_Record | Domain, reversezone, object.                                  |
| User_Field      | Domain, reversezone, organization, object, subnet, user, all. |

**Command line input example**

```
qip-rptinquire -u qipman -p passwd -t network -a 176.200.0.0
```

**Output example**

VitalQIP Inquiry Report  
Date : 1998-07-28 12:10  
Organization: VitalQIP Organization  
User Name : qipman

| Type    | Address     | Name    |
|---------|-------------|---------|
| -----   |             |         |
| Network | 176.200.0.0 | net_176 |



## qip-rptmanaged

---

**qip-rptmanaged** displays the list of administrators who have been assigned the specified infrastructure item as an element of their managed list.

### Synopsis

```
qip-rptmanaged -t owner_type -n name [-a address [-m access_mode]
 [-g loginserver] [-s servername] [-u username] [-p password]
 [-o organization] [-f report_file]
```

### Parameters

**qip-rptmanaged** recognizes the following parameters:

- t *owner\_type*      Specifies the type of item to search for. Options are domain, network, OSPF, addr\_range, obj\_range, user\_group, subnet\_org, subnet, application, object.
- n *name*            Specifies the name of the item. Names are used for Domain, OSPF, Subnet Organization, Application, User Group, and Server.
- a *ip\_address*      Specifies the IP Address or IP Address Range for Network, Subnet, Object, Address Range, and Object Range *owner\_types*. Address Range and Object Range are specified as *start\_ip\_address* - *end\_ip\_address*.
- m *access\_mode*      Specifies if you want the list of administrators who have **read** or **write** access to the name or address. The default *access\_mode* is **write**.
- g *loginserver*      Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*      Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*        Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*        Specifies the password for the associated administrator account.
- o *organization*    Specifies the VitalQIP organization (corporation) name.

`-f report_file` Specifies the file where the report is written. If `-f` is not specified, the report is written to *STDOUT*.



## qip-rptobjectlst

---

**qip-rptobjectlst** creates a report of a list of objects associated with the object owner passed in the command line. The owner of the objects can be one of the following: corporation, domain, OSPF, subnet, organization, application, administrator, location, subnet, or network. The format of the report can be a CSV file if the `-c` parameter is specified. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptobjectlst -a net/subnetIP|-n name -t owner_type
 [-g loginserver] [-s servername] [-u username] [-p password]
 [-o organization] [-l alloc_type] [-d dynamic_type] [-j obj_class]
 [-c] [-f filename]
```

### Parameters

**qip-rptobjectlst** recognizes the following parameters:

- a *net/subnetIP*|-n *name* Specifies the address of the range (for example, network or subnet) or the name of the range from which you want to retrieve the list of objects.
- t *owner\_type* Specifies the owner type under which IP objects belong. Refer to Table 25 following.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- l *alloc\_type* Specifies the Allocation type of the objects to report. Refer to Table 26 following for valid values.

- d *dynamic\_type* Specifies the Dynamic type of object. Refer to Table 27 following for valid values.  
**Important!** Valid only when the Allocation\_Type is “Dynamic”.
- j *obj\_class* Specifies the object class for which you want to retrieve the list of manufacturers. The following object classes are available:
- Workstation
  - X-terminal
  - PC
  - Printer
  - Server
  - Wiring\_HUB
  - Router
  - Bridge
  - Terminal\_Server
  - Switch
  - Legacy\_System
  - Gateway
  - Test\_Equipment
  - Any user-defined object class
  - Undefined
  - Others
- c Requests output in CSV format.
- f *filename* Specifies the file where the report is stored. The default is *STDOUT*.

**Table 25 Owner type fields**

| Owner_Type  | Notes                                                                                                                                                           |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Corp        | Displays objects for an Organization, use <b>-n</b> to specify the organization name.                                                                           |
| Subnetorg   | Displays objects for a Subnet Organization, use <b>-n</b> to specify the subnet organization name.                                                              |
| Domain      | Displays objects for a domain, use <b>-n</b> to specify the domain name.                                                                                        |
| OSPF        | Displays objects for an OSPF Area, use <b>-n</b> to specify the OSPF area name.                                                                                 |
| Network     | Displays objects for a network. Use <b>-a</b> to specify the network address.                                                                                   |
| Subnet      | Displays objects for a subnet. Use <b>-a</b> to specify the subnet address.                                                                                     |
| Admin       | Displays objects for an administrator. Use <b>-n</b> to specify the administrator login name.                                                                   |
| Application | Displays objects for an application. Use <b>-n</b> to specify the application name.                                                                             |
| Location    | Displays objects for the specified location ID. Use <b>-n</b> to specify the location ID. The location ID can be obtained from a call to <b>qip-getloclst</b> . |

**Table 26 Allocation types fields**

| Allocation_Types | Description                                   |
|------------------|-----------------------------------------------|
| All              | All objects regardless of assignment.         |
| Static           | An object that has been statically assigned.  |
| Reserved         | An object that has been reserved.             |
| Dynamic          | An object that has been dynamically assigned. |
| Scheduled        | An object that is scheduled to be moved.      |

**Table 27 Dynamic types fields**

| Dynamic_Types | Description                                                 |
|---------------|-------------------------------------------------------------|
| M-BOOTP       | The address was dynamically assigned using manual bootp.    |
| M-DHCP        | The address was dynamically assigned using manual DHCP.     |
| A-BOOTP       | The address was dynamically assigned using automatic bootp. |
| A-DHCP        | The address was dynamically assigned using automatic DHCP.  |
| D-DHCP        | The address was dynamically assigned using dynamic DHCP.    |
| All           | All dynamic objects                                         |
| None          | Dynamic (None) objects.                                     |

**Command line input example**

```
qip-rptobjectlst -u qipman -p passwd -a 10.0.0.0 -t subnet
```

**Output example**

```
VitalQIP Object List Report
Date : 1998-07-28 11:51
Organization: VitalQIP Organization
User Name : qipman
Range: Domain qtek.com
 Server(s):
 DNO0CORP.qtek.com
 DNO0DNS1.qtek.com
 DNO0DNS2.qtek.com
 DN73DNS1.qtek.com
 DN74DNS1.qtek.com
 DN75DNS1.qtek.com
Object Class: ALL Object Type: ALL

Network 10.0.0.0 net_10.0.0.0
 Subnet 10.98.60.0 Name: SN10.98.60.0 Status: Y
 IP Address Object Name Obj Class
```

-----  
10.98.60.10      BHAM3000.qtek.      Undefined  
Subnet 10.98.62.0 Name: SN10.98.62.0 Status: Y



## qip-rptrole

---

**qip-rptrole** generates an administrative role managed list report. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptrole [-n role_name] [-g loginserver] [-s servername]
 [-u username] [-p password] [-o organization] [-c] [-f filename]
```

### Parameters

**qip-rptrole** recognizes the following parameters:

- n *role\_name*      Specifies the VitalQIP administrative role name.
  
- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*       Specifies the password for the associated administrator account.
  
- o *organization*   Specifies the VitalQIP organization (corporation) name.
  
- c                   Specifies that the output is in CSV format.
  
- f *filename*       Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptrole -u qipman -p passwd -f output.txt
```

### Output example

```
VitalQIP Role Managed List Report
Date : 2005-05-20 14:39
Organization: VitalQIP Organization
User Name : qipman
Role: tester1
 Domain: qa.quadritek.com
 Domain: quadritek.com
```

Domain: do.main  
Subnet Organization: SubOrg  
Subnet Organization: 99

Role: tester2  
Domain: seg1.qa.quadritek.com  
Subnet Organization: ethel  
Address Range: 10.200.100.128,10.200.100.191



## qip-rptzonerr

---

**qip-rptzonerr** generates a report on the resource records for the specified zone. The output is stored in the specified (or default) file.

### Synopsis

```
qip-rptzonerr -n zone_name|-a zone_ip/mask [-g loginserver]
 [-s servername] [-u username] [-p password]
 [-o organization] [-r rr_types] [-e] [-f filename]
```

### Parameters

**qip-rptzonerr** recognizes the following parameters:

- n *zone\_name*      Specifies the name of the zone.
  
- a *zone\_ip/mask*   Specifies the reverse zone IP address and the zone mask.
  
- g *loginserver*   Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*       Specifies the password for the associated administrator account.
  
- o *organization*   Specifies the VitalQIP organization (corporation) name.
  
- r *rr\_types*        Specifies the type of resource record to be reported (for example, CNAME, SOA, NS, WKS, and so on). The default is All.
  
- e                    Specifies the zone extension is to be included
  
- f *filename*        Specifies the file where the report is stored. The default is *STDOUT*.

### Command line input example

```
qip-rptzonerr -u qipman -p passwd -n qtek.com -r A
```

**Output example**

```

; Other object level resource records
;*****
nnsa2.qtek.com. IN A 192.85.154.30
;*****
; Other domain level resource records
;*****
ahmlir3.mail.qtek.com. IN A 20.191.24.43
ahmlir2.mail.qtek.com. IN A 20.191.24.42
ahmlir1.mail.qtek.com. IN A 20.191.24.41
y2ktipskb.csts.qtek.com. IN A 20.230.144.234
```



## qip-scope

---

**qip-scope** defines a DHCP scope by selecting a set of addresses and assigning it an Object Profile. This can be performed by specifying individual parameters within the CLI command or creating a file with all this information in it and placing the filename (**-f *input\_file***) in the parameter.

### Synopsis

```
qip-scope [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] -a subnet_address -t dhcp_type
 -r start,end_Address -d domn_name -m name_service(Y/N)
 [-w dynamic_update] -n dhcp_server -e dhcp_template [-c object_class]
 [-k vendor_class] [-l lease_time] [-x user_class] [-q error_file]
```

**or** (to specify a file containing the parameter information)

```
qip-scope -f input_file [-j reject_file] [-g loginserver]
 [-s servername] [-u username] [-p password] [-o organization]
 [-q error_file]
```

### Parameters

**qip-scope** recognizes the following parameters:

|                             |                                                                                                                   |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------|
| -g <i>loginserver</i>       | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i>        | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| -o <i>organization</i>      | Specifies the VitalQIP organization (corporation) name.                                                           |
| -u <i>username</i>          | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| -p <i>password</i>          | Specifies the password for the associated administrator account.                                                  |
| -a <i>subnet_address</i>    | Specifies the subnet whose IP address(es) is associated with this DHCP scope.                                     |
| -t <i>dhcp_type</i>         | Specifies a DHCP type. The options are A(automatic) or D(dynamic).                                                |
| -r <i>start,end_Address</i> | Specifies the beginning and ending address that defines the DHCP scope of addresses, separated by a comma.        |

|                          |                                                                                                                   |
|--------------------------|-------------------------------------------------------------------------------------------------------------------|
| -d <i>domn_name</i>      | Specifies the fully-qualified domain name.                                                                        |
| -m <i>name_service</i>   | Specifies “Y” or “N” to register this object with the name service.                                               |
| -w <i>dynamic_update</i> | Specifies “Y” or “N” to turn on or off the dynamic update flags.                                                  |
| -n <i>dhcp_server</i>    | Specifies the fully-qualified DHCP server name to which you are assigning this scope.                             |
| -e <i>dhcp_template</i>  | Specifies the DHCP template to be assigned to this scope.                                                         |
| -c <i>object_class</i>   | Specifies the object class description.                                                                           |
| -k <i>vendor_class</i>   | Specifies the vendor class to be assigned to this scope.                                                          |
| -l <i>lease_time</i>     | Specifies the lease time to be assigned to this scope.                                                            |
| -x <i>user_class</i>     | <b>QDHCP servers only:</b> Specifies user class to be assigned to scope.                                          |
| -q <i>error_file</i>     | Specifies the filename where error messages are written, if they occur. The default is <i>STDERR</i> .            |
| <b>or</b>                |                                                                                                                   |
| -f <i>input_file</i>     | Specifies the scope definition format filename, as shown in Table 28.                                             |
| -j <i>reject_file</i>    | Specifies the file where records that were not added or modified are placed.                                      |
| -g <i>loginserver</i>    | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i>     | Specifies the database server name. This value is the value of QIPDATASERVER environment variable.                |
| -u <i>username</i>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |

- `-p password` Specifies the password for the associated administrator account.
- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-q error_file` Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

The `-f input_file` for **qip-scope** is in CSV format, with fields as described in Table 28:

**Table 28** qip-scope file layout

| Field                        | Description                                          |
|------------------------------|------------------------------------------------------|
| SubnetAddress                | In dotted decimal format. <i>Required.</i>           |
| DHCPType                     | A-DHCP=Automatic or D-DHCP=Dynamic. <i>Required.</i> |
| StartAddress                 | In dotted decimal format. <i>Required.</i>           |
| EndAddress                   | In dotted decimal format. <i>Required.</i>           |
| FullyQualifiedDomainName     | <i>Required.</i>                                     |
| UseNameService               | Y=Yes, N=No. Optional.                               |
| DynamicUpdate                | Y=Yes, N=No. Optional                                |
| ObjectClass                  | Optional.                                            |
| FullyQualifiedDHCPServerName | Required.                                            |
| DHCPTemplateName             | Required.                                            |
| VendorClass                  | Optional.                                            |
| LeaseTime                    | Optional.                                            |
| UserClass                    | Optional.                                            |

#### Command line input example 1

```
qip-scope -u qipman -p passwd -a 199.200.139.0 -t A -r 199.200.139.10,199.200.139.63
-c PC -n dhcpserver.bedrock.com -d bedrock.com -e general -m Y -x class1 -l 300
```

## Command line input example 2

```
qip-scope -f parmfile
```

The parameters file (specified as `-f parmfile` in this example) contains the following two records:

```
10.200.80.0,D,10.200.80.2,10.200.80.16,quadritek.com,Y,Y,PC,
dhcpserver1.quadritek.com,10.200.80.0_Template,,120,
10.201.80.0,A,10.201.80.1,10.201.80.1,quadritek.com,Y,N,PC,
dhcpserver1.quadritek.com,10.201.80.0_Template,,4294967295,uclass
```



## qip-search

---

**qip-search** searches through various name or address fields in the database and returns the address, name, and object type of records matching the search criteria. The output is stored in the specified file (or default file), in a | delimited format.

**Important!** Wildcard searches are permitted when searching for a name. Use the asterisk (\*) as the wildcard character, and surround the criteria with double quotes (for example, “p\*”).

### Synopsis

```
qip-search -n name|-a address [-g loginserver] [-s servername]
 [-u username] [-p password] [-o organization] [-t search_type]
 [-b sub_search_type] [-r subRange] [-d owner|data|both] [-f filename]
```

### Parameters

**qip-search** recognizes the following parameters:

- |                                   |                                                                                                                                                                     |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -n <i>name</i>  -a <i>address</i> | Specifies either the name ( <i>not</i> fully-qualified: for example, sec1 <i>not</i> sec1.bedrock.com) or the IP address of the object for which you are searching. |
| -g <i>loginserver</i>             | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.                                                   |
| -s <i>servername</i>              | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                          |
| -u <i>username</i>                | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                    |
| -p <i>password</i>                | Specifies the password for the associated administrator account.                                                                                                    |
| -o <i>organization</i>            | Specifies the VitalQIP organization (corporation) name.                                                                                                             |
| -t <i>search_type</i>             | Specifies the search type that you want to search for. Refer to Table 29 following.                                                                                 |
- Important!** If the *search\_type* is UserFields, you must also use the -b parameter for the UDF type and the -n parameter to specify the User Defined Field value or use a wildcard combination.

- b *sub\_search\_type*      The search type within the *search\_type* value (above). This option is only relevant when your *search\_type* is “Resource Record” or UserFields. Refer to Table 31 following.
  
- r *subRange*              Specifies the “Resource Record” type or **UserFields** name. Refer to Table 30 following.
  
- d *owner | data | both*    Specifies the search target (owner, data, or both). This parameter is only relevant if your *search\_type* is “Resource Record”.
  
- f *filename*                Specifies the file where the information is stored. The default is *STDOUT*.

**Table 29      Search types**

| Search type       | Notes                                                                                                                             |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| All               | Searches all supported fields.                                                                                                    |
| Alias             | Searches for an object’s alias names.                                                                                             |
| Corp              | Searches for an Organization, use -n to specify the organization name.                                                            |
| Subnetorg         | Searches for a Subnet Organization, use -n to specify the subnet organization name.                                               |
| Domain            | Searches for a domain, use -n to specify the domain name.                                                                         |
| OSPF              | Searches for an OSPF Area name, use -n to specify the OSPF area name.                                                             |
| Network           | Searches for a network. Use -n to search for the network name. Use -a to specify the network address.                             |
| Object            | Searches for an object. Use -n to search for the object name. Use -a to specify the object address.                               |
| “Router Group”    | Searches for router group names. Use -n to specify the router group name.                                                         |
| “Mac Address”     | Searches for a MAC address. Use -a to specify the MAC address.                                                                    |
| UserFields        | Searches for values of user defined fields. Use -n to specify the value to search for. Wildcards are valid.                       |
| “DECNet Address”  | Searches for a specific DECNet Address. Use -n to specify the value to search for. Enter the search criteria in AREA,NODE format. |
| “Resource Record” | Searches for the match of the string passed on the Resource Record Owner field.                                                   |

**Table 30 Resource Record types**

| Record type | Description                                 |
|-------------|---------------------------------------------|
| CNAME       | Canonical Name                              |
| A           | Host IPv4                                   |
| HINFO       | Host Information                            |
| MX          | Mail Exchange                               |
| NS          | Name Server                                 |
| PTR         | Pointer                                     |
| TXT         | Text                                        |
| WKS         | Well Known Services                         |
| AAAA        | Host IPv6                                   |
| AFSDB       | Andrew File System                          |
| MB          | Mailbox Name                                |
| MG          | Mail Group                                  |
| MINFO       | Mailbox Information                         |
| MR          | Mail Rename                                 |
| ISDN        | Integrated Services Digital Network         |
| SRV         | Server Resource Record                      |
| X25         | PSDN (Public Switched Data Network) address |

Refer to Table 31 only if the specified *search\_type* is “Resource Record” or UserFields.

**Table 31 Sub search types**

| Sub_Search types | Notes                                                                                                                      |
|------------------|----------------------------------------------------------------------------------------------------------------------------|
| Object           | For resource records or user fields associated with objects. The search type can be “Resource Record” or UserFields.       |
| Domain           | For resource records or user fields associated with domains. The search type can be “Resource Record” or UserFields.       |
| ReverseZone      | For resource records or user fields associated with reverse zones. The search type can be “Resource Record” or UserFields. |

| Sub_Search types | Notes                                                                          |
|------------------|--------------------------------------------------------------------------------|
| Organization     | The user fields defined for organizations. The search type must be UserFields. |
| Subnet           | The user fields associated with subnets. The search type must be UserFields.   |
| User             | The user fields associated with users. The search type must be UserFields.     |
| All              | All of the above sub-search types are included.                                |

### Command line input example 1

- To search for all objects that begin with the string “pc”. The search results are returned in the *search.txt* file:

```
qip-search -u qipman -p passwd -t Object -n "pc*" -f search.txt
```

### Output example 1

```
10.98.4.10|pc0100.qtek.com|OBJECT
10.98.4.11|pc0200.qtek.com|OBJECT
10.98.4.12|pc0300.qtek.com|OBJECT
10.98.4.13|pc0400.qtek.com|OBJECT
10.98.4.14|pc0500.qtek.com|OBJECT
10.98.4.15|pc0600.qtek.com|OBJECT
10.98.4.16|pc0700.qtek.com|OBJECT
10.98.4.17|pc0800.qtek.com|OBJECT
10.98.4.18|pc0900.qtek.com|OBJECT
```

### Command line input example 2

- To search for the subnet in the specific address and return the search results in the *search.txt* file:

```
qip-search -u qipman -p passwd -a 199.200.139.0 -t Subnet -f search.txt
```

### Output example 2

```
198.200.139.0|sn_139|Subnet
```

### Command line input example 3

- To search the records in the resource record with owner fields that consist of “qtek” and limit the search to the A records associated with the object. The search results are returned in the *search.txt* file:

```
qip-search -u qipman -p passwd -n qtek -t "Resource Record" -b Object
-d owner -r A -n "*" -f search.txt
```

### Output example 3

```
dns1.qtek.com A 10.100.23.1|obj1.qtek.com|OBJECT
dhcp1.qtek.com A 10.100.23.100|obj2.qtek.com|OBJECT
```

**Command line input example 4**

- To search for all user-defined fields used for domains where the value starts with “West”:  
qip-search -t UserFields -b Domain -n "West\*"

**Output example 4**

```
West Coast|domain.com|DOMAIN/REGION
```

**Command line input example 5**

- To search for the user-defined field “REGION” of all types with any value:  
qip-search -t UserFields -b All -r "REGION" -n "\*"

**Output example 5**

```
West Coast|domain.com|DOMAIN/REGION
East Coast|example.com|DOMAIN/REGION
```

**Command line input example 6**

- To return entries from Resource Records tab of the Object Profile:  
qip-search -n <object name> -t "Resource Record" -b Object

**Command line input example 7**

- To return only CNAME entries from Resource Records tab of Object Profile (does not include Alias names from Aliases tab of Object Profile):  
qip-search -n <object name> -t "Resource Record" -r CNAME

**Command line input example 8**

- To return object record with <alias name> in the Aliases tab of an Object Profile:  
qip-search -n <alias name> -t Alias



## qip-searchacltemplates

---

**qip-searchacltemplates** returns a list of zones followed by the servers associated with each zone that are associated with ACL Templates, if they exist.

### Synopsis

```
qip-searchacltemplates -tn template_name [-g loginserver]
[-s dataserver] [-o organization] [-u username] [-p password]
[-f output_file] [-i input_file]
```

### Parameters

**qip-searchacltemplates** recognizes the following parameters:

- tn *template\_name* Specifies the name of the template being retrieved.
  
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- f *output\_file* Specifies the file where the report is stored. The default is *STDOUT*.
  
- i *input\_file* Specifies the file that contains the input data.

### Output example

```
ctest.com
server1.test.com
server2.test3.com
secondtest.com
serverprimary.secondtest.com
serversecondary.test.com
```

**Important!** Servers do not have to belong to the same domain to be associated with that domain. They only need to be a primary or secondary server in that domain. Each zone is

listed followed by its associated servers that contain ACL Templates. If the zone, by itself, contains a template, only the zone name is listed.



## qip-setacltemplate

---

**qip-setacltemplate** sets the values for a specific template.

### Synopsis

```
qip-setacltemplate [-a]|-i input_file_name [-tn old_template_name]
 [-g loginserver] [-s servername] [-o organization] [-u username]
 [-p password] [-f output_file]
```

### Parameters

**qip-setacltemplate** recognizes the following parameters:

- |                              |                                                                                                                                                                             |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -a                           | Specifies that a template is added. If you do not specify this parameter, the template specified by the <i>-tn</i> parameter is replaced by the contents of the input file. |
| -i <i>input _filename</i>    | Specifies the file that contains the input data.                                                                                                                            |
| -tn <i>old_template_name</i> | Specifies the name of the template being retrieved.                                                                                                                         |
| -g <i>loginserver</i>        | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.                                                           |
| -s <i>servername</i>         | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                  |
| -o <i>organization</i>       | Specifies the VitalQIP organization (corporation) name.                                                                                                                     |
| -u <i>username</i>           | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                            |
| -p <i>password</i>           | Specifies the password for the associated administrator account.                                                                                                            |
| -f <i>output_file</i>        | Specifies the file where the report is stored. The default is <i>STDOUT</i> .                                                                                               |

### Input file example

Refer to “qip-getacltemplate”, on page 52 for an example of an input file for this CLI.



## qip-setadminrole

---

**qip-setadminrole** adds or modifies an Administrative role, as specified in the input data file.

### Before you begin

- The field **RoleName** is mandatory and must be the first field.
- The field **RoleDescription** is optional, and if it exists must be the second field.
- The managed list can be in any order. Available managed list types are: Domain, Network, OSPF, Subnet Organization, Subnet, Object, Application, User Group, server, Address Range, Object Range, Administrative Role, and Administrator. All managed lists have four fields; Type, Name, Address, and ReadOnly. If any field is blank, use a comma “,” to separate it from the next field, as shown in the “Input file content example”.
- The field **Administrator** extracts the managed list from the administrator and appends it to this creating/modifying role.

### Synopsis

```
qip-setadminrole [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -f input_filename [-r reject_file]
 [-e errmsg_file] [-t]
```

### Parameters

**qip-setadminrole** recognizes the following parameters:

|                                       |                                                                                                                              |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.            |
| <code>-s <i>servername</i></code>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                   |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used for establishing the database connection.                            |
| <code>-p <i>password</i></code>       | Specifies the password for the associated administrator account.                                                             |
| <code>-o <i>organization</i></code>   | Specifies the VitalQIP organization (corporation) name.                                                                      |
| <code>-f <i>input_filename</i></code> | Specifies the directory and filename of the input data. If the name contains spaces, it must be enclosed by quotation marks. |
| <code>-r <i>reject_file</i></code>    | Specifies the file where records that are not added or modified are placed.                                                  |

-e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

-t Sets the flag to modify the administrative role.

### Command line input example

```
qip-setadminrole -u qipman -p passwd -f c:\temp\adminprof.txt
```

### Input file content example

```
RoleName=testrole
RoleDescription=This is a test role
ManagedList:
Application,"Sample Application",,True
Subnet Organization,"psl_suborg1",,True
Domain,"usa.world.com",,True
Network,"net_73",73.0.0.0,True
AddressRange,14.14.14.1-14.14.14.100,Flase
AdministrativeRole,"Subnet Admin Role"
Administrator=subnet-admin
```



## qip-setbillinfo

---

**qip-setbillinfo** adds billing locations, billing user groups, or billing object classes (using the **-t** option) to the specified organization. You cannot modify billing information through the CLI.

### Synopsis

```
qip-setbillinfo [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] [-r reject_file] [-e errmsg_file]
 -t Location|UserGroup|ObjectClass -f filename
```

### Parameters

**qip-setbillinfo** recognizes the following parameters:

|                                                 |                                                                                                                              |
|-------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| <b>-g</b> <i>loginserver</i>                    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.            |
| <b>-s</b> <i>servername</i>                     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                   |
| <b>-u</b> <i>username</i>                       | Specifies the VitalQIP administrator account to be used for establishing the database connection.                            |
| <b>-p</b> <i>password</i>                       | Specifies the password for the associated administrator account.                                                             |
| <b>-o</b> <i>organization</i>                   | Specifies the VitalQIP organization (corporation) name.                                                                      |
| <b>-r</b> <i>reject_file</i>                    | Specifies the file where records that are not added or modified are placed.                                                  |
| <b>-e</b> <i>errmsg_file</i>                    | Specifies the filename where error messages are written, if they occur. The default is <i>STDERR</i> .                       |
| <b>-t</b> <i>Location UserGroup ObjectClass</i> | Specifies the billing type.                                                                                                  |
| <b>-f</b> <i>filename</i>                       | Specifies the directory and filename of the input data. If the name contains spaces, it must be enclosed by quotation marks. |

### Command line input example

```
qip-setbillinfo -u qipman -p passwd -t ObjectClass -f c:\temp\input.txt
```

## Input File Content Example

**Important!** Object classes must not exist in VitalQIP. The following are possible valid examples:

NewBridge  
MyPC



## qip-setclientclass

---

Use the **qip-setclientclass** CLI to specify a client class comprising a vendor class, a device class or one or more user classes, and then attach a DHCP option template or a client class policy template to it. This client class can then be attached to a Lucent DHCP server without being assigned to a specific address scope.

### Before you begin

Although **OptionTemplate** and **ClientClassPolicyTemplate** fields are optional, at least one of them must exist in the input file.

### Synopsis

```
qip-setclientclass [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-r reject_file] [-e errmsg_file]
[-n client_class] -f input_filename
```

### Parameters

**qip-setclientclass** recognizes the following parameters:

- |                                 |                                                                                                                              |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| <b>-g</b> <i>loginserver</i>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.            |
| <b>-s</b> <i>servername</i>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                   |
| <b>-u</b> <i>username</i>       | Specifies the VitalQIP administrator account to be used for establishing the database connection.                            |
| <b>-p</b> <i>password</i>       | Specifies the password for the associated administrator account.                                                             |
| <b>-o</b> <i>organization</i>   | Specifies the VitalQIP organization (corporation) name.                                                                      |
| <b>-r</b> <i>reject_file</i>    | Specifies the file where records that are not added or modified are placed.                                                  |
| <b>-e</b> <i>errmsg_file</i>    | Specifies the filename where error messages are written, if they occur. The default is <b>STDERR</b> .                       |
| <b>-f</b> <i>input_filename</i> | Specifies the directory and filename of the input data. If the name contains spaces, it must be enclosed by quotation marks. |

-n *client class* Specifies the client class information.

### Input file example

For a description of the fields in the input file example, refer to Table 33.

```
ClientClassName=new client class
ClassType=USER
AttachedClass=qa support
OptionTemplate=general
ClientClassPolicyTemplate=policyTemp
```

**Table 32** Input file field definitions

| Field name/label          | Value type                 | Description                                                                                                         |
|---------------------------|----------------------------|---------------------------------------------------------------------------------------------------------------------|
| ClientClassName           | [M] String                 | Specifies the client class name                                                                                     |
| ClassType                 | [M] VENDOR, USER or DEVICE | Specifies the class type of client class                                                                            |
| AttachedClass             | [M] String                 | Specifies the vendor class if the ClassType is set to VENDOR or the user class(es) if the ClassType is set to USER. |
| OptionTemplate            | [O] String                 | Specifies the option template attached to the client class.                                                         |
| ClientClassPolicyTemplate | [O] String                 | Specifies the client class policy template attached to the client class.                                            |



## qip-setcontact

---

**qip-setcontact** allows you to add a new contact, or to modify an existing contact within the VitalQIP database. If you want to modify existing information, you can retrieve the existing contact information by using the command **qip-getcontactlst**. Contact information is set as specified in the *input\_filename* (-f parameter).

### Synopsis

```
qip-setcontact -f input_filename [-g loginserver] [-s servername]
 [-u username] [-p password] [-o organization] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**qip-setcontact** recognizes the following parameters:

- f *input\_filename*** Specifies the file that contains the input data.
- g *loginserver*** Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*** Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*** Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*** Specifies the password for the associated administrator account.
- o *organization*** Specifies the VitalQIP organization (corporation) name.
- r *reject\_file*** Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file*** Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Command line input example

```
qip-setcontact -f contactfile
```

## Input file example

For a description of the fields in the input file example, refer to Table 33.

FirstName=Bob  
LastName=Smith  
PhoneNum=555-5555  
PagerNum=555-1234  
EmailAddress=bsmith@company.com

**Table 33** Input file field definitions

| <b>Field</b> | <b>Description</b>                  |
|--------------|-------------------------------------|
| FirstName    | The first name of the contact.      |
| LastName     | The last name of the contact.       |
| PhoneNum     | The phone number of the contact.    |
| PagerNum     | The pager number of the contact.    |
| EmailAddress | The e-mail address of this contact. |



## qip-setdnsrr

---

**qip-setdnsrr** adds or deletes resource records specified in an input file to/from the associated object, domain, or reverse zone that is passed with the **-t** option.

### Synopsis

```
qip-setdnsrr -t owner type -a ip_address [-n domn_name
-f input_filename [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-r reject_file] [-e errmsg_file]
```

### Parameters

**qip-setdnsrr** recognizes the following parameters:

- |                                 |                                                                                                                   |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <b>-t</b> <i>owner_type</i>     | Specifies the owner type of the resource records. Either object, domain, or reverse zone.                         |
| <b>-a</b> <i>ip_address</i>     | Specifies the IP address of the object or reverse zone with which the resource records are associated.            |
| <b>-n</b> <i>domn_name</i>      | Specifies the domain name with which the resource records are associated.                                         |
| <b>-f</b> <i>input_filename</i> | Specifies the name of the file that contains the input data.                                                      |
| <b>-g</b> <i>loginserver</i>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <b>-s</b> <i>servername</i>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <b>-u</b> <i>username</i>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| <b>-p</b> <i>password</i>       | Specifies the password for the associated administrator account.                                                  |
| <b>-o</b> <i>organization</i>   | Specifies the VitalQIP organization (corporation) name.                                                           |
| <b>-r</b> <i>reject_file</i>    | Specifies the file where records that are not added or modified are placed.                                       |
| <b>-e</b> <i>errmsg_file</i>    | Specifies the filename where error messages are written, if they occur. The default is <i>STDERR</i> .            |

### Command line input examples

```
qip-setdnsrr -u qipman -p passwd -f rr.dat -t object -a 144.144.144.4
qip-setdnsrr -t domain -n lucent.com -f domainrr.dat
qip-setdnsrr -t reverse zone -a 198.200.138.0 -f rz.dat
```

### Input file example

For a description of the fields in the input file example, refer to Table 34 following.

```
ResourceRecOwner=host.qtek.com
ResourceRecType=TXT
ResourceClass=IN
MinimumTTL=-1
ResourceRecText=this is for XXXX
ApplyToZone=1
ResourceRecOwner=host2.qtek.com
ResourceRecType=TXT
ResourceClass=IN
MinimumTTL=3600
ResourceRecText=this is for XXXX
ApplyToZone=1
ChangeFlag=1
ExternalFlag=1
Tombstoned=1
ExternalComment="Comment"
ExternalTimestamp=03/09/2005 20:37:42
```

**Table 34 Input file field definitions**

| Field Name/Label                                                        | Description                                                                                                                                               |
|-------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Important!</b> The following fields appear once per resource record. |                                                                                                                                                           |
| ResourceRecOwner   Owner                                                | Owner of the resource record.                                                                                                                             |
| ResourceRecType   Type                                                  | Type of resource record (see Table 30, “Resource Record types”, on page 213).                                                                             |
| ResourceClass   Class                                                   | Class of resource record (for example, IN, CS, CH, HS).                                                                                                   |
| MinimumTTL   TTL                                                        | TTL Value of the resource record. Note that a value of -1 indicates that no TTL has been specified.                                                       |
| ResourceRecText   Text                                                  | Data area of the resource record. The data content varies depending on the ResourceRecType.                                                               |
| ApplyToZone   Zone                                                      | Zone where the record should be entered:<br>1   Forward   FWD = Forward zone<br>0   Reverse   REV = Reverse zone<br>Only applies to owner_type of object. |

| Field Name/Label    | Description                                                                                                                                                                                                                                                                                  |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ChangeFlag   Change | 1   Add = Add new resource record<br>2   Delete   Del = Delete existing resource record<br>If the input file contains<br>ChangeFlag=<br>(with no value following the equal sign) then the record is skipped. If the<br>input record does not mention ChangeFlag at all, the record is added. |
| External Flag       | Indicates if the record is external. 1 = External; 0=Not external.                                                                                                                                                                                                                           |
| Tombstoned          | Indicates if the record is tombstoned. 1=Tombstoned; 0=Not tombstoned.                                                                                                                                                                                                                       |
| ExternalComment     | Traces how the record entered VitalQIP. This field is for informational purposes only and should not be used.                                                                                                                                                                                |
| ExternalTimeStamp   | The actual date and time at which the record was last updated by external updates. This field is for informational purposes only and should not be used.                                                                                                                                     |



## qip-setdomainfolder

---

**qip-setdomainfolder** assigns a domain or a list of domains to a specific folder.

### Synopsis

```
qip-setdomainfolder -n folderName -d domain_name -f input_filename
[-g loginserver] [-s servername] [-u username] [-p password]
[-o organization] [-r reject_file] [-e errmsg_file]
```

### Parameters

**qip-setdomainfolder** recognizes the following parameters:

- n *folderName*        Specifies the name of an existing folder the domain(s) is to be assigned to.
- d *domain\_name*       Specifies the fully-qualified domain name.
- f *input\_filename*   This file consists of the list of domain names to be assigned to the folder.
- g *loginserver*       Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*       Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*         Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*         Specifies the password for the associated administrator account.
- o *organization*     Specifies the VitalQIP organization (corporation) name.
- r *reject\_file*       Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file*       Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Command line input example

The file specified in the **-f** parameter contains a list of domains to assign to a folder.

```
qip-setdomainfolder -n folder1 -f domain.txt
```

**Input file Content Example**

home.com  
east.com  
west.com



## qip-setlocation

---

**qip-setlocation** allows you to add a new location, or to modify an existing location within the VitalQIP database. If you want to modify existing information, you can retrieve the existing location information by using the **qip-getloctlst** command.

### Synopsis

```
qip-setlocation -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setlocation** recognizes the following parameters:

- f *input\_filename* Specifies the file that contains the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- r *reject\_file* Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Command line input example

```
qip-setlocation -u qipman -p passwd -f location.lst
```

### Input file content example

For a description of the fields in the input file example, refer to Table 35.

```
Street1=10
Street2=Valley Parkway
```

City=Malvern  
State=PA  
Zip=19355  
Country=USA

**Table 35** Input file field definitions

| <b>Field</b> | <b>Description</b>                 |
|--------------|------------------------------------|
| Street1      | The Street1 field of the location. |
| Street2      | The Street2 field of the location. |
| City         | The City of the location.          |
| State        | The State of the location.         |
| Zip          | Zip of the location.               |
| Country      | Country of the location.           |



## qip-setmacpools

---

**qip-setmacpools** configures the global and subnet DHCP Server MAC pools, for both the inclusion and exclusion pools. **qip-setmacpools** replaces all the existing macpools with the ones specified in the input file.

### Synopsis

```
qip-setmacpools [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] [-f filename] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**qip-setmacpools** recognizes the following parameters:

- g *loginserver*      Specifies the login server associated with the MAC Pools.
- s *servername*      Specifies the server name where the MAC Pool is located.
- u *username*        Specifies the username of the MAC Pool administrator.
- p *password*        Specifies the password of the MAC Pool administrator.
- o *organization*    Specifies the organization with which the MAC Pool is associated.
- f *filename*        Specifies the filename that contains the output from this CLI.
- r *reject\_file*     Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file*     Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Input file content example

```
DHCP Server Name=testdhcp.qtek.com
E, AX25, 112233445566
E, IEEE802, 00000c1703ea
I, Token Ring, 1000023abcde
I, Arcnet, 1234123412341234

Subnet Address=200.100.10.1
I, Chaos, 321321*
E, Pronet, aabbccddeeff
```

I,Ethernet,1233211233211234  
E,Token Ring,222\*



## qip-setnmdnsserver

---

**qip-setnmdnsserver** sets the attributes of a non-managed DNS server in the specified filename (**-f** parameter).

### Synopsis

```
qip-setnmdnsserver -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setnmdnsserver** recognizes the following parameters:

- f *input\_filename* Specifies the file that contains the input data.  
e
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- r *reject\_file* Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Command line input example

```
qip-setnmdnsserver -f myinputfile
```

### Input file content example

The first three fields are required in every set of input fields.

```
ServerName=qipdnsserver.lucent.com
NMServerName=nmdnsserver.lucent.com
```

NMServerAddress=1.2.3.4

**Important!** You must specify a full set of fields for *either* Domain Information or Reverse Zone Address information. You cannot specify both DomainName and ReverseZoneAddress in the same set of fields, even though the remaining fields in the set are identical.

You can specify a full set of fields for Domain Information and immediately follow it with a full set of fields for Reverse Zone Address information.

The *first* set of fields from **DomainName** to **AlsoNotify** are for Domain Information.

```
DomainName=lucent.com
RefreshTime=21600
ExpirationTime=604800
RetryPeriod=3600
MinimumTTL=86400
MaxTransferTimeIn=0
ZoneMail=email@lucent.com
CheckNames=Warning
AllowUpdate=any
AllowQuery=any
AllowTransfer=any
Notify=No
MSAllowUpdate=0
Aging=0
AgeRefreshHours=0
AgeNoRefreshHours=0
AlsoNotify=
```

The *second* set of fields from **ReverseZoneAddress** to **AlsoNotify** are for Reverse Zone Information.

```
ReverseZoneAddress= 135.114.105.0/24
RefreshTime=21600
ExpirationTime=604800
RetryPeriod=3600
MinimumTTL=86400
MaxTransferTimeIn=0
ZoneMail=email@lucent.com
CheckNames=Warning
AllowUpdate=any
AllowQuery=any
AllowTransfer=any
Notify=No
MSAllowUpdate=0
Aging=0
AgeRefreshHours=0
AgeNoRefreshHours=0
AlsoNotify=
```



## qip-setobject

---

**qip-setobject** adds or modifies an object or objects. To use this CLI command effectively, first use **qip-getobjectprof** to obtain the Object Profile data; then modify it for the target object.

**Important!** VitalQIP offers a `-t` parameter, which must be used if you only want to *modify a few attributes of the object*, rather than overwriting the entire object. Using the `-t` parameter allows you to update a portion of the record, leaving the rest of the record intact. If you omit the `-t` parameter, the fields you supply replace the *entire* record, and the resulting record contains *only* those fields. Therefore, if you use the `-t` option with this CLI command (to modify Contact or Location information), the individual detail fields for the specified Contact or Location are updated with the supplied values.

With this CLI command, you can modify Contact or Location information by using one of the following three methods:

- Specify the LocationID or ContactID – a number that points to a specific set of information regarding the Location or Contact to be applied to the object. For Location information, the fields are Street1, Street2, City, State, Zip, and Country. For Contact information, the fields are ContactLastName, ContactFirstName, ContactEMail, ContactPhone, and ContactPager. In the following paragraphs, these fields are referred to as the “detail fields”.
- In this method, specify the detail fields instead of specifying the LocationID or ContactID. VitalQIP finds an exact match and supplies the corresponding LocationID or ContactID. If VitalQIP does not find an exact match, it creates a new LocationID or ContactID and associates it with the object.
- In this method, you specify the ID AND any of the detail fields. If they differ from the match that VitalQIP finds, VitalQIP uses the ID field and UPDATES the detail fields with the information you provided, replacing the fields that were originally associated with the ID.

The format file (`-f format_file`) and the rejection file (`-r reject_file`) are also available for this CLI. Any records that are not successfully added or modified are placed in the rejection file (`-r reject_file`). If your data file is in CSV format, specify the format file (`-f format_file`).

**qip-setobject** converts a data file into the correct data file format, according to the *format\_file*. This structure allows users to import data from any format, as long as the example is followed.

### A few things to keep in mind

- The fields can be in any order – with the exception of `ObjectAddress` – **which must be the first field**. Any field not in the data file is represented as a NULL value.

### Synopsis

```
qip-setobject [-t] [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -d data_file [-f format_file]
 [-r reject_file] [-e errmsg_file] [-x]
```

## Parameters

**qip-setobject** recognizes the following parameters:

- t Sets the flag to modify the object.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- d *data\_file* Specifies the file where the information is stored. This parameter is mandatory.
- f *format\_file* Specifies the format filename. Refer to Table 36 for a list of valid field names. "ObjectAddress" must exist, and must be first. Separate field names with a comma.
- r *reject\_file* Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.
- x Allows the selected address to be overwritten.

### Example 1

```
qip-setobject -u qipman -p passwd -d setobj.txt -f setobj.fmt -r setobj.rej
```

### Command line input example 1

Contents of the -d *setobj.txt* (data) file:

```
144.144.144.5, host1
123.123.123.123, ws2
```

Contents of the -f *setobj.fmt* (format) file:

```
ObjectAddress, ObjectName (other field names, as described in Table 36)
```

**Example 2**

```
qip-setobject -u qipman -p passwd -d objprof.txt -r setobj.rej
```

**Command line input example 2**

Contents of the **-d objprof.txt** (data) file:

```
ObjectAddress=199.200.139.28
SubnetAddress=199.200.138.0
ObjectName=hostxyz
DomainName=qtek.com
ObjectClass=PC
Application=PAYROLL
(other field names, as described in Table 36)
```

**Input file content example**

For a description of the fields in the input file example, refer to Table 36 following.

**Important!** For more information on objects, refer to “Manage Objects in a Subnet” in Chapter 4 of the *VitalQIP User’s Guide*.

```
ObjectAddress=144.144.144.5
SubnetAddress=144.144.144.0
ObjectName=ws2
DomainName=quadritek.com
ObjectClass=PC
ExpiredDate=
ServerType=None
Application=
MACAddress=000a0c112233
ObjectTag=abc
LocationID=22
RoomID=
Street1=
Street2=
City=
State=
Zip=
Country=
Manufacturer=abc corp
ModelType=T1
SerialNo=008899
AssetNo=
HostID=
PurchaseDate=02/01/1998 10:00
ObjectDescription=test station
HubName=Hob21.qtek.com
SlotName=slot-21
PortNum=1
ContactID=10
```

```

ContactLastName=
ContactFirstName=
ContactEmail=
ContactPhone=
ContactPager=
RouterGroup=
DynamicConfig=A-BootP
TftpServer=
BootFileName=
HardwareType=Ethernet
Aliases=Workstati-nws2
MailForwarders=
MailHosts=mailHost.qtek.com
HubSlots=
DNSServers=
TimeServers=
DefaultRouters=144.144.144.1
NameService=A,PTR
DynamicDNSUpdate=A,PTR,CNAME,MX
DHCPServer=dhcp1.qtek.com
DHCPOptionTemplate=general
DHCPPolicyTemplate=
LeaseTime=-1
VendorClass=
TTLtime=100
DualProtocol=
Users=
UserClasses=MS Vendor
UsageBillServices=On
UsageBillLocation=malvern
UsageBilluserGroup=qtek
UsageBillObjectClass=Server
Tombstoned=1
ExternalComment="Comment"
ExternalTimestamp=03/09/2005 20:37:42
AllowDHCPClientsModifyDynamicObjectResourceRecords=True

```

**Table 36 Input file field definitions**

| Field name/label | Description                                                                                                                     |
|------------------|---------------------------------------------------------------------------------------------------------------------------------|
| ObjectAddress    | <b>Required.</b> The IP Address of the object.                                                                                  |
| SubnetAddress    | <i>This field is currently ignored and may be utilized in future releases. Subnets are retrieved from the Object's Address.</i> |
| ObjectName       | <b>Required.</b> The hostname of the object. Not fully-qualified.                                                               |

| Field name/label                                                                                                                                                                                   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DomainName                                                                                                                                                                                         | <b>Required.</b> The domain of the object. If no value is provided, the Domain Name is identified as the default domain of the subnet the Object is on. If you choose not to identify a Domain Name, enter <b>None</b> (for example, DomainName=None).                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| ObjectClass                                                                                                                                                                                        | The object class of the object:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
|                                                                                                                                                                                                    | <table border="0"> <tr> <td style="vertical-align: top;"> <ul style="list-style-type: none"> <li>- Workstation</li> <li>- X-terminal</li> <li>- PC</li> <li>- Printer</li> <li>- Server</li> <li>- Wiring_HUB</li> <li>- Router</li> <li>- Bridge</li> </ul> </td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> <li>- Terminal_Server</li> <li>- Switch</li> <li>- Legacy_System</li> <li>- Gateway</li> <li>- Test_Equipment</li> <li>- Undefined</li> <li>- Others</li> <li>-Any user-defined object class name</li> </ul> </td> </tr> </table> | <ul style="list-style-type: none"> <li>- Workstation</li> <li>- X-terminal</li> <li>- PC</li> <li>- Printer</li> <li>- Server</li> <li>- Wiring_HUB</li> <li>- Router</li> <li>- Bridge</li> </ul> | <ul style="list-style-type: none"> <li>- Terminal_Server</li> <li>- Switch</li> <li>- Legacy_System</li> <li>- Gateway</li> <li>- Test_Equipment</li> <li>- Undefined</li> <li>- Others</li> <li>-Any user-defined object class name</li> </ul> |
| <ul style="list-style-type: none"> <li>- Workstation</li> <li>- X-terminal</li> <li>- PC</li> <li>- Printer</li> <li>- Server</li> <li>- Wiring_HUB</li> <li>- Router</li> <li>- Bridge</li> </ul> | <ul style="list-style-type: none"> <li>- Terminal_Server</li> <li>- Switch</li> <li>- Legacy_System</li> <li>- Gateway</li> <li>- Test_Equipment</li> <li>- Undefined</li> <li>- Others</li> <li>-Any user-defined object class name</li> </ul>                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| ExpiredDate                                                                                                                                                                                        | The date when a reserved object expires and is no longer reserved.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| ServerType                                                                                                                                                                                         | Server type. Only applicable if ObjectClass is "server":<br><ul style="list-style-type: none"> <li>- TimeServer – If object is a time server.</li> <li>- TftpServer – If object is a Tftp server.</li> <li>- TftpServer&amp;TimeServer – If object is a TftpServer and TimeServer</li> </ul>                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Application                                                                                                                                                                                        | The 'Primary Application' of the object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| MACAddress                                                                                                                                                                                         | The Mac Address of Object.<br><b>Important!</b> Exclude the colons (:) when entering a MAC address.                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| ObjectTag                                                                                                                                                                                          | Tag field                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| RoomID                                                                                                                                                                                             | Room ID field                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| LocationID                                                                                                                                                                                         | The Location ID. If no Location ID is specified, an attempt is made to match the record EXACTLY with the Street1, Street2, City, State, Zip and/or Country information that is supplied. If a match is not found, a new record is added without an ID. If the Location ID is specified, the information provided updates the current record using the -t option (refer to the note at the beginning of this CLI command).                                                                                                                                                 |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Street1                                                                                                                                                                                            | Street1 field of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Street2                                                                                                                                                                                            | Street2 field of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| City                                                                                                                                                                                               | City of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| State                                                                                                                                                                                              | State of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Zip                                                                                                                                                                                                | Zip of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Country                                                                                                                                                                                            | Country of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |
| Manufacturer                                                                                                                                                                                       | Manufacturer Name                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                    |                                                                                                                                                                                                                                                 |

| Field name/label  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ModelType         | Model Name                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| SerialNo          | Serial Number of Object                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| AssetNo           | Asset Number of object                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| HostID            | Host ID of object                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| PurchaseDate      | Purchase Date/Time of object. Must be in "mm/dd/yyyy HH:MM" format (for example, 09/22/1998 00:00).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| ObjectDescription | Object Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| HubName           | Hub Name(s) assigned to object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| SlotName          | Slot Name(s) assigned to object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| PortNum           | Port Numbers assigned to object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| ContactID         | <p>The Contact ID. If no Contact ID is specified, an attempt is made to match the record EXACTLY with the ContactLastName and ContactFirstName information that is supplied. If no match is found, a new record is added without an ID. If the Contact ID is specified, the information provided updates the current record using the -t option (refer to the note at the beginning of this CLI command).</p> <p><b>Important!</b> If you have multiple users with the same name (first and last names) and you want to enter that same name for another individual, this field value should be 0. This tells the system to add the exact same name.</p> |
| ContactLastName   | Contact last name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| ContactFirstName  | Contact first name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| ContactEmail      | Contact email address.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| ContactPhone      | Contact phone number.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| ContactPager      | Contact pager number.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| RouterGroup       | Router Group of object. Only applicable if the object class is "router".                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| DECNetArea        | DECNet Area. (Valid values are 1-63) Only applicable if Dual Protocol is set to DECNet. The DECNet Address is calculated on the input in the DECNetArea and the DECNet Node.                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| DECNetNode        | DECNet Address. (Valid values are 1-1024) Only applicable if Dual Protocol set to DECNet. The DECNet Address is calculated on the input in the DECNetArea and the DECNet Node.                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| IPXNetworkNumber  | IPX Network number. Only applicable if Dual Protocol is set to IPX.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| IPXNode           | IPX Node. Only applicable if Dual Protocol is set to IPX.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

| <b>Field name/label</b> | <b>Description</b>                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DynamicConfig           | Dynamic Configuration of Object:<br><ul style="list-style-type: none"> <li>- Blank – Static object</li> <li>- None – Dynamic (none)</li> <li>- M-DHCP – Manual DHCP</li> <li>- A-DHCP – Automatic DHCP</li> <li>- D-DHCP – Dynamic DHCP</li> <li>- M-BOOTP – Manual Bootp</li> <li>- A-BOOTP – Automatic Bootp</li> <li>- Reserved - To reserve an object (the IP must be unused).</li> </ul> |
| TftpServer              | TFTP Server of object. Only applicable if object type is M-BOOTP.                                                                                                                                                                                                                                                                                                                             |
| BootFileName            | Bootfile name of object. Only applicable if object type is M-BOOTP.                                                                                                                                                                                                                                                                                                                           |
| HardwareType            | Hardware Type of object. Only applicable for M-BOOTP objects.<br><ul style="list-style-type: none"> <li>- Ethernet</li> <li>- TokenRing</li> <li>- AX.25</li> <li>- Pronet</li> <li>- Chaos</li> <li>- IEEE802</li> <li>- Arcnet</li> </ul>                                                                                                                                                   |
| Aliases*                | Alias names. If multiple, separate by space (for example, www www3).                                                                                                                                                                                                                                                                                                                          |
| MailForwarders*         | Fully-qualified host name with priority (for example, mailF.qtek.com(10) ).                                                                                                                                                                                                                                                                                                                   |
| MailHosts*              | Fully-qualified host name with priority (for example, mailH.qtek.com(100) ).                                                                                                                                                                                                                                                                                                                  |
| Users*                  | User login names. To delete a user place a '(d)' after the login name (for example, admin1(d)).                                                                                                                                                                                                                                                                                               |
| UserClasses             | Specific categories of user options. To enter one or more user classes for this object, enclose a list of users in double quotes (for example, "user1,user2,user3").                                                                                                                                                                                                                          |
| HubSlots*               | List of slot names (port numbers) if the object is a Wiring_HUB.                                                                                                                                                                                                                                                                                                                              |
| DNSServers*             | DNS Servers attached to this object. If multiple, separate by space. Only applicable for M-BOOTP objects.                                                                                                                                                                                                                                                                                     |
| TimeServers*            | Time Servers attached to this object. If multiple, separate by space. Only applicable for M-BOOTP objects.                                                                                                                                                                                                                                                                                    |
| DefaultRouters          | IP Address(es) of Default routers, separated by space if multiples.                                                                                                                                                                                                                                                                                                                           |

| Field name/label                                                        | Description                                                                                                                                                                                                                                                                                                                                                           |
|-------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NameService                                                             | Name Services selection:<br>- No – Name Services checked off.<br>- Yes – Name Services checked on.<br>- A – Name Services checked on, A record checked on.<br>- PTR – Name Services checked on, PTR record checked on.<br>- A PTR – Name Services checked on, A record checked on, PTR record checked on.                                                             |
| DynamicDNSUpdate                                                        | Dynamic Update selection:<br>- A – Dynamic updates for A records.<br>- PTR – Dynamic updates for PTR records.<br>- CNAME – Dynamic updates for CNAME records.<br>- MX – Dynamic updates for MX records.<br><br><b>Important!</b> Multiple selections can be accomplished by specifying each component, separated by a space as a delimiter (for example, A PTR CNAME) |
| DHCPsServer                                                             | The DHCP Server assigned to this object. Only applicable for dynamic objects.                                                                                                                                                                                                                                                                                         |
| DHCPOptionTemplate                                                      | The DHCP Option Template assigned to this object. Only applicable for dynamic objects.                                                                                                                                                                                                                                                                                |
| DHCPPolicyTemplate                                                      | The DHCP Policy template assigned to this object. Only applicable for dynamic objects.                                                                                                                                                                                                                                                                                |
| LeaseTime                                                               | The lease time (in seconds assigned) to this object. Only applicable for dynamic objects.                                                                                                                                                                                                                                                                             |
| VendorClass                                                             | The Vendor Class assigned to this object. Only applicable for dynamic objects.                                                                                                                                                                                                                                                                                        |
| TTLtime                                                                 | Time to live (TTL) value, in seconds.<br><br><b>Important!</b> Since 0 is a valid value, a value of –1 indicates no TTL set.                                                                                                                                                                                                                                          |
| NetbiosName                                                             | NetBIOS Name. Only applicable when dual protocol is set to NETBIOS.                                                                                                                                                                                                                                                                                                   |
| NetbiosDomain                                                           | NetBIOS Domain. Only applicable when dual protocol set to NETBIOS.                                                                                                                                                                                                                                                                                                    |
| DualProtocol                                                            | Dual Protocol of object:<br>- None<br>- DECNet                                                                                                                                                                                                                                                                                                                        |
| UsageBillServices                                                       | Determines whether the Usage Billing service is on or off.                                                                                                                                                                                                                                                                                                            |
| UsageBillLocation                                                       | Location name of the Usage Billing Service.                                                                                                                                                                                                                                                                                                                           |
| UsageBillUserGroup                                                      | User group of the Usage Billing Service.                                                                                                                                                                                                                                                                                                                              |
| UsageBillObjectClass                                                    | Object class of the Usage Billing Service.                                                                                                                                                                                                                                                                                                                            |
| <b>Important!</b> The following fields appear once per resource record. |                                                                                                                                                                                                                                                                                                                                                                       |

| Field name/label                                   | Description                                                                                                         |
|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Tombstoned                                         | [O] 1 indicates that this is a tombstoned external record. 0 indicates that it is not a tombstoned external record. |
| ExternalComment                                    | [O] text (up to 255 characters), a comment indicating the history of this external add.                             |
| ExternalTimestamp                                  | [O] numeric. Specifies the actual date and time.                                                                    |
| AllowDHCPClientsModifyDynamicObjectResourceRecords | True<br>False<br>Same as in Global Policies                                                                         |

**Important!** Fields marked with an asterisk indicate that you can add multiples on one line, separated by a space, as well as on separate lines in normal Field Name=Value format. For example, if there are multiple aliases, they can be listed as:

```
Alias=john sam lynn brian
```

*or*

```
Alias=john
Alias=sam
Alias=lynn
Alias=brian
```



## qip-setospfprof

---

**qip-setospfprof** sets the profile of an OSPF area, based on the data provided in the input file specified in the `-f` parameter.

### Synopsis

```
qip-setospfprof -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setospfprof** recognizes the following parameters:

- `-f input_filename` Specifies the file that contains the input data.
- `-g loginserver` Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- `-s servername` Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- `-u username` Specifies the VitalQIP administrator account to be used in establishing the database connection.
- `-p password` Specifies the password for the associated administrator account.
- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-r reject_file` Specifies the file where records that are not added or modified are placed.
- `-e errmsg_file` Specifies filename for error messages. The default is *STDERR*.

### Command line input example

```
qip-setospfprof -u qipman -p passwd -f ospfprof.txt
```

### Input file content example

**Important!** The Subnet Address must appear before the information associated with that managed subnet. For a description of the fields in this input example and more information on objects, refer to "Define an OSPF Area" in Chapter 3 of the *VitalQIP User's Guide*.

```
OspfAreaName=OSPF1
OspfAreaID=0.0.48.58
WarningPercent=90
WarningType=1
OspfAddressRange=198.200.138.000 198.200.138.031 255.255.255.224;
```



## qip-setpolicy

---

**qip-setpolicy** adds or modifies a policy entry for the enterprise server. For more information on policies, refer to Chapter 2, “Policies and profiles” in the *VitalQIP User’s Guide*.

### Synopsis

```
qip-setpolicy -c class -n policy -d value [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
[-r reject_file] [-e errmsg_file] [-a new_data_type]
```

### Parameters

**qip-setpolicy** recognizes the following parameters:

- |                        |                                                                                                                   |
|------------------------|-------------------------------------------------------------------------------------------------------------------|
| -c <i>class</i>        | Specifies the class of policy you want to modify or add (for example, Billing).                                   |
| -n <i>policy</i>       | Specifies the policy you want to add or modify (for example, PING_ATTEMPTS and ALWAYS_APPEND_ROUTER).             |
| -d <i>value</i>        | Specifies the value you want to assign to the policy. Alphabetic values must be in uppercase letters.             |
| -g <i>loginserver</i>  | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i>   | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| -u <i>username</i>     | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| -p <i>password</i>     | Specifies the password for the associated administrator account.                                                  |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name.                                                           |
| -r <i>reject_file</i>  | Specifies the file where records that were not added or modified are placed.                                      |
| -e <i>errmsg_file</i>  | Specifies the filename where error messages are written, if they occur. The default is <i>STDERR</i> .            |

-a *new\_data\_type* Specifies the new data type. Values are as follows:

|                      |            |                  |
|----------------------|------------|------------------|
| boolean              | True/False | time_of_day_list |
| ip_address           |            | numeric_list     |
| ip_address_list      |            | Delete           |
| ip_address_mask_list |            | numeric_signed   |
| ip_address_pair_list |            | password         |
| ip_mask              |            | text             |
| multi-lined_text     |            | text_list        |
| numeric              |            | time_interval    |

### Command line input example

- To add or modify Billing class policy information for the enterprise server:

```
qip-setpolicy -s QIPSYBASE -u qipman -p passwd -c Billing -n Required_Data
-d "User Group"
```

- To add a new policy to the Dynamic DNS class:

```
qip-setpolicy -c "Dynamic DNS" -n NewPolicy -a Text -d "this is my new policy"
```



## qip-setreclaimschedule

---

**qip-setreclaimschedule** sets or resets the subnet reclaim schedule using the parameters given in the input file.

### Synopsis

```
qip-setreclaimschedule -f input_filename [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization] [-b]
```

### Parameters

**qip-setreclaimschedule** recognizes the following parameters:

- f *input\_filename* Specifies the input file name, which contains the schedule information
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection
- o *organization* Specifies the VitalQIP organization (corporation) name
- p *password* Specifies the password for the associated administrator account
- b If a reclaim schedule for a given subnet exists, the user needs to use "--b" option to overwrite the existing schedule. When -b is used, the existing scheduled reclaim is aborted, a report generated and emailed to the administrator who setup the original schedule.

### Command line input example 1

```
SubnetAddress=144.144.1.0
Reclaim Type=Report Only
Reclaim Email Addresses=root@test.com
Data Collection (Ping) Schedule=By Day
Time of Day=36000
Time of Day=54000
Total Number Of Days=5put Example1
SubnetAddress=144.144.1.0
Reclaim Type=Report Only
Reclaim Email Address=root@test.com
```

```
Data Collection(Ping) Schedule=By Day
Time of Day=36000
Time of Day=54000
Total Number Of Days=5
```

### Command line input example 2

```
SubnetAddress=144.144.1.0
Reclaim Type=Report Only
Reclaim Email Address=root@test.com
Data Collection(Ping) Schedule=By Interval
Time Interval=1 day 20 hours
Total Number Of Times=5
SubnetAddress=144.144.0.0
Reclaim Type=Report Only
Reclaim Email Address=root@test.com
Data Collection(Ping) Schedule=By Day
Time of Day=10:00
Time of Day=20:00
Total Number Of Days=5
```

### Output example

The “Automatic Reclaim Report” is generated only for an existing scheduled reclaim when **-b** option is used. It is generated using the objects statistics accumulated thus far, with a note stating the reason for the abort, and sent to the email address of the administrator who scheduled the reclaim. The report, which has the same format as the report generated by **qip-reclaim** CLI, is also placed in *\$QIPHOME/report* named with the date/time stamp and subnet address for uniqueness.

```
<<<<< AUTOMATIC RECLAIM REPORT >>>>>
```

```
<<<<< Please Note: >>>>>
```

```
Scheduled Reclaim Aborted due to Schedule Reset by : qipman ()
```

```
Report Date: 11/19/2003 14:45 Start Date: 11/19/2003 14:18
```

```
Collection Frequency: Collect: Every Day At (09:03) Total: 1 Days
```

| Last Time<br>Address<br>Status | Reachable | Name                               | Object<br>Class | Object<br>Status | #<br>Try | #<br>In | #<br>DNS | Reclaim<br>Reachable |
|--------------------------------|-----------|------------------------------------|-----------------|------------------|----------|---------|----------|----------------------|
| 10.200.90.1<br>0               | 0         | rtp000006rts.seg4.qa.quadritek.com | Router          |                  |          |         | Static   | 0                    |
| 10.200.90.2<br>0               | 0         |                                    |                 |                  |          |         | Unused   | 0                    |
| 10.200.90.3<br>0               | 0         |                                    |                 |                  |          |         | Unused   | 0                    |
| 10.200.90.4<br>0               | 0         |                                    |                 |                  |          |         | Unused   | 0                    |
| 10.200.90.5<br>0               | 0         | wsp000074WSS.seg4.qa.quadritek.com | Workstation     |                  |          |         | Static   | 0                    |

10.200.90.6  
0 0  
10.200.90.7  
0 0

Unused 0  
Unused 0



## qip-setsnorgprof

---

**qip-setsnorgprof** establishes the profile of a subnet organization based on the data provided in the input file specified in the **-f** parameter.

### Synopsis

```
qip-setsnorgprof -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setsnorgprof** recognizes the following parameters:

- f *input\_filename* Specifies the file that contains the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- r *reject\_file* Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

### Command line input example

```
qip-setsnorgprof -u qipman -p passwd -f subnetorg.txt
```

### Input file content example

For a description of the fields in the input file example, refer to Table 37.

```
SubnetOrgName=snorg1
WarningPercent=95
WarningType=1
```

```

AssignGap=Workstation,1
AssignGap=PC,2
SubnetAddress(net,mask,addr)=144.144.0.0,255.255.255.0,144.144.1.0
SubnetAddress(net,mask,addr)=144.144.0.0,255.255.255.0,144.144.2.0
W2KSiteName=site1
UseSubnetOrgName=false
AssocControllers=dc1.lucent.com
DhcpServer=dhcp1.lucent.com
DhcpOptionTemplate=general

```

**Table 37 Input file field definition**

| Field name/label             | Description                                                                                                                                                                                             |
|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SubnetOrgName                | The Subnet Organization name.                                                                                                                                                                           |
| WarningPercent               | The Warning for the percentage of Managed Addresses, (for example, 10 equals 10 percent).                                                                                                               |
| WarningType                  | Warning Type:<br>0 – No warning<br>1 – Email only<br>2 – Visual only<br>3 – Both email and visual warning                                                                                               |
| AssignGap                    | The object class of the GAP object defined. The relative offset. Note that positive number (>0) indicates "from the beginning" of the subnet. A negative number indicates "from the end" of the subnet. |
| SubnetAddress(net,mask,addr) | The subnet address, network address, and subnet mask of the subnet.                                                                                                                                     |
| W2KSiteName                  | The site name of the Windows 2000 site. If blank, the subnet organization name is used. (Optional)                                                                                                      |
| AssocControllers             | List of Windows 2000 controllers.                                                                                                                                                                       |
| DHCPsServer                  | The default DHCP server for this subnet.                                                                                                                                                                |
| DHCPOptionTemplate           | The default DHCP Option Template for this subnet.                                                                                                                                                       |
| UseSubnetOrgName             | Specifies if the subnet organization name is used.                                                                                                                                                      |



## qip-setsubnet

---

**qip-setsubnet** adds or modifies Subnet Profiles, based on the data provided in the input file specified in the `-f` parameter. To effectively use this CLI command, first use **qip-getsubnetprof** to obtain the Subnet Profile data; then modify it as input. The fields can be in any order except for the first field, which must be the subnet address.

**Important!** VitalQIP offers a `-t` parameter, which must be used if you only want to *modify a few attributes of the subnet*, rather than overwriting the entire profile. Using the `-t` parameter allows you to update a portion of the record, leaving the rest of the record intact. If you omit the `-t` parameter, the fields you supply replaces the *entire* record, and the resulting record contains *only* those fields. Therefore, if you use the `-t` option with this CLI command (to modify Contact or Location information), the individual detail fields for the specified Contact or Location are updated with the supplied values.

With this CLI command, you can modify Contact and Location information by using one of the following three methods:

- Specify the **LocationID** or **ContactID** – a number that points to a specific set of information regarding the Location or Contact to be applied to the subnet. For Location information, the fields are **Street1**, **Street2**, **City**, **State**, **Zip**, and **Country**. For Contact information, the fields are **ContactLastName**, **ContactFirstName**, **ContactEMail**, **ContactPhone**, and **ContactPager**. In the following paragraphs, these fields are referred to as the “detail fields”.
- In this method, *instead of* specifying the **LocationID** or **ContactID**, you specify the detail fields. VitalQIP finds an exact match and supply the corresponding **LocationID** or **ContactID**.
- In this method, you specify the ID *AND* the detail fields. If they differ from the match that VitalQIP finds, VitalQIP uses only the ID field; it ignores your specified detail fields.

### Before you begin

- The fields can be in any order with the exception of SubnetAddress - **which must be the first line**. Any field not in the data file is represented as a NULL value.
- To add a new subnet, the fields NetworkAddress= and SubnetMask= are mandatory. The network must already have been defined in the system

### Synopsis

```
qip-setsubnet -f input_filename [-t] [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setsubnet** recognizes the following parameters:

`-f input_filename` Specifies the file that contains the input data.

- t                        Sets the flag to modify the subnet profile.
  
- g *loginserver*        Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*         Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*            Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*            Specifies the password for the associated administrator account.
  
- o *organization*       Specifies the VitalQIP organization (corporation) name.
  
- r *reject\_file*         Specifies the file where records that are not added or modified are placed.
  
- e *errmsg\_file*         Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

**Command line input example**

```
qip-setsubnet -u qipman -p passwd -f subnet.txt
```

**Input File Content Example**

For a description of the fields in the input file example, refer to Table 38.

```
SubnetAddress=144.144.144.0
SubnetName=SN144_144
SubnetMask=255.255.255.0
NetworkAddress=144.144.0.0
LocationID=2
ContactID=2
ContactLastName=pal
ContactFirstName=pal
ContactEmail=pal@qtek.com
ContactPhone=111-0033
ContactPager=1-800-111-3300
Application=
Domain=quadritek.com
TftpServer=
ShowUsage=N
CheckUsage=N
SubnetDescription=
```

```

SharedNetwork=
HardwareType=ethernet
WarningType=Visual
WarningPercent=80
DNSServers=pallas.quadritek.com
TimeServers=
DefaultRouters=
DHCPServer=dhcp.quadritek.com
DHCPOptionTemplate=general
DHCPPolicyTemplate=
PrimaryInterface=No
UsageBillLocation=malvern
UsageBilluserGroup=qtek
AllowDHCPClientsModifyDynamicObjectResourceRecords=True

```

**Table 38** Input file field definition

| Field          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SubnetAddress  | <b>Required.</b> The subnet address. Must be the first line.                                                                                                                                                                                                                                                                                                                                                                                                               |
| SubnetName     | The subnet name.                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| SubnetMask     | <b>Required when adding a subnet.</b> Subnet mask of the subnet.                                                                                                                                                                                                                                                                                                                                                                                                           |
| NetworkAddress | The network address of this subnet. The network must already be defined within VitalQIP. Required when adding a subnet.                                                                                                                                                                                                                                                                                                                                                    |
| LocationID     | The Location ID can be found by running <b>qip-getloclst</b> . If no Location ID is specified, the record is attempted to be matched EXACTLY with the Street1, Street2, City, State, Zip and/or Country information that is supplied. If it is not matched, a new record is added without an ID. If the Location ID is specified, the information provided updates the current record using the <b>-t</b> option (refer to the note at the beginning of this CLI command). |
| Street1        | Street1 field of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Street2        | Street2 field of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| City           | City of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| State          | State of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Zip            | Zip of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Country        | Country of the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| ContactID      | The Contact ID can be found by running <b>qip-getcontactlst</b> . If no Contact ID is specified, the record is attempted to be matched EXACTLY with the ContactLastName and ContactFirstName information that is supplied. If it is not matched, a new record is added without an ID. If the Contact ID is specified, the information provided updates the current record using the <b>-t</b> option (refer to the note at the beginning of this CLI command).             |

| <b>Field</b>      | <b>Description</b>                                                                                             |
|-------------------|----------------------------------------------------------------------------------------------------------------|
| ContactLastName   | Contact last name.                                                                                             |
| ContactFirstName  | Contact first name.                                                                                            |
| ContactEmail      | Contact email address.                                                                                         |
| ContactPhone      | Contact phone number.                                                                                          |
| ContactPager      | Contact pager number.                                                                                          |
| Application       | Primary Application assigned to this subnet.                                                                   |
| Domain            | Domain(s) assigned separated by spaces.                                                                        |
| TftpServer        | Default Tftp server assigned.                                                                                  |
| ShowUsage         | Show Used Only Flag:<br>0 – No<br>1 – Yes                                                                      |
| CheckUsage        | Check before assign: Y=Ping N=None                                                                             |
| SubnetDescription | Subnet "comment" text area.                                                                                    |
| SharedNetwork     | The Shared Network name.                                                                                       |
| HardwareType      | Default hardware type:<br>- Ethernet<br>- TokenRing<br>- AX.25<br>- Pronet<br>- Chaos<br>- IEEE802<br>- Arcnet |
| WarningType       | Warning Type:<br>0 – No warning<br>1 – Email only<br>2 – Visual only<br>3 – Both email and visual warning      |
| WarningPercent    | Threshold percentage before alarm is issued. Percent range is 0 to 99.                                         |
| DNSServers        | DNS Servers for this subnet, separated by spaces.                                                              |
| TimeServers       | Time Servers for this subnet, separated by spaces.                                                             |
| DefaultRouters    | Default Routers for this subnet, separated by spaces.                                                          |
| DHCPsServer       | The default DHCP server associated with this subnet.                                                           |

| <b>Field</b>                                               | <b>Description</b>                                            |
|------------------------------------------------------------|---------------------------------------------------------------|
| DHCPOptionTemplate                                         | The default DHCP Option Template associated with this subnet. |
| DHCPPolicyTemplate                                         | The default DHCP Policy Template associated with this subnet. |
| PrimaryInterface                                           | Primary Interface – Yes / No                                  |
| UsageBillLocation                                          | Location name of the Usage Billing Service                    |
| UsageBillUserGroup                                         | User group of the Usage Billing Service                       |
| AllowDHCPClientsModify<br>DynamicObjectResourceRe<br>cords | True<br>False<br>Same as in Global Policies                   |



## qip-setudf

---

**qip-setudf** sets the value of a pre-existing user-defined field associated with the specified class (User, Domain, Reverse\_Zone, Organization, Object, Subnet, Prov\_Block\*, or Prov\_Pool\*).

### Synopsis

```
qip-setudf [-g loginserver] [-s servername] [-o organization]
[-u username] [-p password] -c class_name -a ip_address
[-n owner_name] -f field_name [-d field_value][-r reject_file]
[-e errmsg_file]
```

### Parameters

**qip-setudf** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- c *class\_name* Specifies the class with which the user field is associated. Refer to Table 39 following.
- a *ip\_address* Specifies the IP address of the user-defined field class; reverse\_zone, object, or subnet. If the user-defined field class is Reverse\_Zone, the address is xxx.xxx.xxx.xxx/mask.
- n *owner\_name* Specifies the IP address of the class User, Domain, Organization, Prov\_Block\* or Prov\_Pool\*. If the user-defined field class is User, then it requires the user's Login ID, as defined in the User Profile. If this parameter and the -d parameter are not specified, then a new user-defined field is created.
- f *field\_name* Specifies the field name of the value to be updated or added.

- d *field\_value* Specifies the field value of the field to be updated or added. If this parameter and the -n parameter are not specified, then a new user-defined field is created.
- r *reject\_file* Specifies the file where records that are not added or modified are placed.
- e *errmsg\_file* Specifies the filename where error messages are written, if they occur. The default is *STDERR*.

**Table 39 User field classes**

| User field classes | Notes                                                       |
|--------------------|-------------------------------------------------------------|
| User               | Used for User Defined Fields for a User Profile.            |
| Domain             | Used for User Defined Fields for a Domain.                  |
| Reverse_Zone       | Used for User Defined Fields for a Reverse Zone.            |
| Organization       | Used for User Defined Fields for an Organization.           |
| Object             | Used for User Defined Fields for an Object Profile.         |
| Subnet             | Used for User Defined Fields for a Subnet Profile.          |
| Prov_Block         | Used for User Defined Fields for a Network Allocator Block. |
| Prov_Pool          | Used for User Defined Fields for a Network Allocator Pool.  |

**Important!** The classes Prov\_Block and Prov\_Pool are applicable only to the Network Allocator application.

### Command line input examples

```
qip-setudf -u qipman -p passwd -c object -a 100.34.12.12 -f "field desc" -d "this is
bob's pc"
qip-setudf -o "Test Org" -c Reverse_Zone -a 10.0.0.0/24 -f R21 -d Test
qip-setudf -n "USA Org" -c Organization -f "Managed by" -d "John Smith"
```

## qip-setuser

---

**qip-setuser** adds or modifies a VitalQIP User Profile. The information is taken from the input file specified on the command line.

### Before you begin

- If the Location ID for the user location is known, the Street1, Street2, City, State, Zip, and Country can be replaced with `loc_id=<location_ID>` (for example, `loc_id=10`).
- If multiple user-defined fields are to be entered, `UserDefinedFields=<value>` must be specified for each user-defined field.
- Multiple Group Information, Default Subnets, and Managed Range can be entered on single line, which is separated by a comma for each entry. It can also be deleted by using the following format example:

```
ManagedRange=144.144.144.2(D),144.144.144.1
```

If the (D) value is not specified, the value is assumed to be an addition to the User Profile.

### Synopsis

```
qip-setuser [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -f input_filename [-r reject_file]
 [-e errmsg_file] [-t]
```

### Parameters

**qip-setuser** recognizes the following parameters:

- |                                     |                                                                                                                   |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-g <i>loginserver</i></code>  | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code>   | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-u <i>username</i></code>     | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| <code>-p <i>password</i></code>     | Specifies the password for the associated administrator account.                                                  |
| <code>-o <i>organization</i></code> | Specifies the VitalQIP organization (corporation) name.                                                           |
| <code>-r <i>reject_file</i></code>  | Specifies the file where records that are not added or modified are placed.                                       |

- e *errmsg\_file*      Specifies the filename where error messages are written, if they occur. The default is *STDERR*.
  
- f *input\_filename*   Specifies the filename that contains the user profile data.
  
- t                      If specified, updates the entry with the supplied data. Otherwise, it replaces the entry.

### Command line input example

```
qip-setuser -u qipman -p password -f usr_prof.dat
```

### Input file content example

**Important!** For a description of the fields shown below, refer to the “Define VitalQIP users” section in Chapter 4 of the *VitalQIP User’s Guide*.

```
LoginID=pal3
LastName=test
FirstName=pal3
Phone=610-292-1312
E-mailAddress=pal3@qtek.com
Password=333333
PIN=
Description=another user with different id
Street1=5
Street2=Whitney Ave
City=New Haven
State=Ct
Zip=02488
Country=usa
ActivationStatus=1
GroupInformation="isp_qtek2","isp_qtek"
DefaultSubnets=144.144.144.0
ManagedRange=144.144.144.3,144.144.144.2
UserDefinedFields=USER1, user1 udf value for pal3
UsageBillLocation=woburn
UsageBillUserGroup=cis
```

□

## qip-setzonednsoptions

---

**qip-setzonednsoptions** supports multiple zone selection (domain and reverse zones) and specifies a set of DNS options to apply to all those zones. **Qip-setzonednsoptions** requires two input files: One that contains a list of Forward/Reverse (-z) zones, and one that contains a list of data to be changed (-f).

### Before you begin

- If the user-defined field name in the input file can be found in both the domain and reverse zone, it is implemented in both.
- The input fields can be in any order.
- Any field not in the input file is not changed.
- The value format for those fields that allow <Use List> is as follows.
  - <ACL Templates=TemplateName> means the TemplateName is added to the ACL Template list if possible (for example, AllowQueryBIND8X=ACL Templates=ACLTwo)
  - <other=ValueName> means the ValueName is added to the “other” list if possible (for example, AllowQueryBIND8X=other=extraVal)
  - <ACL Templates=TemplateNameD> means the TemplateName is deleted from the original list if possible (for example, AllowQueryBIND8X=ACL Templates=ACLTwoD)
  - \* means ALL templates are deleted from the list if possible (for example, AllowQueryBIND9X=ACL Templates=\* or AllowQueryBIND9X=other=\*)
- The value format for those fields that allow <Use List> for address lists is as follows.
  - <address> means the address is added into the original list if possible (for example, AllowTransferW2K=198.200.138.123)
  - <addressD> means the address is deleted from the original list if possible (for example, NotifyW2K=170.200.12.162D)
  - \* means ALL addresses are deleted from the list if possible (for example, NotifyW2K=\*)

### Synopsis

```
qip-setzonednsoptions [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-r reject_file] [-e errmsg_file]
-f input_filename -z zone_filename
```

## Parameters

**qip-setzonednsoptions** recognizes the following parameters:

- g *loginserver*      Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*      Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*         Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*         Specifies the password for the associated administrator account.
  
- o *organization*     Specifies the VitalQIP organization (corporation) name.
  
- r *reject\_file*       Specifies the file where records that are not added or modified are placed.
  
- e *errmsg\_file*      Specifies the filename where error messages are written, if they occur. The default is *STDERR*.
  
- f *input\_filename*   Specifies the filename that contains the DNS options.
  
- z *zone\_filename*    Specifies the file containing the list of domain names and/or reverse zone addresses.

### Zone file content example (-z)

```
a.com
b.com
c.com
170.20.0.0/16/*the reverse zone format: address/masklength*/
180.20.0.0/17
180.20.0.128/17
```

### Input file content example (-f)

```
ZoneMail=admin@company.com
RefreshTime=21600
ExpireTime=604800
MinimumTime=86400
RetryTime=3600
NegativeCacheTTL=86400
AllowUpdateW2K=Yes
AgingW2K=True
```

```

RefreshIntervalW2K=1
noRefreshIntervalW2K=1
AllowTransferW2K=200.0.0.100D
NotifyW2K=200.0.0.101,198.0.0.198
ZoneOptionW2K=/temp/filename
CheckNamesBIND8X=Fail
AllowUpdateBIND8X=other=extraVal
AllowQueryBIND8X=ACL Templates=ACLTwo,ACLone
AllowTransferBIND8X=ACL Templates=ACLTwo
NotifyBIND8X=Use Server Value
ZoneBlockBIND8X=/tmp/filename
AllowNotifyBIND9X=other=extravalue,ACL Templates=ACLTwo
AllowQueryBIND9X=ACL Templates=*,other=*
AllowTransferBIND9X=None
AllowUpdateBIND9X=None
NotifyBIND9X=Use Server Value
ZoneBlockBIND9X=/temp/filename
ZoneEdupLUCENT3=False
AllowQueryLUCENT3=None
AllowTransferLUCENT3=None
AllowUpdateLUCENT3=None
CheckNamesLUCENT3=Fail
NotifyLUCENT3=Use Server Value
ZoneBlockLUCENT3=/temp/filename
ZoneEdupLUCENT4=False
AllowNotifyLUCENT4=None
AllowQueryLUCENT4=None
AllowTransferLUCENT4=None
AllowUpdateLUCENT4=None
NotifyLUCENT4=Use Server Value
ZoneBlockLUCENT4=/temp/filenameI

```

**Table 40** Input file fields

| Field          | Description                                                                                                                                           |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| ZoneMail       | Any errors posted by DNS are sent to this email address.                                                                                              |
| ExpireTime     | In seconds. When the expire time is reached, the secondary server stops handling information about the data because the data is too old to be useful. |
| RefreshTime    | In seconds. Dictates how often the secondary server should verify its data.                                                                           |
| MinimumTime    | In seconds. Defines the time interval for other servers to cache all resource records in the database file.                                           |
| RetryTime      | In seconds. Dictates the interval for attempting to refresh in the event that the primary server is unavailable.                                      |
| AllowUpdateW2K | Values are yes, no, or Use Server Value.                                                                                                              |

| Field                      | Description                                                                                                                                                                          |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AgingW2K                   | Values are true, false, or Use Server Value.                                                                                                                                         |
| RefreshIntervalW2k         | In hours.                                                                                                                                                                            |
| NoRefreshIntervalW2K       | In hours.                                                                                                                                                                            |
| AllowTransferW2K           | Values are any, none, NameServersOnly, <Use List>, Use Server Value.                                                                                                                 |
| NotifyW2K                  | Values are yes, no, <Use List>, Use Server Value.                                                                                                                                    |
| ZoneOptionW2K              | The filename of the Windows 2000 zone option. The contents of the file is appended to the existing options.                                                                          |
| ZoneBlockBIND8x            | The contents of the file is appended to the existing options.                                                                                                                        |
| CheckNamesBIND8X           | Values are warn, fail, ignore, Use Server Value.                                                                                                                                     |
| AllowUpdateBIND8X          | Values are any, none, localhost, localnets, <Use List>, Use Server Value.                                                                                                            |
| AllowQueryBIND8X           | Values are any, none, localhost, localnets, <Use List>, Use Server Value.                                                                                                            |
| AllowTransferBIND8X        | Values are any, none, localhost, localnets, <Use List>, Use Server Value.                                                                                                            |
| NotifyBIND8X               | Values are yes, no, Use Server Value.                                                                                                                                                |
| AlsoNotifyBIND8X           | Value is <Address List>.                                                                                                                                                             |
| ZoneBlockBIND8X            | The filename of the zone block of <i>named.conf</i> . The contents of the file are appended to the existing options.                                                                 |
| <User-Defined Field Names> | Specify the valid user-defined field name(s). The real field name(s) here are those defined in the Global Policies User-Defined Fields with the type "Domain" and/or "Reverse Zone". |
| NegativeCacheTTL           | The amount of time to cache negative responses (entries that do not exist); a typical value is 600.                                                                                  |
| AllowNotifyBIND9X          | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                                                                            |
| AllowQueryBIND9X           | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                                                                            |
| AllowTransferBIND9X        | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                                                                            |
| AllowUpdateBIND9X          | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                                                                            |
| NotifyBIND9X               | Values are No, Yes, Explicit, Use Server Value                                                                                                                                       |
| ZoneBlockBIND9X            | The filename of the zone block of the <i>named.conf</i> . The contents of the file are appended to the existing options.                                                             |
| ZoneEdupLUCENT3            | Values are True, False                                                                                                                                                               |
| AllowQueryLUCENT3          | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                                                                            |

| <b>Field</b>         | <b>Description</b>                                                                                                       |
|----------------------|--------------------------------------------------------------------------------------------------------------------------|
| AllowTransferLUCENT3 | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| AllowUpdateLUCENT3   | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| CheckNamesLUCENT3    | Values are Warn, Fail, Ignore, Use Server Value                                                                          |
| NotifyLUCENT3        | Values are No, Yes, Use Server Value                                                                                     |
| ZoneBlockLUCENT3     | The filename of the zone block of the <i>named.conf</i> . The contents of the file are appended to the existing options. |
| ZoneEdupLUCENT4      | Values are True, False                                                                                                   |
| AllowNotifyLUCENT4   | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| AllowQueryLUCENT4    | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| AllowTransferLUCENT4 | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| AllowUpdateLUCENT4   | Values are Any, None, localhosts, localnets, <Use List>, Use Server Value                                                |
| NotifyLUCENT4        | Values are No, Yes, Explicit, Use Server Value                                                                           |
| ZoneBlockLUCENT4     | The filename of the zone block of the <i>named.conf</i> . The contents of the file are appended to the existing options. |



## qip-setzoneext

---

**qip-setzoneext** resets the content of the zone extension associated with a domain or domains, a DNS server or DNS servers, a reverse zone or reverse zones. To use this CLI command effectively, first use **qip-getzoneext** to obtain the zone extension data, then modify it for the target object.

### Synopsis

```
qip-setzoneext -a domain|reverse|server [-g loginserver]
 [-s servername] [-u username] [-p password] [-o organization]
 [-r reject_file] [-e errmsg_file] -f filename [-t]
```

### Parameters

**qip-setzoneext** recognizes the following parameters:

- |                                       |                                                                                                                   |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-a domain reverse server</code> | Specifies the zone extensions target (domain, reverse zone, or server).                                           |
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| <code>-p <i>password</i></code>       | Specifies the password for the associated administrator account.                                                  |
| <code>-o <i>organization</i></code>   | Specifies the VitalQIP organization (corporation) name.                                                           |
| <code>-r <i>reject_file</i></code>    | Specifies the file where records that are not added or modified are placed.                                       |
| <code>-e <i>errmsg_file</i></code>    | Specifies the filename where error messages are written, if they occur. The default is STDERR.                    |
| <code>-f <i>filename</i></code>       | Specifies the filename of the input data.                                                                         |

-t Specifies that the zone extensions in the input file are appended to the existing zone extensions. Omitting this flag causes existing zone extensions to be removed and replaced with the zone extensions in the input file.

## Command line input examples

**Important!** For a description of the fields in this input example and more information on Domain Extensions, refer to “Define Domains” in Chapter 3 of the *VitalQIP User’s Guide*.

To set Domain Extensions, execute the following:

```
qip-setzoneext -u qipman -p passwd -a domain -f extfile.txt
```

### Contents of extfile.txt

```
DomainName=qtek.com
OptionType=Extensions
ParameterName=Prefix of zone db file
STTL 6400
ParameterName=Postfix of zone db file
candybar IN A 19.20.21.22
snickers IN CNAME candybar.quadritek.com
candybar IN HINFO "VAX" "UNIX"
quadritek.com. IN MX 10 candybar.quadritek.com.
OptionType=BIND-8.X Options
ParameterName=zone block of named.conf
// extra zone options for BIND 8.X server
qddns {
wins-servers {1,2,3,4:1.2.3.5;}
wins-ttl 0;
};
```

To set the DNS server extensions, execute the following:

```
qip-setzoneext -u qipman -p passwd -a server -f extfile.txt
```

### Contents of extfile.txt

```
ServerName=dnsserver.qtek.com
ParameterName= Corporate Extension
include "allowquery.conf"
include "blackhoe.conf"
ParameterName=db.cache file extension
Include "test.conf"
```

To set the Reverse Zone Extensions, execute the following:

```
qip-setzoneext -u qipman -p passwd -a reverse -f extfile.txt
```

### Contents of extfile.txt

```
ReverseZone=170.200.0.0/16
DomainName=qtek.com
```

```
OptionType=Extensions
ParameterName=Prefix of zone db file
STTL 4600
ParameterName=Postfix of zone db file
68 IN PTR madeup.quadritek.com.
69 IN PTR madeup2.quadritek.com.
70 IN PTR madeup3.quadritek.com.
71 IN PTR madeup4.quadritek.com.
OptionType=BIND-8.X Options
ParameterName=zone block of named.conf
qddns {
wins-servers {1,2,3,4:1.2.3.5;}
wins-ttl 0;
};
OptionType=BIND-9.X Options
ParameterName=zone block of named.conf
// extra zone options for BIND 9.X server
forward only;
zone-statistics yes;
OptionType=LUCENT DNS 3.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 3.X server
// ...
OptionType=LUCENT DNS 4.X Options
ParameterName=zone block of named.conf
// extra zone options for LUCENT DNS 4.X Server
// ...
OptionType=WINDOWS 2000 DNS Options
ParameterName=zone-options
// extra zone options for WINDOWS 2000 DNS Server
// ...
```



## qip-setzoneprof

---

**qip-setzoneprof** assigns the Reverse Zone Profile or Domain Profile information. The file specified in the `-f` parameter contains the input information.

### Synopsis

```
qip-setzoneprof [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] [-r reject_file] [-e errmsg_file]
 -f filename
```

### Parameters

**qip-setzoneprof** recognizes the following parameters:

- `-g loginserver` Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- `-s servername` Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- `-u username` Specifies the VitalQIP administrator account to be used in establishing the database connection.
- `-p password` Specifies the password for the associated administrator account.
- `-o organization` Specifies the VitalQIP organization (corporation) name.
- `-r reject_file` Specifies the file where records that are not added or modified are placed.
- `-e errmsg_file` Specifies the filename where error messages are written, if they occur. The default is *STDERR*.
- `-f filename` Specifies the file containing the input data.

### Command line input example

```
qip-setzoneprof -u qipman -p passwd -f domain.txt
```

### Input File Content Example

**Important!** For description of the fields in this input example and details on Domains, refer to "Define Domains" in Chapter 3, of the *VitalQIP User's Guide*.

**Domain input example**

```

Zone=qtek.com
dnsServers=dns1.qtek.com P 0,dns2.qtek1.com S 0,dns2k.qtek2k.com S 0
RefreshTime=21600
ExpirationTime=604800
RetryPeriod=3600
MinimumTTL=86400
ZoneMail=qtek@qtek.com
Bind8AllowQuery=Use List
Bind8AllowTranfer=Use List
Bind8AllowUpdate=Use List
Bind8CheckNames=Fail
Bind8Notify=Use Server Value
Bind8ZoneBlock=
Bind9AllowNotify=Use List
Bind9AllowQuery=Use List
Bind9AllowTranfer=None
Bind9AllowUpdate=None
Bind9Notify=Use Server Value
Bind9ZoneBlock=
Lucent3ZoneEdup=False
Lucent3AllowQuery=None
Lucent3AllowTranfer=None
Lucent3AllowUpdate=None
Lucent3CheckNames=Fail
Lucent3Notify=Use Server Value
Lucent3ZoneBlock=
Lucent4ZoneEdup=False
Lucent4AllowNotify=None
Lucent4AllowQuery=None
Lucent4AllowTranfer=None
Lucent4AllowUpdate=None
Lucent4Notify=Use Server Value
Lucent4ZoneBlock=
Aging=True
MSAllowTranfer=None
MSAllowUpdate=Yes
AgeNoRefreshHours=1
MSNotify=Yes
AgeRefreshHours=1
MSZoneOptions=options

```

**Reverse Zone input example**

```

Zone=200.200.200.0/24
ParentAddress=
NetworkAddress=
dnsServers=,revdns.rev.com P 0
RefreshTime=21600
ExpirationTime=604800

```

```
RetryPeriod=3600
MinimumTTL=86400
ZoneMail=rev1@rev.com
Bind8AllowQuery=Any
Bind8AllowTransfer=Any
Bind8AllowUpdate=Any
Bind8CheckNames=Warn
Bind8Notify=No
Bind8ZoneBlock=
Bind9AllowNotify=Any
Bind9AllowQuery=Any
Bind9AllowTransfer=Any
Bind9AllowUpdate=Any
Bind9Notify=No
Bind9ZoneBlock=
Lucent3ZoneEdup=False
Lucent3AllowQuery=Any
Lucent3AllowTransfer=Any
Lucent3AllowUpdate=Any
Lucent3CheckNames=Warn
Lucent3Notify=No
Lucent3ZoneBlock=
Lucent4ZoneEdup=False
Lucent4AllowNotify=Any
Lucent4AllowQuery=Any
Lucent4AllowTransfer=Any
Lucent4AllowUpdate=Any
Lucent4Notify=No
Lucent4ZoneBlock=
Aging=False
MSAllowTransfer=Any
MSAllowUpdate=No
AgeNoRefreshHours=0
MSNotify=No
AgeRefreshHours=0
MSZoneOptions=
```

**Important!** DNS servers should be listed as FQDN followed by a “P” or “S” for Primary or Secondary, with each server name/type separated by a comma. Additionally, you must specify a 0 (no) or 1 (yes) to indicate sending secure updates. The server must be a secure server in order to send secure updates.



## qip-sitegen

---

“Sites” are a method of further defining subnet groups, and are equivalent to subnet organizations in Windows 2000. **qip-sitegen** is used to push new and changed configurations to Windows 2000 Domain Controller.

### Synopsis

```
qip-sitegen -n dc_svr [-g loginserver] [-s servername] [-u username]
[-p password] [-o organization] [-d directory] [-l] [-z] [-f]
[-c config_action]
```

### Parameters

**qip-sitegen** recognizes the following parameters:

- n *dc\_svr*                Specifies the fully-qualified domain controller name.
- g *loginserver*        Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*        Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*            Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*            Specifies the password for the associated administrator account.
- o *organization*       Specifies the VitalQIP organization (corporation) name.
- d *directory*           Specifies the directory name where the file is to be placed.
- l                        Generates a local file that describes the changes without entering them in active directory.
- z                        Updates the changed sites and subnets only. If the -z parameter is not specified, all sites and subnets are updated.
- f                        Forces a push to the domain controller if the domain controller server was previously locked. If it is still locked, an error message is returned indicating it is “push locked”.

**Command line input example**

```
qip-sitegen -n dcsvr.test.com
```



## qip-siteimport

---

**qip-siteimport** takes site files and subnet files and attempts to create subnets and a subnet organization in VitalQIP. If the network does not exist, the subnet is rejected. If there are conflicts creating the subnet organization (such as when a subnet organization with the same name already exists) the subnet organization creation fails. If there are difficulties creating the subnet (for example, the network does not yet exist for that subnet), the creation of that subnet fails. Only the items that have problems are rejected. If some subnets for a site are valid but others are not, the valid ones are imported into VitalQIP. The invalid ones are written out to a reject file.

**Important!** **qip-siteimport** does not stop processing if an error is encountered and a site or subnet is rejected; it logs the error to the reject file and continues processing. The error message file is populated with non-recoverable type errors, such as “unable to connect to the database”.

### Synopsis

```
qip-siteimport [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -i sitefile -b subnetfile
 [-r reject_file] [-e errormsg_file]
```

### Parameters

**qip-siteimport** recognizes the following parameters:

- g *loginserver*      Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*      Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*          Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*          Specifies the VitalQIP administrator password to be used in establishing the database connection.
- o *organization*      Specifies the VitalQIP organization (corporation) name.
- i *sitefile*           Specifies the name of the file that the exported site information is located in. (Generated by Windows 2000 Domain Controller “Idifde” utility.)

- b *subnetfile* Specifies the name of the file in which the exported subnet information is located. (Generated by Windows 2000 domain controller Idifde utility.)
  
- r *reject\_file* Specifies the name of the file where rejected sites and subnets are logged.
  
- e *errmsg\_file* Specifies the name of the file where error messages are logged.

### Command line input example

Use the “Idifde” utility preceding this CLI in order to generate the input file. For example,

```
ldifde -f sitefile -d cn=sites,cn=configuration,dc=alexis,dc=cube,dc=com -r Objectclass=site
ldifde -f subnetfile -d cn=sites,cn=configuration,dc=alexis,dc=cube,dc=com -r Objectclass=subnet
```

where `dc=alexis`, `dc=cube`, `dc=com` would be replaced with the user’s active directory domain branch. The file name specified after the `-f` option is the name of the output files; one for sites, one for subnets.



## qip-splitmergeenum

---

**qip-splitmergeenum** allows you to split NAPTR (Naming Authority Pointer Record) resource records from a parent zone into one or more child zones, and to merge one or more child zones into a parent zone. Regardless of whether you are splitting or merging zones, you only specify the child zones with this command. The parent zone is always detected by the system.

**Important!** You can only use this command if ENUM is installed.

### Synopsis

```
qip-splitmergeenum -fr domains|-to domains -g Login_Server
-s servername -u UserName -p Password [-o Organization]
[-f Output_File] [-df Data_Format]
```

### Parameters

**qip-splitmergeenum** recognizes the following parameters:

- fr|-to *domains* Specifies if you are merging one or more child domains to a parent domain (-fr) or splitting a parent domain into one or more child domains (-to).  
If you are merging child domains to a parent domain, the domains you specify must already exist. The merged child domains are deleted from VitalQIP.  
If you are splitting a parent domain into one or more child domains, if the domains do not exist, the CLI creates them. The domains must be child domains of the parent domain.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the VitalQIP administrator password to be used in establishing the database connection.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *output\_file* Specifies the directory and filename where the output data is to be placed.

`-df data_format` Determines the output format. The *data\_format* variable must be set to `x` for XML output.

## Command line input examples

The following commands show a zone split and a zone merge.

### Zone Split

The following example splits a parent zone into two child zones.

```
qip-splitmergeenum -to 0.1.6.1.ea64.arpa,5.1.2.1.e164.arpa
-g myloginserver -s mydbsrvr -u qipman -p password
```

The output from the above line is as follows:

```
2 NAPTRs moved from: e164.arpa to: 0.1.6.1.e164.arpa
4 NAPTRs moved from: e164.arpa to: 5.1.2.1.e164.arpa
Please perform ASAP: the pushes of zones changed by the split/merge
operation.
```

### Zone Merge

The following example merges two child zones into their parent zone.

```
qip-splitmergeenum -fr 0.1.6.1.ea64.arpa,5.1.2.1.e164.arpa
-g myloginserver -s mydbsrvr -u qipman -p password
```

The output from the above command is as follows:

```
4 NAPTRs moved from: 5.1.2.1.e164.arpa to: e164.arpa
2 NAPTRs moved from: 0.1.6.1.e164.arpa to: e164.arpa
Please perform ASAP: the pushes of zones changed by the split/merge
operation.
```



## qip-splitrevzone

---

**qip-splitrevzone** provides the capability to split reverse zones.

### Synopsis

```
qip-splitrevzone [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization] -f fromZone/mask -t toZones/mask
```

### Parameters

**qip-splitrevzone** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- f *fromZone/mask* Specifies the existing address and mask of the reverse zone. Use a comma to separate addresses, and precede the mask with a slash (/).
- t *toZones/mask* Specifies the resultant subnet address or addresses and the new mask that you want. Use a comma to separate addresses, and precede the mask with a slash (/).

### Command line input example

```
qip-splitrevzone -u qipman -p passwd -f 144.144.0.0/16 -t 144.144.0.0,144.144.64.0/19
```

## qip-subnetmacpool

---

**qip-subnetmacpool** provides the capability to add, delete and query MAC addresses in a subnet's MAC address pool.

### Synopsis

```
qip-subnetmacpool [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization]
-n subnet_addr -a query|add|delete -m mac_address [-x]
[-t hardware_type]
```

### Parameters

**qip-subnetmacpool** recognizes the following parameters:

- |                        |                                                                                                                                                                                                                        |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -g <i>loginserver</i>  | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.                                                                                                      |
| -s <i>servername</i>   | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                                                             |
| -u <i>username</i>     | Specifies the VitalQIP administrator account used in establishing the database connection.                                                                                                                             |
| -p <i>password</i>     | Specifies the password for the associated administrator account.                                                                                                                                                       |
| -o <i>organization</i> | Specifies the VitalQIP organization (corporation) name.                                                                                                                                                                |
| -n <i>subnet_addr</i>  | Specifies the address of the subnet.                                                                                                                                                                                   |
| -a query add delete    | Specifies the process: to add or delete a MAC address, or to query the available MAC pool. The optional argument (-x) is available for <b>-a add</b> only, and indicates that the specific address should be excluded. |
| -m <i>mac_address</i>  | Specifies the MAC address to be added or deleted. Supports the trailing wildcard character (*).                                                                                                                        |
| -x                     | Excludes the MAC address from the processing.                                                                                                                                                                          |

`-t hardware_type` Specifies the hardware type of the object with the MAC address to be added. The following options are available:

- Ethernet
- Token Ring
- AX.25
- Pronet
- Chaos
- IEEE802
- Arcnet

### Command line input example

The following example adds the excluded MAC address “123456\* type ethernet” to the MAC pool on subnet 10.100.30.0.

```
qip-subnetmacpool -u qipman -p passwd -n 10.100.30.0 -a add -m 123456* -x -t ethernet
```



## qip-syncexternal

---

**qip-syncexternal** retrieves A, AAAA, PTR, SRV, CNAME or TXT resource records from a DNS server. It tries to create External objects with A and PTR resource records. If an object cannot be created, it adds a resource record managed by external updates to a domain or reverse zone.

When invoked with a `-z` parameter, **qip-syncexternal** retrieves resource records from a particular zone. Otherwise it retrieves resource records for every primary zone on the DNS server.

When invoked with the `-a` parameter (**Append Only mode**), new resource records and objects are added to VitalQIP – no objects or resource records will be tombstoned in VitalQIP.

Resource records that are tombstoned by the `qip-syncexternal` CLI, can be deleted from VitalQIP with the **qip-tombstonepurge** CLI.

**Important!** The **qip-syncexternal** CLI command is restricted to capturing CNAME and SRV resource records that are added to DNS by a Windows operating system.

The **qip-syncexternal** CLI replaces the **qip-miniddma** CLI.

Other than for A and PTR resource records which can appear anywhere in a zone, `qip-syncexternal` only records or tombstones resource records that exist in the `_msdcs hive` directory.

### Synopsis

```
qip-syncexternal [-g loginserver] -d dns_server_name [-s servername]
 [-o organization] [-u username] [-p password] [-z zone_name] [-a]
 [-w] CNAME|SRV|PTR|A|AAAA|TXT|All
```

### Parameters

**qip-syncexternal** recognizes the following parameters:

|                                        |                                                                                                                   |
|----------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-g <i>loginserver</i></code>     | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-d <i>dns_server_name</i></code> | Specifies the fully-qualified name of a DNS server to retrieve records from.                                      |
| <code>-s <i>servername</i></code>      | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-o <i>organization</i></code>    | Specifies the VitalQIP organization (corporation) name.                                                           |

|                                                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>-u <i>username</i></code>                    | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>-p <i>password</i></code>                    | Specifies the password for the administrator account.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>-z <i>zone_name</i></code>                   | Specifies the name of a zone on the DNS server to retrieve records from. If this parameter is omitted, <b>qip-syncexternal</b> transfers records for all primary forward zones on this DNS server.                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>-a</code>                                    | Specifies <b>Append Only</b> mode, in which new resource records and objects are added to VitalQIP – no objects or resource records are deleted from VitalQIP.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <code>-w</code>                                    | Sets the CLI in verbose mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>CNAME   SRVA   AAAA   PTR   TXT   All</code> | Specifies the records to retrieve from the DNS server. Code one or more of these parameters, separated by a space:<br><b>CNAME</b> Retrieves Canonical Name (CNAME) records.<br><b>SRV</b> Retrieves Server Resource (SRV) records.<br><b>A</b> Retrieves Host Ipv4 (A) records and creates object records with A records that are published in DNS.<br><b>AAAA</b> Retrieves Host Ipv6(AAAA) records.<br><b>PTR</b> Retrieves pointer (PTR) records and creates object records with PTR records that are published in DNS<br><b>TXT</b> Retrieves Text (TXT) records.<br><b>All</b> Retrieves CNAME, SRV, A, and PTR records. |

### Command line input example

- To sync A, CNAME, SRV and PTR records for all primary zones on the DNS server *dns.example.com*:  

```
qip-syncexternal -d dns.example.com All
```
- To sync CNAME and SRV records for all primary zones on the DNS server *dns.example.com*:  

```
qip-syncexternal -d dns.example.com CNAME SRV
```
- To sync A, CNAME, SRV and PTR records for the zone *zone.example.com* on the DNS server *dns.example.com*:  

```
qip-syncexternal -d dns.example.com -z zone.example.com All
```

- To sync A, CNAME, SRV and PTR records for all primary zones on the DNS server *dns.example.com* without tombstoning any records or objects in VitalQIP:

```
qip-syncexternal -d dns.example.com -z zone.example.com -a All
```

- To sync A, CNAME, SRV and PTR records for all primary zones on the DNS server *dns.example.com* while printing progress messages to standard out:

```
qip-syncexternal -d dns.example.com -w All
```



## qip-template

---

**qip-template** creates DHCP templates based on the data provided in the input file passed as command parameter `-f`.

### Before you begin

- Multiple templates can be added/modified in one input file.
- Use field “template-type” to specify the type of template and a value. The available values are: `OPTION`, `SUBNET_POLICY`, `CLIENTCLASS_POLICY`, `SCOPE_POLICY`.
- To assign options to policy templates (the template-type is set to `SUBNET_POLICY`, `CLIENTCLASS_POLICY`, or `SCOPE_POLICY`), specify the option name explicitly (for example, `renew-address-shuffle=on`).
- For options that can have the value “SAME AS IN SUBNET PROFILE” or “SAME AS IN OBJECT PROFILE”, a string value can be set.

### Synopsis

```
qip-template -t template_type -f input_filename [-g loginserver]
[-s servername] [-u username] [-p password] [-o organization]
[-r reject_file] [-e errmsg_file]
```

### Parameters

**qip-template** recognizes the following parameters:

- |                                       |                                                                                                                                                                                             |
|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>-t <i>template_type</i></code>  | Specifies the type of template you are creating. Allowable values are:<br><code>OPTION</code><br><code>SUBNET_POLICY</code><br><code>CLIENTCLASS_POLICY</code><br><code>SCOPE_POLICY</code> |
| <code>-f <i>input_filename</i></code> | Specifies the name of the file that contains the input data.                                                                                                                                |
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the <code>LOGIN</code> environment variable.                                                              |
| <code>-s <i>servername</i></code>     | Specifies the VitalQIP database server. This value is the value of the <code>QIPDATASERVER</code> environment variable.                                                                     |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                            |
| <code>-p <i>password</i></code>       | Specifies the password for the associated administrator account.                                                                                                                            |

- `-o organization`      Specifies the VitalQIP organization (corporation) name.
- `-r reject_file`        Specifies the directory and filename for the rejected records.
- `-e errmsg_file`        Specifies the filename where error messages are written, if they occur.

### Command line input example

```
qip-template -s qipdataserver -u qipman -p passwd -f tmpl.dat
```

### Input file content examples

#### Option Template

```
template-name=Testtempl
template-type=DHCP Option Template
001=User Defined
001>User Defined=198.200.138.0
002=1300
003=User Defined
003>User Defined=198.200.138.156
004=User Defined
004>User Defined=198.200.138.65
005=198.200.138.60
006=User Defined
006>User Defined=198.200.138.45
007=198.200.138.5
008=198.200.138.8
009=198.200.138.46
010=198.200.138.54
011=198.200.138.59
012=User Defined
012>User Defined=nametest
013=43243
014=32534
015=User Defined
015>User Defined=Domain2.qa.quadritek.com
016=198.200.138.4
017=/opt/qip50/qip/rootpath
018=/opt/qip50/path
019=False
020=True
021=198.200.138.0 255.255.255.0
022=65432
023=10000
024=9098
025=5222
026=0
027=True
028=User Defined
028>User Defined=10.200.70.10
```

```

029=False
030=False
031=True
032=198.200.138.179
033=198.200.138.179 198.200.138.165
034=True
035=9999
036=False
037=34938
038=30600
039=False
040=User Defined
040>User Defined=NISDomain
041=User Defined
041>User Defined=198.200.138.53 10.200.60.99
042=User Defined
042>User Defined=198.200.138.5 10.200.60.4
043=whatever1
044=10.200.70.99
045=10.200.60.100
046=B-node
047=testfield1
048=198.200.138.4
049=198.200.138.179
051=Limited
051>Limited Lease Time=8046183
052=60000
058=34502
059=45454
060=101010
061=ci4
064=25345
065=198.200.138.52 10.200.60.4
066=Name Server
067=test bootfile
068=10.200.70.100 198.200.138.77
069=10.200.60.55 10.200.100.6
070=198.200.138.52 10.200.160.2
071=10.200.160.5 198.200.138.74
072=10.200.160.211 198.200.138.211
073=198.200.138.52 198.200.138.56 198.200.138.59
074=10.200.160.53 198.200.138.46
075=198.200.138.47 10.200.160.5
076=10.200.100.66 10.200.100.166
126=/opt/qip50/qip/bootfile
127=7

```

### **Subnet Policy Template**

```

template-name=testsubnet1
template-type=DHCP Policy Template - Subnet
init-reboot-address-shuffle=on

```

### **Client Class Policy Template**

```

template-name=testcc1

```

```
template-type=DHCP Policy Template - Client Class
init-reboot-address-shuffle=off
renew-address-shuffle=on
renew-address-shuffle-max-renews=4
```



## qip-tombstonepurge

---

**qip-tombstonepurge** deletes tombstoned external objects and resource records from the VitalQIP database. It looks at the record's EDUP timestamp and the database policy DYNDNS/Tombstone Max Life, and deletes any records in the current organization that are older than their natural lifespan. The **qip-tombstonepurge** CLI is installed on the enterprise server and is called periodically by the Schedule Service.

### Synopsis

```
qip-tombstonepurge [-g loginserver] [-s servername] [-u username]
 [-p password] [-o organization]
```

### Parameters

**qip-tombstonepurge** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.

### Command line input example

To delete tombstoned objects and resource records in organization "VitalQIP Organization":  
 qip-tombstonepurge -o "VitalQIP Organization"

### Output example

Successfully purged 0 Objects and 0 Resource Records in Org VitalQIP Organization.



## qip-ungetdecnetaddr

---

**qip-ungetdecnetaddr** releases a DECNet node in a specified DECNet area from the VitalQIP database. Once you run **qip-getdecnetaddr** to get a DECnet node and you decide not to use that address, run **qip-ungetdecnetaddr**. That address is marked as "selected". However, the **qipd** daemon cleans up all selected entries every day.

### Synopsis

```
qip-ungetdecnetaddr -a DECnet_area -d DECnet_node
[-g loginserver] [-s servername] [-u username] [-p password]
[-o organization]
```

### Parameters

**qip-ungetdecnetaddr** recognizes the following parameters:

- a *DECnet\_area* Specifies the DECNet area.
- d *DECnet\_node* Specifies the DECNet node.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.

### Command line input example

To release the DECNet Node 123 on DECNet Area 10:

```
qip-ungetdecnetaddr -u qipman -p passwd -a 10 -d 123
```



## qip-ungethubport

---

**qip-ungethubport** is used in conjunction with **qip-gethubport**. If you run **qip-gethubport** to get an available hub port and you do not want to use that port, run **qip-ungethubport**. Running **qip-gethubport** marks the port number as “selected”. However, the **qipd** daemon cleans up all selected entries every day.

### Synopsis

```
qip-ungethubport -b hub_name -l slot_name -n hub_port_num
 [-g loginserver] [-s servername] [-u username] [-p password]
 [-o organization]
```

### Parameters

**qip-ungethubport** recognizes the following parameters:

- b *hub\_name*            Specifies the hub name.
  
- l *slot\_name*            Specifies the slot name.
  
- n *hub\_port\_num*        Specifies the port number.
  
- g *loginserver*        Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*        Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*            Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*            Specifies the password for the associated administrator account.
  
- o *organization*        Specifies the VitalQIP organization (corporation) name.

### Command line input example

```
qip-ungethubport -u fred -p passwd -b hub1 -l slot3 -n 55
```



## qip-ungetipaddr

---

**qip-ungetipaddr** is used in conjunction with **qip-getipaddr**. If you run **qip-getipaddr** to get an available IP address, but you decide not to use that address, run **qip-ungetipaddr** if possible. Running **qip-getipaddr** marks the address as “selected”. However, the **qipd** daemon cleans up all selected entries every day.

### Synopsis

```
qip-ungetipaddr -a object_address [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization]
```

### Parameters

**qip-ungetipaddr** recognizes the following parameters:

- a *object\_address* Specifies the object address.
  
- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.

### Command line input example

```
qip-ungetipaddr -u fred -p passwd -a 199.200.139.27
```



## qip-unlock

---

**qip-unlock** clears all addresses in a “Selected” state.

**Important!** Objects should not ordinarily be in a Selected state. If objects are in this state, there may be an issue with the database. Please contact Technical Support (page xvii) before using this CLI.

### Synopsis

```
qip-unlock -n name|-a address|-c [-g loginserver] [-s servername]
 [-u username] [-p password] [-o organization]
```

### Parameters

**qip-unlock** recognizes the following parameters:

- n *name*                      Specifies the hostname of the object (-n) or the address (-a) that is in a Selected state and needs to be cleared. If you pass -c, all addresses in a Selected state are cleared.
- | -a *address*
- | -c
  
- g *loginserver*              Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*              Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- u *username*                  Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*                  Specifies the password for the associated administrator account.
  
- o *organization*              Specifies the VitalQIP organization (corporation) name.

### Command line input examples

```
qip-unlock -u qipman -p passwd -a 199.200.139.27
qip-unlock -u qipman -p passwd -n myhostname
qip-unlock -u qipman -p passwd -c // unlocks all
```

□

## qip-util

---

**qip-util** drops database tables, indexes, and so on, from either the LAM (Audit Manager) *or* VitalQIP database and the Sybase or Oracle databases. All tables in the Audit Manager database *or* VitalQIP database are changed.

### Synopsis

```
qip-util [-t QIPDBASE] [-s QIPDATASERVER] [-u user] [-p password]
 [-l log_file] [-q output_file] [-i script_path] [-b db_name] [-a]
 FUNCTION VALUES... [-z]
```

### Parameters

**qip-util** recognizes the following parameters:

- |                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -t <i>QIPDBASE</i>      | Specifies the database server type: <b>Oracle</b> or <b>Sybase</b> .                                                                                                                                                                                                                                                                                                                                                                                                   |
| -s <i>QIPDATASERVER</i> | Specifies the name of the database server. Note the following: <ul style="list-style-type: none"><li>• The database server name must match the Sybase server name or the Oracle database alias name.</li><li>• This parameter is optional if the \$QIPDATASERVER environment variable is set.</li><li>• The command line argument overrides the environment variable.</li><li>• For Audit Manager, the Audit Manager database server name must be specified.</li></ul> |
| -u <i>user</i>          | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                       |
| -p <i>password</i>      | Specifies the password for the associated administrator account.                                                                                                                                                                                                                                                                                                                                                                                                       |
| -l <i>log_file</i>      | Specifies the name of the log file.                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| -q <i>output_file</i>   | Specifies Quiet Mode. If this parameter is omitted, the output is sent to <i>STDOUT</i> .                                                                                                                                                                                                                                                                                                                                                                              |
| -i <i>script_path</i>   | Specifies the directory where the SQL scripts are located.                                                                                                                                                                                                                                                                                                                                                                                                             |
| -b <i>db_name</i>       | Specifies the database name; VitalQIP or LAM (Audit Manager). You cannot specify both databases at the same time. The default is VitalQIP.                                                                                                                                                                                                                                                                                                                             |

- a Appends function values discussed in Table 41 to the log file. The default is to overwrite the log file.
- z Allows commands to be executed with the encrypted password.

**Table 41 Function values**

| "Function Values" name   | Description                                                                                                                                                         | Database          | Product                    |
|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|----------------------------|
| CalculateQIPSize         | Estimates the size of the VitalQIP database based on the following:<br><number_of_objects/number_of_subnets>                                                        | Sybase/<br>Oracle | VitalQIP                   |
| CalculateLAMSize         | Estimates the size of the LAM (Audit Manager) database, based on the following:<br><number_of_dhcp_clients><br><number_of_static_objects><br><number_of_nt_objects> | Sybase/<br>Oracle | Audit Manager              |
| CheckDatabaseLogin       | Checks to see if the connection to Sybase/Oracle is OK.                                                                                                             | Sybase/<br>Oracle | VitalQIP/<br>Audit Manager |
| CheckDBProcesses         | Shows the number of processes that are currently connected to the Sybase/Oracle database by using: <database_name>. The default is <b>QIP</b> .                     | Sybase/<br>Oracle | VitalQIP/<br>Audit Manager |
| CheckVersionFromDatabase | Checks the version information from the qip_version table in the Sybase/Oracle database by using: <database_name>. The default is <b>QIP</b> .                      | Sybase/<br>Oracle | VitalQIP/<br>Audit Manager |
| CheckVersionFromData     | Checks the version information from the qip_version (qef) file by using:<br><export_path>.                                                                          | N/A               | VitalQIP/<br>Audit Manager |
| CheckVersionFromScript   | Checks the version information from the <b>table.sql</b> script.                                                                                                    | Sybase/<br>Oracle | VitalQIP/<br>Audit Manager |
| CheckSybaseDevice        | Checks to see if a Sybase device exists by using: <logical_device_name>.                                                                                            | Sybase            | VitalQIP/<br>Audit Manager |
| CheckSybaseDatabase      | Checks to see if a Sybase database exists by using: <database_name>. The default is <b>QIP</b> .                                                                    | Sybase            | VitalQIP/<br>Audit Manager |
| CheckUserExists          | Checks to see if the database login user exists by using: <login_name>                                                                                              | Oracle            | VitalQIP                   |

| <b>"Function Values" name</b> | <b>Description</b>                                                                                                                                                                                                                                                            | <b>Database</b>   | <b>Product</b>                |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|-------------------------------|
| ClearAdmin                    | Removes all logins and users assigned to the database with the related roles or groups by using: <database_name>. The default is <b>QIP</b> .                                                                                                                                 | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| ClearData                     | Truncates all data from all user tables by using: <database_name>. The default is <b>QIP</b> . For Sybase, drops all tables with type equal to "U" (user tables -- not system tables). For Oracle, drops all tables owned by the user running <b>qip-util</b> .               | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| CreateAccess                  | Calls <b>create_access.sql</b> . This function should only be called by the installation.                                                                                                                                                                                     | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| CreateSybaseDatabase          | Creates a Sybase database by using:<br><data_device_name><data_size><br><log_device_name><log_size><br><database_name>                                                                                                                                                        | Sybase            | VitalQIP/<br>Audit<br>Manager |
| CreateSybaseDevice            | Creates a Sybase device by using:<br><logical_device_name/<br>physical_name size/device_size>                                                                                                                                                                                 | Sybase            | VitalQIP/<br>Audit<br>Manager |
| DropIndex                     | Drops all indexes on all user indexes in the database by using: <database_name>. The default is <b>QIP</b> . For Sybase, drops all indexes on all user tables. For Oracle, drops all indexes owned by the user running <b>qip-util</b> .                                      | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| DropSP                        | Drops all stored procedures in the database by using: <database_name>. The default is <b>QIP</b> . For Sybase, drops all stored procedures in the database. For Oracle, drops all stored procedures owned by the user running <b>qip-util</b> .                               | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| DropTable                     | Drops all user tables in the database by using: <database_name>. The default is <b>QIP</b> . For Sybase, drops all user tables in the database. For Oracle, drops all tables owned by the user running <b>qip-util</b> .                                                      | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| DropTrigger                   | Drops all triggers on all user tables in the database by using: <database_name>. The default is <b>QIP</b> . For Sybase, drops all triggers on all user tables in the database. For Oracle, drops all triggers on all user tables owned by the user running <b>qip-util</b> . | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |

| "Function Values" name  | Description                                                                                                                                                                                                                          | Database          | Product                       |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|-------------------------------|
| EditTextFile            | Finds the text file name and the first line containing the text you want to delete. You can then establish what you would like to add in its place with the "add_line" (optional). Use:<br><text_file_name><delete_line> <add_line>. | N/A               | N/A                           |
| EstimateRequiredSpace   | Estimates the minimum disk space used during qip-import using:<br><database_name>.                                                                                                                                                   | Sybase/<br>Oracle | VitalQIP/<br>Audit<br>Manager |
| GetAndSetSybaseDBOption | Gets and sets the value of the Sybase dboption trunc.Log on chkpt by using <option_value><database_name>. The default is <b>QIP</b> .                                                                                                | Sybase            |                               |
| GetDatabaseSize         | Gets and sets the size of the database by using: <database_name>. The default is <b>QIP</b> .                                                                                                                                        | Sybase            | VitalQIP/<br>Audit<br>Manager |
| OracleReCompile         | Recompiles all stored procedures and triggers owned by the user running <b>qip-util</b> by using: <database_name>. The default is <b>QIP</b> .                                                                                       | Oracle            | VitalQIP/<br>Audit<br>Manager |
| RemoveInvalidChars      | Removes all invalid (unprintable) characters from all files in the specified directory by using:<br><export_path/file_name>                                                                                                          | N/A               | VitalQIP                      |
| RemoveSpaceFields       | Goes through all the string fields on all user tables and removes all space-only fields by using: <table_name>                                                                                                                       | Sybase            | VitalQIP                      |
| SearchReplace           | Searches some special characters, and replaces them with proper characters by using: <search_char> <replace_char>. By default, changes "\n" to a single space; a double quote to a single quote; and "^" to a single space.          | Sybase            | VitalQIP                      |
| SetSybaseConfigure      | Sets Sybase configuration values based on the file <config_file>. VitalQIP and LAM (Audit Manager)databases require the following settings:<br>- Procedure cache percent=22<br>- Total memory=21577<br>- Number of locks=100000      | Sybase            | VitalQIP/<br>Audit<br>Manager |

| <b>"Function Values" name</b> | <b>Description</b>                                               | <b>Database</b> | <b>Product</b>                |
|-------------------------------|------------------------------------------------------------------|-----------------|-------------------------------|
| SybaseUpdateStatistics        | Runs "update statistics" on all tables by using: <database_name> | Sybase          | VitalQIP/<br>Audit<br>Manager |
| OracleUpdateStatistics        | Runs an analysis on all objects within the database.             | Oracle          | VitalQIP/<br>Audit<br>Manager |



## vercheck

---

**Important!** To run this utility, you should have the Java Runtime Environment installed, and located in the execution path of your environment. If you do not, you will not be able to display the version of any Java-based Lucent product components such as VitalQIP.

The **vercheck** utility lists the VitalQIP version number of every VitalQIP program in a specific directory (and its subdirectories). The resulting information can be displayed on the screen or output to a file and includes the filename, file size, date/time stamp, file type, version number, and the file's "checksum". Using this utility helps you to maintain consistency between upgrades.

### Synopsis

```
vercheck [-d directory] [-m field_mask] [-j java_FQN] [-c] [-e] [-z]
 [-5] [filename]
```

### Parameters

**vercheck** recognizes the following parameters:

- d *directory* Specifies the directory for which version information is to be obtained. The default is the current working directory.
  
- m *field\_mask* Identifies the fields that are to be displayed by **vercheck**. The fields are identified using a field mask consisting of "0"s and "1"s. A "1" indicates that the field should be displayed, and a "0" indicates that the field should not be displayed. The fields, in order of specification, are:
  - File Name
  - File Size
  - File Owner
  - File Permissions
  - File Creation Date
  - File Modification Date
  - File Type
  - File Version
  - File Checksum
 If this parameter is omitted, all fields are displayed.

- j *java\_FQN* This option allows you to specify the location (the fully-qualified file name) of the JAVA runtime executable (java on UNIX and *java.exe* on Windows) to be used to determine the version of the Java archive (JAR) file. If this parameter is not specified, the utility assumes that the directory containing the JAVA runtime executable is defined in the PATH environment variable. For example, on a UNIX system, you can determine the version of the *acmengine.jar* file using either of the following methods (this assumes Bourne Shell or K-Shell in this example):
 

```
vercheck -z -j /usr/local/jdk1.3/bin/java
 acmengine.jar
 acmengine.jar: 1.1.5
```

 or
 

```
PATH=$PATH:/usr/local/jdk1.3/bin; export PATH
vercheck -z acmengine.jar
acmengine.jar: 1.1.5
```
  
- c Outputs the information in CSV format.
  
- e Provides VitalQIP environment information.
  
- z Provides only the filenames and product version numbers.
  
- 5 Searches only for VitalQIP 5.x versions. If this parameter is omitted, **vercheck** searches for all VitalQIP versions.
  
- filename* Provides information about the specified file. If this parameter is omitted, all files in the current directory or the directory specified by the -d option is processed. The subdirectories are also be processed.



## 2 Exporting and importing whole databases

### Overview

---

VitalQIP has various types of import and export capabilities. You can import and export whole VitalQIP databases with the **qip-import** and **qip-export** CLI commands. These CLI commands are mainly used for upgrades or backup/recovery purposes.

**Important!** Make sure you have shut down your VitalQIP services before running either of these CLI commands.

## qip-export

---

**qip-export** exports all data from either the VitalQIP database *or* Audit Manager database for use with Sybase and Oracle. This can be performed on either a UNIX or Windows platform. It does not change the data in the database. Error data is output to the *QIPHOME/log/qip-export.log* file.

### Synopsis

```
qip-export [-t QIPDBASE] [-s QIPDATASERVER] [-u user] [-p password]
 [-l log_file] [-q output_file] [-i script_path] [-b database_name]
 [-d export_path] [-a] [-k] [-z encrypted_password]
```

### Parameters

**qip-export** recognizes the following parameters:

- |                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -t <i>QIPDBASE</i>      | Specifies the type of database – either <b>SYBASE</b> or <b>ORACLE</b> . This parameter is optional if the <i>\$QIPDBASE</i> environment variable is set. The command line argument overrides the environment variable.                                                                                                                                                                                                                                                       |
| -s <i>QIPDATASERVER</i> | Specifies the name of the database server. Note the following: <ul style="list-style-type: none"><li>• The database server name must match the Sybase server name or the Oracle database alias name.</li><li>• This parameter is optional if the <i>\$QIPDATASERVER</i> environment variable is set.</li><li>• The command line argument overrides the environment variable.</li><li>• For Audit Manager, the Audit Manager database server name must be specified.</li></ul> |
| -u <i>user</i>          | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                              |
| -p <i>password</i>      | Specifies the password for the associated administrator account.                                                                                                                                                                                                                                                                                                                                                                                                              |
| -l <i>log_file</i>      | Specifies the file name of the log file.                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| -q <i>output_file</i>   | Quiet Mode. If this parameter is omitted, the output is sent to <i>STDOUT</i> .                                                                                                                                                                                                                                                                                                                                                                                               |
| -i <i>script_path</i>   | Specifies the directory where the SQL scripts reside.                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -b <i>db_name</i>       | Specifies the database name; QIP <i>or</i> LAM (Audit Manager). You cannot specify both databases at the same time. The default is QIP.                                                                                                                                                                                                                                                                                                                                       |

- d *export\_path* Specifies the path and directory of the export data file.
- a Appends to the log file. The default is to overwrite the log file.
- k Skips the prompt.
- z *encrypted password* Allows commands to be executed with the encrypted password.

**Command line input example**

```
qip-export -u qipadmin -p <password> -d /tmp/qip
```



## qip-import

---

**qip-import** imports either all VitalQIP *or* all Audit Manager data from an earlier database export into the database for use with Sybase and Oracle. It overwrites the entire database. Error data is output to the file `$QIPHOME/log/qip-import.log`. The **qip-import** command is only installed with the enterprise server component and can only be run on the VitalQIP enterprise server. The `-d` parameter is required for the **qip-import** command.

**Important!** To ensure optimal database performance, Oracle database administrators can have the **qip-import** utility automatically analyze the VitalQIP tables. Remove the comment character (`#`) from the following line in the Regular Import section of the `$QIPHOME/script/qipinst.config` file:

```
#qip-util(oracle)::OracleUpdateStatistics
```

### Synopsis

```
qip-import [-t QIPDBASE] [-s QIPDATASERVER] [-u user] [-p password]
 [-l log_file] [-q output_file] [-i script_path] -d input_path [-a]
 [-k] [-z encrypted_password]
```

### Parameters

**qip-import** recognizes the following parameters:

- |                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>-t QIPDBASE</code>      | Specifies the type of database - either SYBASE or ORACLE. This parameter is optional if the <code>\$QIPDBASE</code> environment variable is set. The command line argument overrides the environment variable.                                                                                                                                                                                                                                                                      |
| <code>-s QIPDATASERVER</code> | Specifies the name of the database server. Note the following: <ul style="list-style-type: none"><li>• The database server name must match the Sybase server name or the Oracle database alias name.</li><li>• This parameter is optional if the <code>\$QIPDATASERVER</code> environment variable is set.</li><li>• The command line argument overrides the environment variable.</li><li>• For Audit Manager, the Audit Manager database server name must be specified.</li></ul> |
| <code>-u user</code>          | Specifies the VitalQIP administrator account to be used in establishing the database connection. If the administrator is not specified, it will use the information in the <code>qip.pcy</code> file.                                                                                                                                                                                                                                                                               |
| <code>-p password</code>      | Specifies the password for the associated administrator account. If the administrator is not specified, it will use the information in the <code>qip.pcy</code> file.                                                                                                                                                                                                                                                                                                               |
| <code>-l log_file</code>      | Specifies the file name of the log file.                                                                                                                                                                                                                                                                                                                                                                                                                                            |

- q *output\_file* Specifies Quiet Mode. If this parameter is omitted, the output is sent to *STDOUT*.
- i *script\_path* Specifies the directory where the SQL scripts reside.
- d *input\_path* **Required.** Specifies the path (directory) of the input data.
- a Appends to the log file. The default is to overwrite the log file.
- k Skips the prompt.
- z *encrypted password* Allows commands to be executed with the encrypted password.

**Command line input example**

```
qip-import -u qipadmin -p <password> -d /tmp/qip
```





# 3 Exporting and importing DNS and Bootptab files

## Overview

---

VitalQIP offers DNS and Bootptab file import and export utilities, which allow you to extract current BIND 4.9.x, BIND 8.x, BIND 9.x, or bootp data, and import it into your VitalQIP system.

These import/export utilities are available on both UNIX and Windows platforms and can be used for either upgrades or new installations.

## Exporting DNS files

---

The **qip-dnscsv** CLI command searches all DNS database files and extracts data to create CSV (comma separated variable length) files capable of being imported into VitalQIP. (Alternately, you can use the CLI, in interactive mode, to transfer DNS data directly from an operational DNS server.) These files are created in the format specified in the sections describing each CLI command.

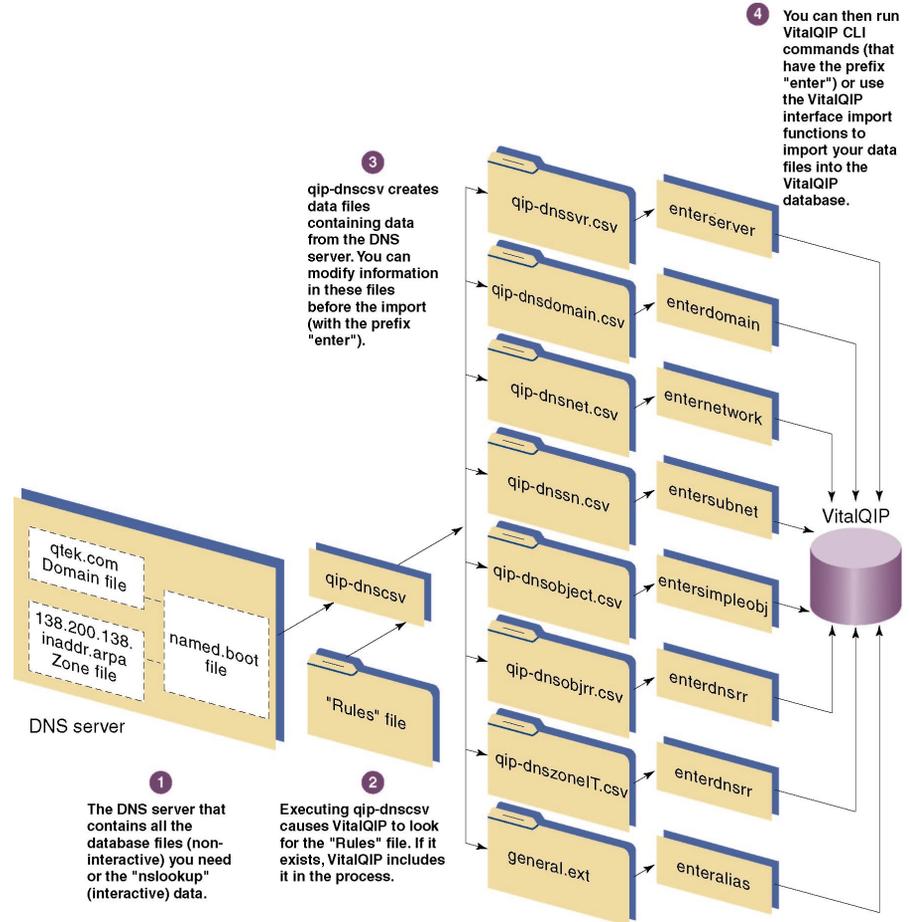
The extracted data can be imported by using the CLI commands listed in Table 42 (and explained later in this chapter).

**Table 42** CLI import commands

| CLI Command                          | Description                                            |
|--------------------------------------|--------------------------------------------------------|
| enterserver                          | Use to import DNS servers.                             |
| enterdomain                          | Use to import Domain information.                      |
| enternetwork                         | Use to import Network information.                     |
| entersubnet                          | Use to import Subnet information.                      |
| entersimpleobj <i>or</i> qipbulkload | Use to import Object information.                      |
| enterdnsobj                          | Use to import Resource Record information for objects. |
| enterdnsrr                           | Use to import Resource Record information for zones.   |
| enteralias                           | Use to import additional Object Alias information.     |

Specific information must be provided to the CLI, such as DNS BIND version information, the location of the *named.boot* (BIND 4.9.x) or *named.conf* (BIND 8.x or BIND 9.x) file, the location of the output files, the subnet mask to derive the subnet address, and the location of the “rules” file. The “rules” file is explained later.

Figure 2 explains the process:



**Figure 2 The DNS export process**

`qip-dnscsv` does the following:

1. Read the DNS startup file (`named.boot` or `named.conf`) to find the DNS zones and database files. If the `-z` parameter is used to specify the zone in the interactive mode, the DNS startup file is not read.
2. Optionally, read the "rules" file to obtain user preferences as to how to extract information from the database files. The "rules" file enables the user to force `qip-dnscsv` to obtain information from the Resource Records, specifically the network and subnet mask, to extract domain names from SOA only or from all objects.
3. Format the information into specific CSV formats so data can be imported into VitalQIP.

4. Retrieve the DNS information directly from the server in interactive mode. Otherwise, it will retrieve the information from reading the *named.boot/named.conf* file, locating the directory of the database files, and going through all database files to extract the required information.
5. Additionally, it reads resource records and retrieves the corresponding information, as shown in Table 43 following.
6. From the reverse zone file, **qip-dnscsv** obtains the PTR record to determine which name to use as object name when multiple A records of the same IP are present.

**Important!** **qip-dnscsv** only imports data from Master servers (not slaves).

**Table 43 Resource Records**

| Resource Records | Retrieved information                                                                                         |
|------------------|---------------------------------------------------------------------------------------------------------------|
| SOA              | Domain Name, Primary Server Name, Mail address of contact, Refresh Interval, Retry Time, TTL, and minimum TTL |
| NS               | Secondary Servers                                                                                             |
| A                | Host Name and IP Address*                                                                                     |
| CNAME            | Host Alias                                                                                                    |
| HINFO            | CPU Type, OS Type                                                                                             |
| WKS              | Comment                                                                                                       |
| TXT              | Object Description (free form information)                                                                    |
| \$ORIGIN         | The base Domain for the subsequent records                                                                    |
| \$INCLUDE        | The filename and, if available, the new ORIGIN                                                                |

**A few things to keep in mind\***

- If the \$ORIGIN statement exists, there may also be an \$INCLUDE record with it. If the \$ORIGIN domain name and the \$INCLUDE domain name differ, the domain with the \$INCLUDE record overrides the domain in the SOA and the \$ORIGIN statement. However, the \$ORIGIN record may not exist. If so, the \$INCLUDE record has the domain name following the file name.
- If the Owner of an A resource record is a domain, “NULL” is assigned as the object name. Only one object is created. If more than one object shares an IP address, the other objects assigned to that IP address are created as resource records.
- \$GENERATE statements are supported. For example:  

```
$GENERATE 10-20 QIPDNS$ 10.20.30.$
```

Please refer to “Output File 5 - qip-dnsobject.csv”, on page 326 for details of resource records generated by the \$GENERATE statement.



## qip-dnscsv

---

### When to use

This section describes how to use the **qip-dnscsv** CLI. First, you need to gather certain information and do some preparation.

### Procedure

Before using the **qip-dnscsv** CLI, do the following:

- 1 Formulate the optional “rules” file, based on the description in the section on “Using the “Rules” file with qip-dnscsv”, on page 318”, and give it an appropriate name, for example: */home/you/named/dnsrule.dat*.
- 2 Find out the DNS BIND version in your facility (for example, BIND 4.9.x or BIND 8.x or BIND 9.x).
- 3 Find out the directory name of the DNS startup files. Normally, it is in */etc/named/named.boot* (for BIND 4.9.x) or */etc/named/named.conf* (for BIND 8.x or BIND 9.x).
- 4 For interactive mode, obtain the IP address of the DNS server to be used as the source of the information.
- 5 Decide what directory you want the output to go to.
- 6 Find out where **qip-dnscsv** resides and verify that it is in your path.
- 7 Initiate the process by typing the command in the command line. For example:

```
qip-dnscsv -b 4 -n /etc/named/named.boot -o /home/you/named -r /home/you/named/dnsrule.dat
```

In this example, you have BIND 4.9.x in your facility, and you want to retrieve specific DNS information from the Resource Records by using the “rules” file specified in the `-r` parameter, specifically:

```
/home/you/named/dnsrule.dat
```

In another example, the process is initiated in interactive mode:

```
qip-dnscsv -b 8 -n /etc/named/named.conf -o /home/you/named -r /home/you/named/dnsrule.dat
-i -s 10.200.20.10
```

In this example, you have BIND 8.x running on the server at IP address 10.200.20.10. The zones to be processed are read from the *named.conf* files. The same rule file is used as in the previous example.

- 
- 8 In both examples in Step # 7, the output files are placed in the `/home/you/named` directory by using the `-o` parameter. They are labeled, as follows:
- `qip-dnssvr.csv` (the DNS server information)
  - `qip-dnsdomain.csv` (the domain information)
  - `qip-dnsnet.csv` (the network information)
  - `qip-dnssn.csv` (the subnet information)
  - `qip-dnsobject.csv` (the object information)
  - `qip-dnsobjrr.csv` (the resource record information associated with objects)
  - `qip-dnszonerr.csv` (the resource record information associated with zones)
  - `general.ext` (additional object alias or subnet domain information)
- There is also a log file, called `qip-dnscsv.log` (the process log), in the `/home/you/named` directory.
- 

- 9 Use your favorite editor to review and modify, as required, before you import.
- 

- 10 Import, using the CLI commands discussed in Chapter 4, “Exporting and importing with CLIs”, on page 337.
- 

END OF STEPS

---

## Synopsis

```
qip-dnscsv -b 4|8|9 -n [named.boot|named.conf]
-r rules_file_name|-m subnet_mask [-d SOA|A110bj]
[-o output_directory] [-c default_object_class] [-v] [-h]
```

*or*

```
qip-dnscsv -b 4|8|9 -i -s dns_svr_IPaddr -z zone_name|
-n named.boot|named.conf [-d SOA|A110bj] [-o output_directory]
[-c default_object_class] [-v] [-h]
```

## Parameters

**qip-dnscsv** recognizes the following parameters:

- b 4|8|9** Specifies the DNS BIND version number. The Values are **4** for BIND 4.9.x, **8** for BIND 8.x, and **9** for BIND 9.x.
- n named.boot|named.conf** Specifies the location of the DNS startup file. This must be the full file name(for example, `n/etc/named/named.boot`).
- Important!** This is optional if running in interactive mode and specifying the zone to be retrieved using the `-z` parameter.

- o output\_directory* Specifies the location of the output files. This must be the full directory name (for example, */home/youroutputfile/named*).
- r rules\_file\_name* Specifies the location of the “rules” file. You must specify the full file name (for example, */home/you/named/dnsrule.dat*).
- m subnet\_mask* Specifies the mask to use to derive network mask length and subnet address.
- d SOA|AllObjects* To create domain records only or from all objects. This can be specified in the rules file also. The default is *AllObjects*.
- c default\_object\_class* If no rule can be found pertaining to the object class, the value supplied here is used as the default object class. If no value is supplied, the class “Undefined” is assigned.
- i* Specifies interactive mode. This specification allows the CLI to retrieve information directly from the server specified by the *-s* parameter.
- s dns\_svr\_IPaddr* Specifies the IP address of the DNS server from which to retrieve the information. This is mandatory if running in interactive mode.
- z zone\_name* Specifies the zone name to be retrieved from the DNS server. This is only valid in interactive mode if the *-n* parameter is not specified.



## Using the “Rules” file with qip-dnscsv

---

### When to use

You must create the “rules” file manually if you want to have the **qip-dnscsv** CLI command retrieve specific information from the DNS Resource Records and place them in the appropriate `.csv` files.

The “rules” file is called whenever **qip-dnscsv** is run. Below are two examples of the “rules” file. The second example uses the Token List so you can “map” the specific data to other data in `qip-dnsobject.csv`.

**Important!** Multiple rules can be defined in one “rules” file.

The “rules” file allows you to:

- 1 Specify how the network and subnet masks are to be used, since that information is not available in most BIND database files.
- 2 Specify how to derive an object class description by mapping information from the A, CNAME or TXT Resource Record. For example, the alias for the **host101** is **gwto101** as it appears on a CNAME record. You can specify a “rule” to search for “gw” at the beginning of a name, and if found, attach the object class description for that object as “Gateway”.

The “rules” file setup must be entered manually in sets of `name=value` pairs. The directory path and file name of the “rules” file is passed to the program as:

```
qip-dnscsv -b 4 -n /etc/named.boot -o /home/usr/qip/csvdir -r /usr/qip/rulesfile -m 255.255.255.0
```

The “name” portion of the pair can be any or all of the names in Table 44 following. The “value” choices for each “name” are discussed in the Value Description column.

**Table 44** “Rules” file definitions

| Name    | Value description                                                                                                                                                                                             | Value example                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CSVFile | This indicates to which CSV file the rule is applicable. Currently, the only choice is <code>SimpleObject</code> .                                                                                            | <code>CSVFile=SimpleObject</code><br>This rule is applicable during the creation of the object CSV file ( <code>qip-dnsobject.csv</code> ).                                                                                                                                                                                                                                                                                                |
| Target  | This indicates to which data item on the output record the rule is applicable. The choices are <code>Comment</code> or <code>ObjectClass</code> .<br>Reference the import CLI “entersimpleobj”, on page 379 . | <code>Target=Comment</code><br>This target places the found information of the Resource Record (for example, TXT record) in the “Comment” field of the <code>qip-dnsobject.csv</code> file.<br><code>Target=ObjectClass</code><br>This target places the found information (based on the value of the Source below) of the Resource Record (for example, TXT record) in the Object Class field of the <code>qip-dnsobject.csv</code> file. |

| Name          | Value description                                                                                                                                                                                                                                                                                                               | Value example                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Source        | <p>The Source is directly related to the Target. The Source value indicates which input record type and field is used to derive the output information. That output information is placed in the Target (either Comment or Object Class). The valid values are TXT&lt;txt-strings&gt; or A&lt;owner&gt;.</p>                    | <p>Source=TXT&lt;txt-strings&gt;<br/>           If the Source is TXT&lt;txt-strings&gt;, the information in this portion of the TXT record is placed in the specified Target (either Object Class or Comment).<br/>           Source=A&lt;owner&gt;<br/>           If the Source is A&lt;owner&gt;, the information in this portion of the A record is placed in the specified Target (either Object Class or Comment). &lt;owner&gt; is the hostname read to the first ".". In other words, if the information in the &lt;owner&gt; field is <i>quad1.quadritek.com</i>, the data used will only be quad1.</p> |
| Method        | <p>This value is directly related to the Source and the Target. This indicates the source information is to be placed in the target field as a whole or is parsed and mapped to some other value based on the provided listing. The choices are Insert or Map.</p>                                                              | <p>Method=Insert<br/>           This indicates the data from the Source is placed in the Target field as a "whole".<br/>           Method=Map<br/>           This indicates that the data from the Source is mapped to a specific value before placing it in the Target field. Refer to Example #2 below for a more complete explanation.</p>                                                                                                                                                                                                                                                                   |
| TokenList     | <p>The Token List lists all the targets (Object Class or Comment) you want to map to other data in the object record. Refer to Example #2 following for more information.</p>                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Network Masks | <p>The network mask length can be specified in the rules file; otherwise, <b>qip-dnscsv</b> will take default action, assuming a standard class A, B, C boundary.</p>                                                                                                                                                           | <p>NetworkMask=begin<br/>           20.0.0=255.255.0.0<br/>           NetworkMask=end</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Subnet Masks  | <p>The subnet mask to be used for a network can be specified in the rules, if it is not just a common mask for subnets. With this rule defined, <b>qip-dnscsv</b> loops through all the Object IPs that have A records, creates subnet IP addresses, and places them in the <b>qip-dnssn.csv</b> file. Refer to Example #2.</p> | <p>SubnetMask=begin<br/>           20.100.23.0=255.255.255.0<br/>           20.100.100.0=255.255.255.0<br/>           SubnetMask=end</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

| Name          | Value description                                                                                                                                                                                                                                                                                                               | Value example                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Source        | <p>The Source is directly related to the Target. The Source value indicates which input record type and field is used to derive the output information. That output information is placed in the Target (either Comment or Object Class). The valid values are TXT&lt;txt-strings&gt; or A&lt;owner&gt;.</p>                    | <p>Source=TXT&lt;txt-strings&gt;<br/>           If the Source is TXT&lt;txt-strings&gt;, the information in this portion of the TXT record is placed in the specified Target (either Object Class or Comment).<br/>           Source=A&lt;owner&gt;<br/>           If the Source is A&lt;owner&gt;, the information in this portion of the A record is placed in the specified Target (either Object Class or Comment). &lt;owner&gt; is the hostname read to the first “.”. In other words, if the information in the &lt;owner&gt; field is <i>quad1.quadritek.com</i>, the data used will only be quad1.</p> |
| Method        | <p>This value is directly related to the Source and the Target. This indicates the source information is to be placed in the target field as a whole or is parsed and mapped to some other value based on the provided listing. The choices are Insert or Map.</p>                                                              | <p>Method=Insert<br/>           This indicates the data from the Source is placed in the Target field as a “whole”.<br/>           Method=Map<br/>           This indicates that the data from the Source is mapped to a specific value before placing it in the Target field. Refer to Example #2 below for a more complete explanation.</p>                                                                                                                                                                                                                                                                   |
| TokenList     | <p>The Token List lists all the targets (Object Class or Comment) you want to map to other data in the object record. Refer to Example #2 following for more information.</p>                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Network Masks | <p>The network mask length can be specified in the rules file; otherwise, <b>qip-dnscsv</b> will take default action, assuming a standard class A, B, C boundary.</p>                                                                                                                                                           | <p>NetworkMask=begin<br/>           20.0.0=255.255.0.0<br/>           NetworkMask=end</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Subnet Masks  | <p>The subnet mask to be used for a network can be specified in the rules, if it is not just a common mask for subnets. With this rule defined, <b>qip-dnscsv</b> loops through all the Object IPs that have A records, creates subnet IP addresses, and places them in the <b>qip-dnssn.csv</b> file. Refer to Example #2.</p> | <p>SubnetMask=begin<br/>           20.100.23.0=255.255.255.0<br/>           20.100.100.0=255.255.255.0<br/>           SubnetMask=end</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

| Name          | Value description                                                                                                                                                                              | Value example                                                                                                                                                                                                                                                                                                                                   |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Domain Record | The Domain Record specification indicates whether to extract domains from SOA only or from all objects. The choices are SOAonly or FromAllObject. Example #1 illustrates the use of this rule. | DomainRecord=SOAonly<br>Indicates extraction from SOA records only. If this is specified and the A record contains an object name with a dotted name, it will be written to the <i>domain.txt</i> file as a resource record instead of having an object record created.<br>DomainRecord=FromAllObject<br>Indicates extraction from all objects. |

**Example #1**

```
DomainRecord=FromAllObject
CSVFile=SimpleObject
Source=TXT<txt-strings>
Target=Comment
Method=Insert
```

The above example in the “rules” file defines that the data in the <txt-strings> portion of the TXT resource record should be placed in the “Comment” field of *qip-dnsobject.csv*. It also specifies that the data should be placed there as a “whole” (Insert).

If the <txt-strings> field of the TXT resource record specifies “This is a wiring hub”, the Comment field in the *qip-dnsobject.csv* record would read exactly that. The *qip-dnsobject.csv* would look like this:

```
usa.world.com,198.200.234.2,hub-ny-1,,Wiring_HUB,This is a wiring hub
CSVFile=SimpleObject
Source=A<owner>
Target=Object Class
Method=Insert
```

The above example in the “rules” file defines that the data in the <owner> portion of the A resource record should be placed in the Object Class field of *qip-dnsobject.csv*. It also specifies that the data should be placed there as a “whole”.

If the <owner> field specifies “This is a wiring hub”, the Object Class field in the *qip-dnsobject.csv* record would read exactly that. The *qip-dnsobject.csv* would look like this:

```
usa.world.com,198.200.234.2,hub-ny-1,,This is a wiring hub
```

**Example #2**

```
;rule1
CSVFile=SimpleObject
Target=Comment
Source=TXT<txt-strings>
Method=Insert
;rule2
CSVFile=SimpleObject
Source=A<owner>
Target=ObjectClass
Method=Map
```

```

TokenList=Begin
Search end=hub=Wiring_HUB
Search begin=gateway=Gateway
Search Anywhere=router=Router
Search Begin=rtr=Router
TokenList=End
;rule3
mask=begin
10=255.255.255.0
10.230.255=255.255.255.240
100=255.255.0.0
74=255.255.0.0
75=255.255.0.0
205=255.255.255.0
mask=end
NetworkMask=begin
209.249.64=255.255.252.0
216.200.28=255.255.252.0
216.200.232=255.255.252.0
209.185.209=255.255.252.0
NetworkMask=end
SubnetMask=begin
209.249.64=255.255.254.0
216.200.28=255.255.252.0
216.200.232=255.255.255.128
209.185.209=255.255.252.0
SubnetMask=end

```

**Important!** Comments must be preceded by # (a pound sign) or ; (a semi-colon).

Example #2, rule 1, indicates that the data in the <txt-strings> portion of the **TXT** Resource Record should be placed in the Comment field of *qip-dnsobject.csv*. It also specifies that the data should be placed there as a “whole” (Insert).

However, the second rule (rule2) states that the <owner> portion of the **A** record should be placed in the Object Class field of *qip-dnsobject.csv*. The information should be replaced with other information (indicated by Method=Map) as per the Token List. All values used to designate Object Class are placed between TokenList=Begin and TokenList=End.

Looking more closely at the Token List example from the above rule, you can see that the program can determine that if any data in the <owner> portion of the **A** record ends with “hub”, it will be replaced in the “Object Class” field of *qip-dnsobject.csv* with the word “Wiring\_HUB”.

```

TokenList=Begin
 Search end=hub=Wiring_HUB
 Search begin=gateway=Gateway
 Search Anywhere=router=Router
 Search Begin=rtr=Router
TokenList=End

```

- If the <owner> portion of the A record contains the word “router” *anywhere* in the data it is replaced with the word “Router” in the “Object Class”.
- If any data in the <owner> portion of the A records *ends* with “hub”, it is replaced with “Wiring\_Hub” in the “Object Class” description.

**Important!** The information in the Token List is case-insensitive.

This example (rule 3) also creates subnet for the associated object as follows, as indicated by the `mask=begin` and `mask=end` code.

- Subnet 10.114.60.0 for object 10.114.60.10
- Subnet 10.230.4.0 for object 10.230.4.20
- Subnet 10.100.0.0 for object 10.100.18.64
- Subnet 205.200.10.0 for object 205.200.10.10

These masks are placed in the *qip-dnssn.csv* file.



## qip-dnscsv output files

---

This section uses examples to help explain the output file layouts for the eight files created by **qip-dnscsv**. As shown in the examples, each field is separated by a comma. For a full explanation of the file formats, refer to Chapter 4, “Exporting and Importing using the VitalQIP CLI”.

### Output File 1 - qip-dnssvr.csv

This file contains the DNS servers found in the DNS database files (obtained mostly from the NS records). To import this information, use the **enterserver** CLI command, specifying *qip-dnssvr.csv* as the value for the **-f** parameter (for example, **-f qip-dnssvr.csv**).

QIPDNS2, qip.com, c:\\qip\\named\\, xyz@lucent.com, 12345

For a description of the contents of the following file, refer to Table 45.

**Table 45 Output File Definitions**

| Record Value   | Field Name                                | Comments                                                                                    |
|----------------|-------------------------------------------|---------------------------------------------------------------------------------------------|
| QIPDNS2        | Host Name                                 |                                                                                             |
| qip.com        | Domain Name                               |                                                                                             |
| c:\\qip\\named | Default Directory                         |                                                                                             |
| xyz@lucent.com | Email address for local and reverse zones |                                                                                             |
| 12345          | RNDC key                                  | Plain text representation of RNDC-key definition in named.conf. Applicable only for BIND 9. |

### Output File 2 - qip-dnsdomain.csv

This file contains the DNS Domain information. To import this information, use the **enterdomain** CLI command, specifying *qip-dnsdomain.csv* as the value for the **-f** parameter (for example, **-f qip-dnsdomain.csv**).

For a description of the contents of the following file, refer to Table 46.

usa.world.com, 21600, 604800, 3600, 86400, root@dns1.world.com, dns1.world.com, P, dns2.usa.world.com, S, dns3.usa.world.com, S, dns4.uk.world.com, S

**Table 46 Output file definitions**

| Record value  | Field name  | Derived from | Comments |
|---------------|-------------|--------------|----------|
| usa.world.com | Domain Name | SOA Record   | Required |

| Record value        | Field name          | Derived from                                                   | Comments                                                   |
|---------------------|---------------------|----------------------------------------------------------------|------------------------------------------------------------|
| 21600               | Refresh Time        | SOA Record                                                     | Required                                                   |
| 604800              | Expire Time         | SOA Record                                                     | Required                                                   |
| 3600                | Retry Time          | SOA Record                                                     | Required                                                   |
| 86400               | Minimum Time        | SOA Record                                                     | Required                                                   |
| root@dns1.world.com | Zone e-mail Address | SOA Record                                                     | Required                                                   |
| dns1.world.com      | Primary Server      | SOA Record                                                     | Optional                                                   |
| P                   | Primary Server Type | From the <b>-b</b> parameter of the <b>qip-dnscsv</b> process. | Required<br>P=primary                                      |
| dns2.usa.world.com  | Additional Server 1 | NS Record                                                      | Optional                                                   |
| S                   | Server Type         |                                                                | Required if a Server 2 exists.<br>P=primary<br>S=secondary |
| dns3.usa.world.com  | Additional Server 2 | NS Record                                                      | Optional                                                   |
| S                   | Server 2 Type       |                                                                | Required if a Server 2 exists.<br>P=primary<br>S=secondary |
| dns4.uk.world.com   | Additional Server 3 | NS Record                                                      | Optional                                                   |
| S                   | Server 3 Type       | NS Record                                                      | Required if a Server 3 exists.<br>P=primary<br>S=secondary |

### Output File 3 - qip-dnsnet.csv

This file contains the DNS network information. Use the **enternetwork** CLI command to import this file, using *qip-dnsnet.csv* as the value for the **-f** parameter (for example, **-f qip-dnsnet.csv**).

The following example is used to explain the output file contents. Note that the example (and Table 47 following) only explains the first row in the file. More “Optional” fields (that can be imported) are available.

```
150.1.6.0,24,root@unilever.com,net_150.1.6.0,N
```

**Table 47** Output file definitions

| Record value | Field name      | Derived from    | Comments |
|--------------|-----------------|-----------------|----------|
| 150.1.6.0    | Network Address | Xxxin-addr.arpa | Required |

| Record value      | Field name          | Derived from     | Comments |
|-------------------|---------------------|------------------|----------|
| 24                | Network Mask length | System-generated | Required |
| root@unilever.com | e-mail address      | System-generated | Required |
| Net_150.1.6.0     | Network Name        | System-generated | Optional |
| N                 | CIDR indicator      | System-derived   | Optional |

**Output File 4 - qip-dnssn.csv**

This file contains the DNS subnet information. Use the **entersubnet** CLI command to import this file, using *qip-dnssn.csv* as the value for the -f parameter (for example, -f qip-dnssn.csv).

The following example is used to explain the output file contents. Note that the example (and Table 48 following) only explains the first row in the file. More “Optional” fields (that can be imported) are available.

150.1.6.0,255.255.255.192,150.1.0.0,SN\_150.1.6.0,usa.world.com

**Table 48 Output file definitions**

| Record value  | Field name        | Derived from                       | Comments |
|---------------|-------------------|------------------------------------|----------|
| 150.1.6.0     | Subnet Address    | Zone (startup file)                | Required |
| 255.255.255.0 | Subnet Mask       | System-generated*                  | Required |
| 150.1.0.0     | Network Address   | Reverse Zone                       | Required |
| SN_150.1.60.0 | Subnet Group Name | System-generated*                  | Required |
| usa.world.com | Domain Name       | SOA Record of the DNS startup file | Optional |

**Important!** Derivation of the Network Address\* = The Network address is obtained from the numeric part of the zone name when it is in the form of X.Y.Z.IN-ADDR.ARPA, where X,Y, and Z are numerical. If a mask is not provided in the rules file or passed as a command parameter, and if there are two numeric values in front of IN-ADDR.ARPA, a 255.255.0.0 mask is assumed. If there are three numeric values in front of IN-ADDR.ARPA, a 255.255.255.0 mask is assumed for the network IP of Z.Y.X.0.

Derivation of the Subnet Address\* = The Subnet Address is derived from subnet mask and the object IP address.

**Output File 5 - qip-dnsobject.csv**

The file contains the DNS object information. Use the **entersimpleobj** CLI command to import this file, using *qip-dnsobject.csv* as the value for the -f parameter (for example, -f qip-dnsobject.csv).

The following example is used to explain the output file contents. Note that the example (and Table 49 following) only explains the first row in the file.

```
1980.200.234.2,alps,,world.com,Router,,,,,,,,,router-alps,Router-1
```

Use of \$GENERATE statements in zone files is supported. For example, the following statement:

\$GENERATE 10-20 QIPDNS\$ 10.20.30\$ statement for qip.com zone will result in the following records being created in *qip-dnsobject.csv*:

```
10.20.30.10, QIPDNS10, , , qip.com, Undefined
10.20.30.11, QIPDNS11, , , qip.com, Undefined
10.20.30.12, QIPDNS12, , , qip.com, Undefined
10.20.30.13, QIPDNS13, , , qip.com, Undefined
10.20.30.14, QIPDNS14, , , qip.com, Undefined
10.20.30.15, QIPDNS15, , , qip.com, Undefined
10.20.30.16, QIPDNS16, , , qip.com, Undefined
10.20.30.17, QIPDNS17, , , qip.com, Undefined
10.20.30.18, QIPDNS18, , , qip.com, Undefined
10.20.30.19, QIPDNS19, , , qip.com, Undefined
10.20.30.20, QIPDNS20, , , qip.com, Undefined
```

**Table 49** Output file definitions

| Record Value  | Field Name               | Derived From                       | Comments  |
|---------------|--------------------------|------------------------------------|-----------|
| 198.200.234.2 | Object IP Address        | A Record                           | Required  |
| Alps          | Object (Device) Name     | A Record                           | Required  |
| world.com     | Domain Name              | SOA Record or the DNS startup file | Optional  |
| Router        | Object Class Description | TXT Record or A Record             | Optional  |
| router-alps   | Alias Name 1             | CNAME Record                       | Optional. |
| Router-1      | Alias Name 2             | CNAME Record                       | Optional. |

#### Output File 6 - qip-dnsobjrr.csv

This file contains the DNS resource record information. Use the **enterdnsrr** CLI command to import this file, using *qip-dnsobjrr.csv* as the value for the **-f** parameter and object as the owner type (in the **-t** parameter); for example, **-f qip-dnsobjrr.csv -t object**.

The following example is used to explain the output file contents. Note that the example (and Table 50 following) only explains the first row in the file. More “Optional” fields (that can be imported) are available.

```
150.1.6.10,pco10.qtek.com,,TXT,,"This is a test field"
```

**Table 50 Output file definitions**

| Record value         | Field name           | Derived from    | Comments                          |
|----------------------|----------------------|-----------------|-----------------------------------|
| 150.1.6.10           | Owner IP             | A record        | The owner of the resource record. |
| pc010.qtek.com       | Owner Name           | A record        |                                   |
| Txt                  | Resource Record type | Resource Record |                                   |
| This is a text field | Resource Record text | Resource Record |                                   |

**Output File 7 – qip-dnszonerr.csv**

This file consists of resource records that are associated with zones. Use the **enterdnsrr** CLI command to import this file, using *qip-dnszonerr.csv* as the value for the -f option and -t domain as the owner type (for example, -f qip-dnszonerr -t domain).

```
Qtek.com.,,pc010.qtek.com,,MX,,"This is a test field"
```

**Output File 8 - general.ext**

This file consists of object alias records that the **qip-dnscsv** utility associated to an IP address, but there were too many aliases to be written to the *qip-dnsobject.csv* file. The records contain that object IP address with a list of aliases. Refer to **enteralias** for further details.

```
198.200.138.10,mail.city.com, cckk.net
```



## Importing DNS files

---

The DNS CSV files must be imported in the following sequence:

1. *qip-dnssvr.csv* is imported into VitalQIP by using **enterserver**.
2. *qip-dnsdomain.csv* is imported into VitalQIP by using **enterdomain**. (If there are additional DNS servers specified in the record, those DNS servers must exist.)
3. *qip-dnsnet.csv* is imported into VitalQIP by using **enternetwork**.
4. *qip-dnssn.csv* is imported into VitalQIP by using **entersubnet**.
5. *qip-dnsobject.csv* is imported into VitalQIP by using **entersimpleobj**.
6. *qip-dnsobjrr.csv* is imported into VitalQIP by using **enterdnsrr**.
7. *qip-dnszonerr.csv* is imported into VitalQIP by using **enterdnsrr**.
8. *general.ext* is imported into VitalQIP by using **enteralias**.

For detailed information of these import CLI commands, refer to the previous section.



## Exporting Bootptab files

---

The Bootptab File Export utility CLI command, **qip-bootptabcsv**, searches the Bootptab file and captures Bootp object and subnet information in CSV formats. These files are created in the same format as the import utilities currently available in VitalQIP.

**Important!** Mac addresses in the bootptab file are required to have a '0x' prefix, as is the case with the VitalQIP-generated bootptab files. When using the **qip-bootptabcsv** CLI with bootptab files from other servers this is not always the case and can result in the MAC address being truncated. Lucent recommends you check the bootptab file and add a '0x' prefix where necessary.

**COMMENT:** This note addresses QVIP00009335.

The import of the Manual Bootp object information is performed using the **qip-setobject** import module. The Network information is imported using the **enternetwork** module. The Subnet information is imported using the **entersubnet** module.



## qip-bootptabcsv

---

**qip-bootptabcsv** reads the Bootptab file and creates the following output CSV files, if possible:

- *qip-bootpobject.fmt* - This file contains the Object information *field names* from the Bootptab file for the **qip-setobject** import file.
- *qip-bootpobject.csv* - This file holds the *values* of the Object information field names as listed in the *qip-bootpobject.csv* file.
- *qip-bootpnet.csv* - This file contains the network information in the format required by the **enternetnetwork** module.
- *qip-bootpsubnet.csv* - This file contains the subnet information in the format required by the **entersubnet** module.

### Synopsis

```
qip-bootptabcsv [-i bootptab_file_name] [-o output_file_directory]
 [-d default_domain] [-v] [-h]
```

### Parameters

**qip-bootptabcsv** recognizes the following parameters:

- i *bootptab\_file\_name*     **Required.** Specifies the location and file name of the Bootptab file. This must be the full file name (for example: */etc/boot/bootptabfile*).
- o *output\_file\_directory* **Required.** Specifies the location (Directory name) of the output files. This must be the full file name (for example, */home/your/bootp*).
- d *default\_domain*         **Optional.** Specifies the default domain name. This must be the full file name (for example, *qtk.com*).

# Using qip-bootptabcsv

---

## When to use

This section describes how to use the **qip-bootptabcsv** CLI. First, you need to gather certain information and do some preparation.

## Procedure

Before using the **qip-bootptabcsv** CLI, do the following:

- 1 Find out the location and the file name of the bootptab file (for example, */etc/bootp/bootptab*).

---
- 2 Decide where the output of the process should go (for example, */home/your/bootp*).

---
- 3 You must provide a default domain name (for example, *qtek.com*) in case records within the bootptab file are missing a domain. Decide what you want the default domain name to be.

---
- 4 Make sure that **qip-bootptabcsv** is in your path.

---
- 5 Initiate the process by typing the command in the command line. For example:  
`qip-bootptabcsv -i /etc/bootp/bootptabfile -o /home/your/bootp -d qtek.com`

---
- 6 The output files are stored in the output directory you specified above with the following names:  
*qip-bootpnet.csv, qip-bootpsn.csv, qip-bootpobject.fmt, qip-bootpobject.csv and qip-bootptabcsv.log*

---
- 7 Use **qip-setobject** to import the data. Refer to **qip-setobject** for details.

END OF STEPS

---



## qip-bootptabcsv output files

---

Here are some sample file layouts for the output files created by **qip-bootptabcsv**. Each field is separated by a comma. For a full description of the file formats, refer to “The Import Utilities” section.

### Output File 1 - qip-bootpobject.fmt

The *qip-bootpobject.fmt* file is the file that contains the object field names (not the values), which are then imported using **qip-setobject**. Some of the values for this file are derived using the Bootptab file tags. More “Optional” fields, which can be imported, are available. See Chapter 4 for more information.

### Example

`ObjectAddress,ObjectName,DomainName,MACAddress,ObjectClass,TftpServer,DefaultRouters,DynamicConfig,HardwareType,BootFileName`

**Table 51** Output file definitions

| Value in the record | Field name     | Derived from                           | Bootptab file tag                                         | Comments                                      |
|---------------------|----------------|----------------------------------------|-----------------------------------------------------------|-----------------------------------------------|
| ObjectAddress       | ObjectAddress  | Bootptab file                          | ip                                                        | Required                                      |
| ObjectName          | Object Name    | Bootptab file                          |                                                           | Required                                      |
| DomainName          | DomainName     | Bootptab file                          | dn or the -d parameter of the <b>qip-bootpcsv</b> process | Required                                      |
| MACAddress          | MACAddress     | Bootptab file                          | ha                                                        | Required                                      |
| ObjectClass         | ObjectClass    | Bootptab file                          | ht                                                        | Optional                                      |
| TftpServer          | TftpServer     | Bootptab file                          | ts                                                        | Optional                                      |
| DefaultRouters      | DefaultRouters | Bootptab file                          | gw                                                        | Optional                                      |
| DynamicConfig       | DynamicConfig  |                                        |                                                           | Optional. This is assumed to be Manual Bootp. |
| HardwareType        | HardwareType   | Bootptab file                          | ht                                                        | Optional                                      |
| BootFileName        | BootFileName   | The CLI <b>qip-bootptabcsv</b> command | hd and bf                                                 | Required                                      |

## Output File 2 - qip-bootpobject.csv

The *qip-bootpobject.csv* file is the file that contains the *values* of the object field names, which are then imported using **qip-setobject**. Table 52 only explains the first row in the file. Some of the values for this file are derived using the Bootptab file tags. More “Optional” fields, which can be imported, are available. Refer to Chapter 4, “Exporting and importing with CLIs” for more information.

### Example

```
198.200.41.64,pllcilas1,qtek.com,0800094b72fc,,198.200.41.1,Manual_Bootp,ethernet,,
198.200.35.147,pl1bo1as3,qtek.com,0800095dc634,,198.200.35.1,Manual_Bootp,ethernet,,
198.200.35.146,pl1bo1as2,qtek.com,0800095d0199,,198.200.35.1,Manual_Bootp,ethernet,,
198.200.35.145,pl1bo1as1,qtek.com,0800095d01d9,,198.200.35.1,Manual_Bootp,ethernet,,
```

**Table 52** Output file definitions

| Value in the record | Field name     | Derived from                    | Bootptab file tag | Comments                                      |
|---------------------|----------------|---------------------------------|-------------------|-----------------------------------------------|
| 198.200.41.64       | ObjectAddress  | Bootptab file                   | ip                | Required                                      |
| pllcilas1           | Object Name    | Bootptab file                   |                   | Required                                      |
| qtek.com            | DomainName     | Bootptab file                   | dn                | Required                                      |
| 0800094b72fc        | MACAddress     | Bootptab file                   | ha                | Optional                                      |
|                     | ObjectClass    | Bootptab file                   | ht                | Optional                                      |
|                     | TftpServer     | Bootptab file                   | ts                | Optional                                      |
| 198.200.41.1        | DefaultRouters | Bootptab file                   | gw                | Optional                                      |
| Manual_Bootp        | DynamicConfig  |                                 |                   | Optional. This is assumed to be Manual Bootp. |
| ethernet            | HardwareType   | Bootptab file                   | ht                | Optional                                      |
|                     | BootFileName   | The CLI qip-bootptabcsv command | hd and bf         | Required                                      |

## Output File 3 - qip-bootpnet.csv

The *qip-bootpnet.csv* file is the file that contains the network information, which are then imported using **enternetwork**. Table 53 explains the first row in a *qip-bootpnet.csv* file. There are more “Optional” fields, which can be imported. Refer to Chapter 4 for more information.

### Example

```
161.251.0.0,24,root@unilever.com,net_161.251.0.0,Y
```

**Table 53** Output file definitions

| Record value      | Field name      | Derived from                                | Comments |
|-------------------|-----------------|---------------------------------------------|----------|
| 161.251.0.0       | Network Address | Bootptab file ip and sm bootptab file tags. | Required |
| 24                | Mask bit length | Bootptab file (sm tag)                      | Required |
| root@unilever.com | e-mail address  | System-generated*                           | Required |
| net_161.251.0.0   | Network Name    | System-generated*                           | Optional |
| Y                 | Cidr indicator  | System-derived                              | Optional |

**Output File 4 - qip-bootpsn.csv**

The *qip-bootpsn.csv* file is the file that contains the subnet information, which is then imported by using **entersubnet**. Table 54 explains the first row in a *qip-bootpsn.csv* file. There are more “Optional” fields, which can be imported. Refer to Chapter 4 for more information.

**Example**

```
161.251.6.0,255.255.255.0,161.251.0.0,subnet_161.251.6.0,,,,,,,,,qtek.com
```

**Table 54** Output file definitions

| Value in the record | Field name      | Derived from                                                         | Comments |
|---------------------|-----------------|----------------------------------------------------------------------|----------|
| 161.251.6.0         | Subnet Address  | Bootptab file ip and sm bootptab file tags.                          | Required |
| 255.255.255.0       | Subnet Mask     | Bootptab file (sm tag)                                               | Required |
| 161.251.0.0         | Network Address | System-generated                                                     | Required |
| subnet_161.251.6.0  | Subnet Name     | System-generated                                                     | Optional |
| qtek.com            | Domain Name     | Bootptab file or the -d parameter of the <b>qip-bootpcsv</b> process | Optional |

**Important!** The subnet address in the Network Address and Subnet Name fields is derived from the sm tag and the object IP address.



## Importing Bootptab files

---

Network information (*qip-bootpnet.csv*) is imported into VitalQIP by using **enternetwork**. For more detailed information, refer to “enternetwork”, on page 362.

Subnet information (*qip-bootpsn.csv*) is imported into VitalQIP by using **entersubnet**. For more detailed information, refer to “entersubnet”, on page 384.

Manual Bootp object information (*qip-bootpobject.fmt* and *qipbootpobject.csv*) is imported into VitalQIP by using **entersimpleobj**. For more detailed information, refer to “entersimpleobj”, on page 379.



## 4 Exporting and importing with CLIs

### Overview

---

VitalQIP offers commands to export and import data into or from the VitalQIP database. Commands are available to export data for domains, OSPF areas, subnet organizations, subnets, MAC address pools, or objects. Commands also allow the importation of data for domains, OSPF areas, subnet organizations, subnets, MAC address pools, or objects. These import/export commands are available on UNIX and Windows platforms and can be used for upgrades or new installations.

**Important!** In general, import/export commands should NOT be used to modify existing data. The `qip-get*/qip-set*` CLIs may be more appropriate.

**Important!** Lucent Technologies recommends that you test the format of the import of your files with just a few records before attempting an import of many objects.

## Importing files

This section discusses the file formats required for data input using the Import function in the VitalQIP interface. You can import data for domains, OSPF areas, subnet organizations, subnets, MAC address pools, or objects.

The import file format is in CSV format. Throughout tables in this chapter, the following notation is used for input values.

M = Mandatory, O = Optional

**Important!** If optional information (for example, domain, contact, location, servers) is entered as part of the data, it must already exist in the VitalQIP database.



## Order of importing

---

Lucent recommends a logical order of importing data into your database.

1. Servers (**enterserver**)
2. Domains (**enterdomain**)
3. Networks (**enternetnetwork**)
4. Subnets (**entersubnet**)
5. Simple Objects (**entersimpleobj**)

During the import, it is recommended that you redirect error messages to a file to trap the potential error messages issued by the VitalQIP, Sybase and/or Oracle database servers. For example, when you run **enterdomain** to import domain information, run:

```
enterdomain -f datafile.dat -e err.dat
```



## enteraddrange

---

**enteraddrange** is the CLI command for defining managed address ranges for networks, and for defining managed object ranges within subnets. These ranges can then be assigned an administrator to manage them. The input must be in VitalQIP CSV (comma delimited) format, as described in Table 55, “Address range data in VitalQIP CSV format”, on page 340.

### Synopsis

```
enteraddrange -t address_type -f input_filename [-g loginserver]
 [-s servername] [-o organization] [-u username] [-p password]
 [-r reject_file] [-e errmsg_file]
```

### Parameters

**enteraddrange** recognizes the following parameters:

- t *address\_type*      Defines a range within a network or subnet. Options are **network** (define a range of subnets within a network) or **subnet** (define a range of objects within a subnet).
- f *input\_filename*    Specifies the directory and filename of the input data.
- g *loginserver*        Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*        Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization*      Specifies the VitalQIP organization (corporation) name.
- u *username*            Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*            Specifies the password for the associated administrator account.
- r *reject\_file*        Specifies the directory and filename for the rejected records.
- e *errmsg\_file*        Specifies the filename to which this CLI command writes error messages, if they occur. The default is *STDERR*.

**Table 55 Address range data in VitalQIP CSV format**

| <b>Field</b>              | <b>Value type</b>                   | <b>Description</b>                                                                                                  |
|---------------------------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Network/Subnet IP Address | [M] Numeric with decimals (15 char) | Enter the IP address of the network or subnet (for example, 100.12.0.0, which must already be defined in VitalQIP). |
| First Address             | [M] Numeric with decimals (15 char) | Enter the beginning IP address for this range. (for example, 100.12.1.1)                                            |
| Last Address              | [M] Numeric with decimals (15 char) | Enter the ending address for this range. (for example, 100.12.1.255)                                                |

**Important!** Data lines must end with a carriage return, or they cannot be imported.

**Command line input examples**

With the -t option as network:

```
144.144.0.0,144.144.144.100,144.144.144.150
144.144.0.0,144.144.144.151,144.144.144.200
144.144.0.0,144.144.144.201,144.144.144.250
```

With the -t option as subnet:

```
144.144.144.0,144.144.144.1,144.144.144.100
```



## enteralias

---

**enteralias** is the CLI command for importing object aliases.

**Important!** The object included in the alias csv must exist in the database for the alias to be imported. Use **entersimpleobj** or **qipbulkload** to import the object information first.

### Synopsis

```
enteralias -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**enteralias** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data. The format is described in Table 56.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the database server name. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the organization name.
- u *username* Specifies a VitalQIP username.
- p *password* Specifies a VitalQIP password.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages, if they occur. The default is *STDERR*.

**Table 56**    **enteralias file format**

| Field             | Value type                                 | Description                                                                 |
|-------------------|--------------------------------------------|-----------------------------------------------------------------------------|
| Object IP address | Dotted decimals<br>(up to 15 chars length) | The ip address of an existing object with which the aliases are associated. |

| Field | Value type                       | Description                                                                                                                                   |
|-------|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| alias | Alphanumeric<br>(up to 92 chars) | Alias name, can be fully qualified. If more than one exists, each should be separated with a comma (the maximum line length is OS dependent.) |

**Important!** All data lines must end with a carriage return, or they cannot be imported.

**Command line input examples**

199.232.138.33,dev1.qtek.com,dev2.qtek.com,dev3.qtek.com,dev4.qtek.com  
199.232.138.99,tst1.qtek.com

## entercontact

---

**entercontact** is the CLI command for importing contact information. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
entercontact -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**entercontact** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data. The format is described in Table 57.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages, if they occur. The default is *STDERR*.

**Table 57** Contact file format

| Field              | Value type | Description                                    |
|--------------------|------------|------------------------------------------------|
| Contact Last Name  | [M]        | Last name of the contact (for example, Smith). |
| Contact First Name | [M]        | First name of the contact (for example, John). |

| Field                  | Value type | Description                                                       |
|------------------------|------------|-------------------------------------------------------------------|
| Contact e-mail address | [O]        | Email address of the contact (for example, jsmith@quadritek.com). |
| Contact phone number   | [O]        | Contact's phone number (for example, 610-555-1212).               |
| Contact pager number   | [O]        | Contact's pager number (for example, 610-722-1111).               |

**Important!** All data lines must end with a carriage return, or they are not imported.

**Command line input examples**

Smith,John,john.smith@qip.com,888-555-1212,888-800-8888  
 Alberts,Judy,judy.alberts@qip.com,888-555-1213,888-800-8889  
 Doe,John,john.doe@qip.com,888-555-1214,888-800-8890  
 Collins,Steven  
 Majors,Maureen,maureen.majors@qip.com



## enterdnsobj

---

**enterdnsobj** is the CLI command for importing A, MX, HINFO, and CNAME information in DNS BIND format to a VitalQIP dataserver. An A record creates an object for defined subnets. The HINFO record fills in the **Object Description** field of the **Object** tab (in the Object Profile) for an existing object. The CNAME record fills in the **Aliases** tab for an existing object.

For details on importing specific record formats, refer to *DNS and BIND, 4<sup>th</sup> Edition* by Paul Albitz & Cricket Liu, published by O'Reilly & Associates. The input must be in space delimited format.

**Important!** The External Comment, External Timestamp and Tombstoned flags are only set on an external object type and cannot be manually set when adding an object. These flags can be added to resource records.

### Synopsis

```
enterdnsobj -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] -d domain
 [-r reject_file] [-w] [-e errmsg_file]
```

### Parameters

**enterdnsobj** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- d *domain* Specifies the domain name of the domain where you want these objects to be.
- r *reject\_file* Specifies the directory and filename for the rejected records.

- w Overwrites with warning if the addresses already exist.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages, if they occur. The default is *STDERR*.

**Input file content example**

```

;=====; Addresses
; and other host information for zone: quadritek.com
;*****
; A records
;*****
suppaix IN A 198.200.201.167
test4 0 IN A 198.200.138.7
;*****
; CNAME records
;*****
alias1 IN CNAME suppaix1.quadritek.com
full-alias.quadritek.comINCNAMEsuppaix1.quadritek.com
;*****
; Other resource records
;*****
suppaix1.quadritek.comINMX20test4.quadritek.com
suppaix1.quadritek.comINHINFOAIX system

```



## enterdnsrr

---

**enterdnsrr** is the CLI command for importing DNS resource record information in VitalQIP format. The associated objects, domain, or reverse zone must already exist. Also, the input data must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterdnsrr -t owner_type -f input_filename [-g loginserver]
 [-s servername] [-o organization] [-u username] [-p password]
 [-r reject_file] [-e errmsg_file] [-a]
```

### Parameters

**enterdnsrr** recognizes the following parameters:

- t *owner\_type*            Specifies the owner type with which the resource records are associated; for example, an object, a domain, or a reverse zone.
  
- f *input\_filename*       Specifies the directory and filename of the input data. The format is described in Table 58.
  
- g *loginserver*           Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*           Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization*         Specifies the VitalQIP organization (corporation) name.
  
- u *username*             Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*             Specifies the password for the associated administrator account.
  
- r *reject\_file*           Specifies the directory and filename for the rejected records.
  
- e *errmsg\_file*           Takes the directory and filename provided in the command line to output error messages during the process. An example output would be "rec10: the subnet for object 144.144.144.10 does not exist!" The corresponding record #10 will be listed in the reject file as is. The default is *STDERR*.
  
- a                         Creates a PTR record for an entered A record.

**Table 58 DNS resource record file input file format**

| Field                                                                                                                                                                                              | Value type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Owner IP/Name                                                                                                                                                                                      | [M] text                    | Owner of the Resource Record (for example, IP address, if associated with an object; domain name if associated with a domain; and the IP address and mask length if associated with a reverse zone).                                                                                                                                                                                                                                                                                                                                                                   |
| Resource Record Owner                                                                                                                                                                              | [M]text<br>(up to 255 char) | The owner field of the resource record as defined in RFC 1035.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Resource Record Class                                                                                                                                                                              | [O] text<br>(6 char)        | The type of network or software for each class of records. Options are: IN, CS, CH, HS.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Resource Record Type                                                                                                                                                                               | [O]text<br>(10 char)        | The type for the resource record (for example, CNAME, SOA, NS, WKS, etc.)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Minimum Time to live                                                                                                                                                                               | [O]numeric                  | Specifies the time interval (time to live, in seconds) that the resource record can be cached before the source of the information should again be consulted. A zero value is interpreted to mean that the resource record can only be used for the transaction in progress, and should not be cached. For example, SOA records are always distributed with a zero TTL to prohibit caching. Zero values can also be used for extremely volatile data. If no TTL is desired (for example, it never expires), then this field should be left blank. (Refer to RFC 1035). |
| Resource Record Text                                                                                                                                                                               | [M]text<br>(up to 255 char) | The <rdata> of the resource record, as defined in RFC 1035.<br><br><b>Note:</b> If the text field consists of double quotes, each double quote should be paired with another double quote, and the entire field should be enclosed in a set of double quotes. For example, the text field entry for "N3" is ""N3""                                                                                                                                                                                                                                                     |
| Place in                                                                                                                                                                                           | [O] R or F                  | Forward or reverse zone. Enter R for Reverse zones or F for Forward zones. The default is F (Forward). Applicable for object resource records only.                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Action Flag                                                                                                                                                                                        | [O] A or D                  | Add or delete the record. The default is Add.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Important!</b> The following four fields are used when adding resource records to the domain or reverse zones <i>only</i> . These flags are ignored when adding a resource record to an object. |                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| External Flag                                                                                                                                                                                      | [O] 0 or 1                  | 0 indicates it is not an EDUP record, 1 indicates that is is an EDUP record.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Tombstoned                                                                                                                                                                                         | [O] 0 or 1                  | 1 indicates that this is a tombstoned external record. 0 indicates that it is an active record.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

| Field              | Value type                      | Description                                                                                               |
|--------------------|---------------------------------|-----------------------------------------------------------------------------------------------------------|
| External Comment   | [O] text (up to 255 characters) | A comment indicating the history of this external add.                                                    |
| External Timestamp | [O] numeric                     | Specifies the time in seconds since January 1, 1970 that an external DNS authority mentioned this record. |

**Important!** All data lines must end with a carriage return, or they are not imported.

### Input file format examples

- For *owner\_type* OBJECT (two examples):

```
150.1.6.4,dns02.lu.com,IN,TXT,200,""Location""Floor 3"",F,A
150.183.140.2,lab2,IN,A,,150.183.140.2,F,0,0,,
```

- For *owner\_type* DOMAIN (two examples):

```
qtek.com,dns.usa.com,IN,TXT,200,""Location""Floor 3"",A
plo.com,lab2,IN,A,,150.183.140.2,A
```

- For *owner\_type* Reverse Zone (two examples):

```
150.246.40.0/24,us.test.com,IN,HINFO,,""Apple Macintosh""OS 8.1, FileMaker Pro"",A
113.33.0.0/16,113.33.132.in-addr.arpa:1,IN,PTR,g01c.qns.qtek.com,A
```



## enterdomain

---

**enterdomain** is the CLI command for importing domain and DNS zone option information. The input must be in VitalQIP CSV (comma delimited) format.

**Important!** The DNS server included in the domain csv must exist in the database for the domain and associated DNS zone options to be imported. Use **enterserver** to import the DNS server information first.

### Synopsis

**Important!** You must specify `-df` and an appropriate value.

```
enterdomain -n [folder_name] -w [-g loginserver] [-s dataserver]
[-u username] [-p password] [-o organization] -i input_file
-df dataformat
```

### Parameters

**enterdomain** recognizes the following parameters:

|                                       |                                                                                                                   |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-n <i>folder_name</i></code>    | Specifies the folder name (full path) to which the domains in the input file ( <b>-i</b> parameter) are assigned. |
| <code>-w</code>                       | Overwrite if the record already exists.                                                                           |
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>dataserver</i></code>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| <code>-p <i>password</i></code>       | Specifies the password for the associated administrator account.                                                  |
| <code>-o <i>organization</i></code>   | Specifies the VitalQIP organization (corporation) name.                                                           |
| <code>-i <i>input_filename</i></code> | Specifies the directory and filename of the input data. The format is described in Table 59.                      |

`-df dataformat`

Specifies the input file format. Possible values are `c` for common header and `sh` for separate header.

If `-df c` is specified, a common header is used for all the domains in the input file, and for all the domains, only those fields shown in the Domain Profile in the VitalQIP GUI are included in the input file.

If `-df sh` is specified, a separate header is used for each domain and each domain is separated by a space line in the input file. The zone option fields will also be included in the input file. If any DNS server is associated with a domain, that DNS server information will also be included in the input file.

**Table 59 Domain and DNS Zone data file format**

| Field                   | Value type               | Description                                                                                                                                                             |
|-------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Domain Name             | [M] text(190 chars)      | Specifies a fully qualified domain name (for example, mydomain.qtek.com).                                                                                               |
| Refresh Time            | [M] numeric (in seconds) | Dictates how often the secondary server should verify its data against the primary server.                                                                              |
| Expire Time             | [M] numeric (in seconds) | If the secondary server has been able to contact the primary server within the 'expire' period, the secondary server will shut down since its data will be out of date. |
| Retry Time              | [M] numeric (in seconds) | Dictates the interval for attempting to refresh in the event that the primary server is unavailable.                                                                    |
| Default TTL             | [M] numeric (in seconds) | Defines the time interval for other servers to cache all resource records in the database file if the TTL is not defined at the object level.                           |
| Negative Cache TTL      | [M] numeric (in seconds) | Defines the time interval in which to cache all negative responses from the name servers authoritative for that zone.                                                   |
| Zone e-mail address     | [M] text (62 char)       | Specifies the e-mail address where the errors caused by DNS are posted.                                                                                                 |
| Start Zone Options      | [O]                      | There is never any data in this field, it is a placeholder only. See "Input file layout", on page 355 for details                                                       |
| Extensions              | [O]                      | There is never any data in this field. It is a placeholder only. See "Input file layout", on page 355 for details.                                                      |
| Prefix of zone db file  | [O] text                 | Text is displayed before the domain in the zone file and is added to the domain zone file exactly as it is entered.                                                     |
| Postfix of zone db file | [O] text                 | Text appears after the domain in the zone file and is added to the domain zone file exactly as it is entered.                                                           |
| Bind-8.x Options        | [O]                      | There is never any data in this field. It is a placeholder only. See "Input file layout", on page 355 for details.                                                      |

| Field                    | Value type | Description                                                                                                                                                                                                                                                                                                                                                           |
|--------------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                                       |
| ACL Templates            | [O] text   | This field is only available when its parent parameter is set to "Use List". It allows you to select a predefined ACL template in VitalQIP.<br><br><b>Important!</b> This field may appear in any Zone Options (BIND-9.X options, Lucent DNS 3.X options, Lucent DNS 4.X options, or Windows 2000 DNS options) where the parent zone option value is set to Use List. |
| other                    | [O] text   | This field is only available when its parent parameter is set to "Use List" and it allows you to enter free form text.<br><br><b>Important!</b> This field may appear in any Zone Options (BIND-9.X options, Lucent DNS 3.X options, Lucent DNS 4.X options, or Windows 2000 DNS options) where the parent zone option value is set to Use List.                      |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                           |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                         |
| check-names              | [O] text   | Specifies the type of name verification processing. The options are: Warn, Fail, Ignore, Use Server Value.                                                                                                                                                                                                                                                            |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Use Server Value.                                                                                                                                                                                                                                                                                |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the <i>named.conf</i> file.                                                                                                                                                                                                                                                                  |
| BIND-9.X Options         | [O]        | There is never any data in this field, it is a placeholder only. See "Input file layout", on page 355 for details.                                                                                                                                                                                                                                                    |
| allow-notify             | [O] text   | Specifies whether to accept notification messages from name servers other than the configured master name server for a zone. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                            |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                                       |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                           |

| Field                    | Value type | Description                                                                                                                                                                                                |
|--------------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                              |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Explicit, Use Server Value.                                                                                                           |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the <i>named.conf</i> file.                                                                                                       |
| LUCENT DNS 3.X Options   | [O]        | There is never any data in this field. It is a placeholder only. See “Input file layout”, on page 355 for details.                                                                                         |
| Import External Updates  | [O] text   | Specifies whether this domain is EDUP enabled. The options are true or false.                                                                                                                              |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                            |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                              |
| check-names              | [O] text   | Specifies the type of name verification processing. The options are: Warn, Fail, Ignore, Use Server Value.                                                                                                 |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Use Server Value.                                                                                                                     |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the <i>named.conf</i> file.                                                                                                       |
| LUCENT DNS 4.X Options   | [O]        | There is never any data in this field. It is a placeholder only. See “Input file layout”, on page 355 for details                                                                                          |
| Import External Updates  | [O] text   | Specifies whether this domain is EDUP enabled. The options are true or false.                                                                                                                              |
| allow-notify             | [O] text   | Specifies whether to accept notification messages from name servers other than the configured master name server for a zone. The options are: Any, None, localhost, localnets, Use List, Use Server Value. |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                            |

| Field                    | Value type             | Description                                                                                                                                                   |
|--------------------------|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| allow-transfer           | [O] text               | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.   |
| allow-update             | [O] text               | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value. |
| notify                   | [O] text               | Specifies whether to perform notification. The options are: No, Yes, Explicit, Use Server Value.                                                              |
| zone block of named.conf | [O] text               | Any text typed in this free text field will appear in the zone blocks of the <i>named.conf</i> file.                                                          |
| WINDOWS 2000 DNS Options | [O]                    | There is never any data in this field, it is a placeholder only. See “Input file layout”, on page 355 for details.                                            |
| aging                    | [O] text               | Specifies whether the aging and scavenging are enabled for this zone. The options are: false, true, Use Server Value.                                         |
| allow-transfer           | [O] text               | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, Name Servers Only, Use List, Use Server Value.      |
| allow-update             | [O] text               | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: no, yes, Use Server Value.                                   |
| no-refresh-interval      | [O] numeric (in hours) | Specifies the no refresh interval for scavenging. If the value is set to 0, the value will not be pushed.                                                     |
| notify                   | [O] text               | Specifies whether to perform notification. The options are: No, Yes, Use List, Use Server Value.                                                              |
| refresh-interval         | [O] numeric (in hours) | Specifies the refresh interval for scavenging. If the value is set to 0, the value will not be pushed.                                                        |
| zone-options             | [O] text               | This is free form text where information can be entered in dnscmd format.                                                                                     |
| End Zone Options         | [O]                    | There is never any data in this field. It is a placeholder only. See “Input file layout”, on page 355 for details.                                            |
| Start DNS Servers        | [O]                    | There is never any data in this field. It is a placeholder only. See “Input file layout”, on page 355 for details.                                            |
| Primary DNS Server Name  | [O] text               | Specifies the fully qualified name of a primary DNS server for this domain.                                                                                   |
| DNS Server Name          | [O] text (253 char)    | Specifies the fully qualified name of an authoritative DNS server for this domain.                                                                            |

| Field               | Value type  | Description                                                                                                                                                   |
|---------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DNS Server Type     | [O] text    | Specifies whether the server is a primary or a secondary server for the zone. The Options are: P, S.                                                          |
| Sec DNS Update Flag | [M] numeric | Specifies whether the name server is secure and if it is secure, then the dynamic updates to the server for this zone must be secure. The options are: 1 , 0. |
| End DNS Servers     | [O]         | There is never any data in this field. It is a placeholder only. See “Input file layout”, on page 355 for details.                                            |

### Input file layout

The input file for **enterdomain** is data driven. If any Zone Options are specified in the input file, the ‘Start Zone Options’ and the ‘End Zone Options’ placeholder columns are required. The fields between these tags may vary based on the setting of other fields.

For example, the ‘ACL Templates’ and ‘other’ fields are only applicable if their parent’s value (for example, the option ‘allow query’) is set to a value of ‘Use List’. Consequently, if an option is to be set, it should come before the child fields or an error will be generated. If you want any Prefix/Postfix of zone db files, the ‘Extensions’ column must be placed before its prefix and postfix of the zone db file field.

Similarly, if you want to enter any BIND-8.X Options, the ‘BIND-8.X Options’ placeholder column must be placed before any of its actual option columns. The same rule applies to any BIND-9.X Options, LUCENT DNS 3.X Options, LUCENT DNS 4.X Options and WINDOWS 2000 DNS Options.

If any DNS server is associated with a domain, the ‘Start DNS Servers’ and the ‘End DNS Servers’ columns are required (again, to act as placeholders). Additionally, if any DNS server is associated with a domain, the server type of the first server has to be a primary server.

**Important!** All data lines must end with a carriage return, or they are not imported.

### Command line input example

```
enterdomain -n boston -i domn.txt -df c
```

**Important!** A correct input file format is very important to ensure the **enterdomain** CLI runs successfully. Users should run **exportdomain** to get the correct input file format and modify the input file data as needed before running **enterdomain**. You can get both input file formats from **exportdomain** whether or not you use the `-sh` parameter in the command line. Refer to “exportdomain”, on page 412 for further information.



## enterlocalobj

---

**enterlocalobj** is the CLI command for importing object information in Local format. VitalQIP automatically creates appropriate object records in the specified domain and in the appropriate subnets. The object class of these objects is set to undefined. The format of the input file (specified by the `-f` parameter) is the same as that of `/etc/hosts`, as shown below.

**Important!** The input file is space-delimited, not comma-delimited.

### Before you begin

- All data lines must end with a carriage return, or they cannot be imported.
- Within the VitalQIP GUI, the default address option for importing an object is “Overwrite if address exists”. **enterlocalobj** defaults to a warning that duplicates exist, and then skip that address.
- You can add multiple Alias Names. Continue to add Alias names with spaces (or tabs) between them.

### Synopsis

```
enterlocalobj -f input_filename -d domain_name [-g loginserver]
[-s servername] [-o organization] [-u username] [-p password]
[-r reject_file] [-e errmsg_file] [-w]
```

### Parameters

**enterlocalobj** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data. The format is described in Table 60. Note that host files and their directory are used by **enterlocalobj** to input data (for example, `-f /etc/hosts`). If other files are used, an error message is displayed.
- d *domain\_name* Specifies the domain name belonging to the object.
- g *loginserver* Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.

- p *password*            Specifies the password for the associated administrator account.
  
- r *reject\_file*        Specifies the directory and filename for the rejected records.
  
- e *errmsg\_file*        Specifies the filename to which this CLI command writes error messages, if they occur. The default is *STDERR*.
  
- w                        Overwrites with warning if the addresses already exist.

**Table 60     Data in local file format**

| Field       | Value type | Description                                                                                                                                                                                                    |
|-------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IP Address  | [M]        | The IP address of the object.                                                                                                                                                                                  |
| Host Name   | [M]        | The host name of the object.                                                                                                                                                                                   |
| Alias Name1 | [O]        | The alias of the object. If a fully qualified hostname is supplied, the parsing rule will be based on the policy setting of the <i>ALLOW_DOTTED_HOSTNAME</i> policy in the Global Policies portion of the GUI. |
| Alias Name2 | [O]        | An additional alias of the object.                                                                                                                                                                             |
| Alias Name3 | [O]        | An additional alias of the object.                                                                                                                                                                             |

**Input file example**

```
199.16.12.4 ws00050sw johnsws domnws
```



## enterlocation

---

**enterlocation** is the CLI command for importing location information. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterlocation -f input_filename [-g loginserver] [-s servername]
[-o organization] [-u username] [-p password] [-r reject_file]
[-e errmsg_file] [-h] [-v]
```

### Parameters

**enterlocation** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages (if any). The default is *STDERR*.

**Table 61** Location file format

| Field                | Value type | Description                                                                  |
|----------------------|------------|------------------------------------------------------------------------------|
| Room ID or Address 1 | [M]        | The first line of the address of the location. (for example, 400 Lapp Road). |
| Address 2            | [O]        | The second line of the address of the location (for example, Suite 20).      |

| Field   | Value type | Description                                         |
|---------|------------|-----------------------------------------------------|
| City    | [O]        | The city of the location (for example, Malvern).    |
| State   | [O]        | The state of the location (for example, PA).        |
| Zip     | [O]        | The zip code for the location (for example, 19355). |
| Country | [O]        | The country of the location (for example, US).      |

**Important!** All data lines must end with a carriage return, or they are not imported.

**Input file examples**

```
400 Lapp Road,Suite 20,Malvern,PA,19355,US
1 Sycamore Lane,Mailstop 123245AS23,Malvern,PA,19355
```



## entermanufacturer

---

**entermanufacturer** is the CLI command for importing manufacturer information to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
entermanufacturer -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password][-r reject_file]
 [-e errmsg_file] [-h] [-v]
```

### Parameters

**entermanufacturer** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 62** Manufacturer data file format

| Field                    | Value type                  | Description                                                                                                                      |
|--------------------------|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| Manufacturer Name        | [M] Text<br>(up to 48 char) | Enter the manufacturer name of the model being defined, it must already exist in the VitalQIP database for a given object class. |
| Object Class Description | [M] Text (20 char)          | Object Class.                                                                                                                    |

| Field            | Value type                  | Description                                                                                                        |
|------------------|-----------------------------|--------------------------------------------------------------------------------------------------------------------|
| Preferred Prefix | [M] Hexadecimal<br>(6 char) | The prefix of the manufacture's MAC address.                                                                       |
| Model Name       | [O] Text (20 char)          | The model name. (If specified, then it must be followed by the tag name and value.)                                |
| Tag Name         | [O] Text (30 char)          | Model Tag name.                                                                                                    |
| Tag Value        | [O]                         | The value associated with the Tag Name. (You can have multiple pairs of Tag Name and Tag Value for a given model.) |

**Important!** All data lines must end with a carriage return, or they are not imported.

#### Input file examples

```
3Com,Workstation,001122,mod1,mtag1,tagval1,mtag2,tagval2
3Com,Workstation,001122,mod2,mtag21,tagval21,mtag22,tagv22
```



## enternetwork

---

**enternetwork** is the CLI command for importing network information. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enternetwork -f input_filename [-g loginserver] [-s servername]
[-o organization] [-u username] [-p password] [-r reject_file]
[-e errmsg_file] [-v] [-h]
```

### Parameters

**enternetwork** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 63** Network data file format

| Field               | Value type                            | Description                                                                   |
|---------------------|---------------------------------------|-------------------------------------------------------------------------------|
| Network Address     | [M]numeric with decimals<br>(15 char) | The network address you are assigning subnets to (for example, 100.12.128.0). |
| Network mask length | [M] numeric<br>(up to 32 characters)  | The number of bits in the network portion of the address.                     |

| Field           | Value type                   | Description                                                                                                                             |
|-----------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| E-mail Address  | [M]text (63 characters)      | The e-mail address for the reverse zone.                                                                                                |
| Network Name    | [O] text<br>(32 characters)  | A name signifying the network you are building.                                                                                         |
| CIDR            | [O] text<br>(1 characters)   | Whether to support CIDR (Y = yes, N= no).                                                                                               |
| Warning Percent | [O] numeric                  | A warning is posted if the percentage of allocated IP addresses is greater than the percentage defined in this field. Values are 0-100. |
| Warning Type    | [O] numeric                  | The type of warnings to receive:<br>0 = none<br>1 = e-mail<br>2 = visual<br>3 = e-mail & visual                                         |
| DNS Server Name | [O] text<br>(253 characters) | The fully qualified server name authoritative for the reverse zone you created.                                                         |

**Important!** All data lines must end with a carriage return, or they are not imported.

#### Input file examples

```
172.17.0.0,16,root@nsroot.qip.com,QIPNET-1,N,10,2,nsroot.qip.com
172.18.0.0,16,root@nsroot.qip.com,QIPNET-2,N,10,2,nsroot.qip.com
172.19.0.0,16,root@nsroot.qip.com,QIPNET-3,N,10,2,nsroot.qip.com
192.168.0.0,16,root@nsroot.qip.com,QIP-CIDR-NET-4,Y,10,2,nsroot.qip.com
10.0.0.0,8,root@nsroot.qip.com,QIPNET-5,N,0,0
```



## enterorganization

---

**enterorganization** is the CLI command for importing organization information. The input must be in VitalQIP CSV (comma delimited) format. This command inputs the maximum object count.

### Synopsis

```
enterorganization -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-r reject_file] [-e errmsg_file]
```

### Parameters

**enterorganization** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data. The format is described in Table 64.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 64** Organization data file format

| Field                             | Value type           | Description                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Organization ID                   | [M] numeric          | The Organization ID is exported by <i>exportorganization</i> for reference purposes. Therefore, <i>enterorganization</i> expects an Organization ID. However, the Organization ID is not used by <i>enterorganization</i> because of potential conflicts with previously existing Organization IDs in the database. When you manually create the datafiles, therefore, you can enter any number for the Organization ID. |
| Organization Name                 | [M] text (32 char.)  | The Organization name.                                                                                                                                                                                                                                                                                                                                                                                                   |
| Description                       | [O] text (255 char.) | The description of the organization.                                                                                                                                                                                                                                                                                                                                                                                     |
| Maximum Number of Objects Allowed | [M] numeric          | Indicates the maximum number of objects allowed in the organization. Zero indicates that there is no limit.                                                                                                                                                                                                                                                                                                              |

**Important!** All data lines must end with a carriage return, or they cannot be imported.

#### Input file examples

```
2,Lucent,"Description of the Lucent Corporation",0
3,VitalQIP,"Description of the VitalQIP Organization",10000
```



# enterospf

---

**enterospf** is the CLI command for importing OSPF information. The input must be in VitalQIP CSV (comma delimited) format.

## Synopsis

```
enterospf -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file] [-w]
```

## Parameters

**enterospf** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.
- w Overwrites warning messages.

**Table 65** OSPF area data file format

| Field          | Value Type         | Description                                                           |
|----------------|--------------------|-----------------------------------------------------------------------|
| OSPF Area Name | [M] text (32 char) | The unique name applied to this OSPF area (for example, OSPF Area 1). |

| Field                              | Value Type                           | Description                                                                                                                                                                 |
|------------------------------------|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OSPF Area ID                       | [O] numeric, with decimals (15 char) | A unique ID that represents the backbone and has already been established (for example, 100.12.128.0).                                                                      |
| OSPF Compliant                     | [O] text (1 char)                    | Whether OSPF Compliant (1=yes, 2=no). The default is 0.                                                                                                                     |
| Area Warning Percent               | [O] numeric, 0-100                   | The threshold for area warnings. Delivers an alarm when the number of managed addresses put into service for this OSPF area reaches the defined threshold.                  |
| Area Warning Type                  | [O] numeric, 0-3                     | The type of warnings to receive:<br>0 = none<br>1 = e-mail<br>2 = visual<br>3 = e-mail & visual                                                                             |
| Managed Subnets                    | [O]                                  | List of the managed subnet IP address surrounded by double quotes.                                                                                                          |
| Start Subnet Range                 | [O] numeric, with decimals           | The start address of the subnet for this OSPF area (for example, 100.12.128.0).                                                                                             |
| OSPF Area Mask or End Subnet Range | [O] numeric, with decimals (15 char) | The above Start Subnet Range field must be coupled with either an OSPF Area Mask or an End Subnet Range. There can be multiple sets of these pairs listed with each record. |

**Important!** All data lines must end with a carriage return, or they are not imported.

### Input file examples

- The following example is based on using the Start Subnet Range and End Subnet Range pairs:

```
OSPF-In1,0.0.0.2,N,85,3,"172.16.64.0,172.16.128.0",172.16.64.1,172.16.64.255,
172.16.128.1,172.16.128.255
```

- The following example is based on using the Start Subnet Range and OSPF Area Mask pairs:

```
OSPF-In2,0.0.0.2,Y,80,2,"172.16.192.0",172.16.192.0,255.255.192.0
```



## enterreversezone

---

**enterreversezone** is the CLI command for importing reverse zones (split zones) information to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterreversezone -df dataformat -i input_filename [-bps] [-g
 loginserver] [-s servername] [-o organization] [-u username]
 [-p password]
```

### Parameters

**enterreversezone** recognizes the following parameters:

- |                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>-df <i>dataformat</i></code>    | Specifies the input file format: If <code>-df c</code> is specified, then a common header is used for all the reverse zones in the input file, and for all the reverse zones, only those fields shown in the reverse zone profile window in VitalQIP GUI will be included in the input file. If <code>-df sh</code> is specified, then a separate header is used for each reverse zone and each reverse zone record is separated by a space line in the input file. All of the zone option fields will be included in the input file. If any DNS server is associated with the reverse zones, that DNS server information will also be included in the input file. |
| <code>-i <i>input_filename</i></code> | Specifies the directory and filename of the input data. Each field in the input file is described in Table 66.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <code>-bps</code>                     | Specifies that for each reverse zone specified in the input file, its associated zone options and DNS server list will be inherited from its parent reversezone.<br><b>Important!</b> When <code>-bps</code> is specified in the command line, the input file must be in “common headers” format, that is, <code>-df c</code> must be specified in the command line.                                                                                                                                                                                                                                                                                               |
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server’s IP address. This value is the equivalent of the LOGIN environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>-s <i>servername</i></code>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>-o <i>organization</i></code>   | Specifies the VitalQIP organization (corporation) name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

-p *password* Specifies the password for the associated administrator account.

**Table 66 Reverse zone data file format**

| Field                    | Value type               | Description                                                                                                                                                             |
|--------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Reverse zone IP Address  | [M] dotted decimal       | The IP address of the reverse zone.                                                                                                                                     |
| Reversezone Address Mask | [M] numeric              | Number of Mask bits for the reverse zone IP address.                                                                                                                    |
| Network Address          | [M] dotted decimal       | The IP address of the network the zone is on.                                                                                                                           |
| Parent Address           | [M] dotted decimal       | The IP address of the parent zone to split.                                                                                                                             |
| Parent Address Mask      | [M] numeric              | Number of Mask bits for the parent zone's address.                                                                                                                      |
| Refresh Time             | [M] numeric (in seconds) | Dictates how often the secondary server should verify its data against the primary server.                                                                              |
| Expire Time              | [M] numeric (in seconds) | If the secondary server has been able to contact the primary server within the 'expire' period, the secondary server will shut down since its data will be out of date. |
| Retry Time               | [M] numeric (in seconds) | Dictates the interval for attempting to refresh in the event that the primary server is unavailable.                                                                    |
| Default TTL              | [M] numeric (in seconds) | Defines the time interval for other servers to cache all resource records in the database file if the TTL is not defined at the object level.                           |
| Negative Cache TTL       | [M] numeric (in seconds) | Defines the time interval in which to cache all negative responses from the name servers authoritative for that zone.                                                   |
| Zone e-mail address      | [M] text (62 char)       | Specifies the e-mail address where the errors caused by DNS are posted.                                                                                                 |
| Start Zone Options       | [O]                      | There is never any data in this field, it is a placeholder only. See Notes * below for details                                                                          |
| Extensions               | [O]                      | There is never any data in this field. It is a placeholder only. See Notes * for details.                                                                               |
| Prefix of zone db file   | [O] text                 | Text appears before the domain in the zone file and is added to the domain zone file exactly as it is entered.                                                          |

| Field                    | Value type | Description                                                                                                                                                                                                                                                                                                                                                                       |
|--------------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Postfix of zone db file  | [O] text   | Text appears after the domain in the zone file and is added to the domain zone file exactly as it is entered.                                                                                                                                                                                                                                                                     |
| Bind-8.x Options         | [O]        | There is never any data in this field. It is a placeholder only. See Notes * for details.                                                                                                                                                                                                                                                                                         |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                                                   |
| ACL Templates            | [O] text   | This field is only available when its parent parameter is set to "Use List" and it allows you to select a predefined ACL template in VitalQIP.<br><b>Important!</b> This field may appear in any Zone Options (for example, BIND-9.X options, Lucent DNS 3.X options, Lucent DNS 4.X options, or Windows 2000 DNS options) where the parent zone option value is set to Use List. |
| other                    | [O] text   | This field is only available when its parent parameter is set to "Use List" and it allows you to enter free form text.<br><b>Important!</b> This field may appear in any Zone Options (for example, BIND-9.X options, Lucent DNS 3.X options, Lucent DNS 4.X options, or Windows 2000 DNS options) where the parent zone option value is set to Use List.                         |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                                       |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                                                                                                                                                                                     |
| check-names              | [O] text   | Specifies the type of name verification processing. The options are: Warn, Fail, Ignore, Use Server Value.                                                                                                                                                                                                                                                                        |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Use Server Value.                                                                                                                                                                                                                                                                                            |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the <i>named.conf</i> file.                                                                                                                                                                                                                                                                              |
| BIND-9.X Options         | [O]        | There is never any data in this field, it is a placeholder only. See Notes* for details.                                                                                                                                                                                                                                                                                          |

| Field                    | Value type | Description                                                                                                                                                                                                |
|--------------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| allow-notify             | [O] text   | Specifies whether to accept notification messages from name servers other than the configured master name server for a zone. The options are: Any, None, localhost, localnets, Use List, Use Server Value. |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                            |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value                                               |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Explicit, Use Server Value.                                                                                                           |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the named.conf file.                                                                                                              |
| LUCENT DNS 3.X Options   | [O]        | There is never any data in this field. It is a placeholder only. See Notes * for details.                                                                                                                  |
| Import External Updates  | [O] text   | Specifies whether this domain is EDUP enabled. The options are true or false.                                                                                                                              |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                            |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                              |
| check-names              | [O] text   | Specifies the type of name verification processing. The options are: Warn, Fail, Ignore, Use Server Value.                                                                                                 |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Use Server Value.                                                                                                                     |

| Field                    | Value type | Description                                                                                                                                                                                                |
|--------------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the named.conf file.                                                                                                              |
| LUCENT DNS 4.X Options   | [O]        | There is never any data in this field. It is a placeholder only. See Noted * below for details                                                                                                             |
| Import External Updates  | [O] text   | Specifies whether this domain is EDUP enabled. The options are true or false.                                                                                                                              |
| allow-notify             | [O] text   | Specifies whether to accept notification messages from name servers other than the configured master name server for a zone. The options are: Any, None, localhost, localnets, Use List, Use Server Value. |
| allow-query              | [O] text   | Specifies which hosts are allowed to ask questions of a resolver. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                            |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                                |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: Any, None, localhost, localnets, Use List, Use Server Value.                                              |
| notify                   | [O] text   | Specifies whether to perform notification. The options are: No, Yes, Explicit, Use Server Value.                                                                                                           |
| zone block of named.conf | [O] text   | Any text typed in this free text field will appear in the zone blocks of the named.conf file.                                                                                                              |
| WINDOWS 2000 DNS Options | [O]        | There is never any data in this field, it is a placeholder only. See Notes * below for details.                                                                                                            |
| aging                    | [O] text   | Specifies whether the aging and scavenging are enabled for this zone. The Options are: false, true, Use Server Value.                                                                                      |
| allow-transfer           | [O] text   | Specifies which hosts are allowed to receive zone transfers from this server. The options are: Any, None, Name Servers Only, Use List, Use Server Value.                                                   |
| allow-update             | [O] text   | Specifies which hosts are allowed to submit dynamic DNS updates to this server. The options are: no, yes, Use Server Value.                                                                                |

| Field                   | Value type             | Description                                                                                                                                                  |
|-------------------------|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| no-refresh-interval     | [O] numeric (in hours) | Specifies the no refresh interval for scavenging. If the value is set to 0, the value will not be pushed.                                                    |
| notify                  | [O] text               | Specifies whether to perform notification. The options are: No, Yes, Use List, Use Server Value.                                                             |
| refresh-interval        | [O] numeric (in hours) | Specifies the refresh interval for scavenging. If the value is set to 0, the value will not be pushed.                                                       |
| zone-options            | [O] text               | This is free form text where information can be entered in dnscmd format.                                                                                    |
| End Zone Options        | [O]                    | There is never any data in this field. It is a placeholder only. See Notes * below for details                                                               |
| Start DNS Servers       | [O]                    | There is never any data in this field. It is a placeholder only. See Notes * below for details                                                               |
| Primary DNS Server Name | [O] text               | Specifies the fully qualified name of a primary DNS server for this domain.                                                                                  |
| DNS Server Name         | [O] text (253 char)    | Specifies the fully qualified name of an authoritative DNS server for this domain.                                                                           |
| DNS Server Type         | [O] text               | Specifies whether the server is a primary or a secondary server for the zone. The Options are: P, S.                                                         |
| Sec DNS Update Flag     | [M] numeric            | Specifies whether the name server is secure and if it is secure, then the dynamic updates to the server for this zone must be secure. The options are:1 , 0. |
| End DNS Servers         | [O]                    | There is never any data in this field. It is a placeholder only. See Notes * below for details.                                                              |

### Input file layout

The input file for **enterreversezone** is data driven. If any Zone Options are specified in the input file, the 'Start Zone Options' and the 'End Zone Options' placeholder columns are required. The fields between these tags may vary based on the setting of other fields.

For example, the 'ACL Templates' and 'other' fields are only applicable if their parent's value (for example, the option 'allow query') is set to a value of 'Use List'. Consequently, if an option is to be set, it should come before the child fields or an error will be generated. If you want any Prefix/Postfix of zone db files, the 'Extensions' column must be placed before its prefix and postfix of the zone db file field.

Similarly, if you want to enter any BIND-8.X Options, the 'BIND-8.X Options' placeholder column must be placed before any of its actual option columns. The same rule applies to any BIND-9.X Options, LUCENT DNS 3.X Options, LUCENT DNS 4.X Options and WINDOWS 2000 DNS Options.

If any DNS server is associated with a domain, the 'Start DNS Servers' and the 'End DNS Servers' columns are required (again, to act as placeholders). Additionally, if any DNS server is associated with a domain, the server type of the first server has to be a primary server.

**Important!** All data lines must end with a carriage return, or they are not imported.



## enterserver

---

**enterserver** allows you to load one or more servers with their definition and parameters into the VitalQIP database.

### Before you begin

- If the **enterserver** command is run with a server name that exists in the VitalQIP database and a zone e-mail address is not specified, no indication is given by the **enterserver** command that it does not exist. It continues processing without providing a notice that the zone e-mail address is missing. Ensure the zone e-mail address is specified when you are adding a server.
- The **enterserver** command modifies servers if the server name specified exists in the database. This functionality is different from the other import commands.

### Synopsis

```
enterserver -f input_filename [-g loginserver] [-s servername]
[-u username] [-p password] [-o organization] [-r reject_file]
[-e errmsg_file]
```

### Parameters

**enterserver** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**enterserver** parses the file until it finds the “Type=” label. When it finds this, it parses out the server type. Based upon the server type, it knows what parameters to expect.

The **Server Type** consists of one of the following :

**DNS:**

- BIND-4.9X
- BIND-8.X
- BIND-9.X
- LUCENT DNS 3.X
- LUCENT DNS 4.X
- MICROSOFT-NT 4.0
- WINDOWS 2000 DNS

**DHCP:**

- LUCENT DHCP 5.3
- LUCENT DHCP 5.4
- IBM AIX DHCP
- MS NT 4.0 DHCP
- IBM NT DHCP
- WINDOWS 2000 DHCP

**Other Server Classes:**

- NIS
- LOCAL HOST
- Bootp
- Windows 2000 DC

The format consists of a record identifier “Format=” label. It is followed by a comma delimited list which indicates which parameters you are including in the CLI to send to the database. ***The first two fields must be the server name and domain name.***

**Input file format**

```
Type=<Server Type><CR>
Format=<Format><CR>
```

```

<Data Values><CR>
<Data Values><CR>
Type=<Server Type><CR>
Format=<Format><CR>
<Data Values><CR>
etc...

```

The `Data Values` are values you wish to submit to the server in the order of the parameter names listed in the `Format` line. If any of those values are incorrect, an error is generated and the CLI fails, for example:

```

Type=Bootp
Host Name, Domain Name, Default Directory, Time Interval
Hostname,quadritek.com,/abc/temp,4500

```

Every parameter and sub-parameter listed in the `Server Profiles` can be specified as a field in the format, except for multi-line fields such as “Additional Policies” for DHCP and “Corporate Extensions” for DNS.

The Lucent DHCP parameters are described in Chapter 5 of the *VitalQIP Administrator Reference Manual* in a section titled “Lucent DHCP Server Type”. The parameters for all other server Policies are described in Chapter 5 of the *VitalQIP Administrator Reference Manual* in a section titled “Defining Servers.”

### Input Parameter Rules

Rules for the input of data are described following; also refer to the input examples.

- The first item must be the server name, and the second item must be the domain name, followed by the rest of the server parameters.
- The server parameter and its sub-parameters must be grouped together; parent followed by child, then by grandchild.
- If a parameter has a parent field with no value, the `, ,` placeholder must be used.
- If the value consists of commas, then the value must be enclosed in a pair of double quotes.
- If a parameter consists of multiple values, they must be delimited by a `|` (pipe).

#### Example 1

```

Type=LUCENT DHCP 5.4
Format=Name,Domain Name,Managed Range,Managed OSPF Area List,.....
dhcp.quadritek.com,OSPF Area,"ospf|ospf test1|ospf (test, 2)",.....

```

#### Example 2

```

Type=WINDOWS 2000 DC
Format=Host Name,Domain Name,LDAP Port,Active Directory Domain Name,User
DN>Password,Scheduled Automatic Updates
dc.quadritek.com,389,quadritek.com,"cn=liz
zar,cn=users,dc=quadritek,dc=com",liz,None,

```

## Input file format

In the following, the “Type” must be the first line of a set of data records, followed by a Format line which specifies the data field name CSVs leading with server and domain name. These are followed by lines of data value CSVs which are in the order specified in the Format line.

```
Type = <Server Type><CR>
Format=<Field,Field><CR>
<Data Values><CR>
<Data Values><CR>
Type = <Server Type><CR>
Format=<Field,Field><CR>
<Data Values><CR>
etc.....
```

### Input example 1

The following example contains the format:

```
Type=LUCENT DHCP 5.4
Format=Host Name,Domain Name,Managed Range,Default Directory,DHCP
Template,Accept Client Names,Client Class,Failover Server Type,Use
Server Policy Template,DHCP 5.2 Server
Policies,ActiveLeaseExpiration,Option81Support,
ShareAutoBootpAndDynDhcp,SupportAutoRelease
```

The following example contains the corresponding values. Note the , , placeholder for Server Policies:

```
dhcpsvr,lucent.com,Corporation,d:/qip/dhcp,general,True,"Test
Class",Standalone/Primary,False,,Off,Suppress,True,False
```

### Input example 2

```
Type=BIND-8.X
Format=Host Name,Domain Name,Default Directory,Email address for local
and reverse zones,Scheduled Automatic Updates,Time of Day,Fully
Managed,Remote Server Proxy
dnssvr,lucent.com,d:/qip/named,admin@lucent.com,By
Day,4:30|18:00,True,200.200.200.200
```



## entersimpleobj

---

**entersimpleobj** is the CLI command for importing object information in VitalQIP format. The input must be in VitalQIP CSV (comma delimited) format.

**Important!** For a more abbreviated import of objects in large numbers, refer to the **qipbulkload** CLI command.

### Synopsis

```
entersimpleobj -f input_filename [-g loginserver] [-s servername]
[-o organization] [-u username] [-p password] [-r reject_file]
[-e errmsg_file] [-w(overwrite)] [-x] [-i]
```

### Parameters

**entersimpleobj** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
  
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- r *rejectfile* Specifies the directory and filename for the rejected records.
  
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.
  
- w(overwrite) Overwrites duplicate addresses.
  
- i Overwrites GAP addresses.

- x

Allows the selected address to be overwritten.

**Table 67 Object data in VitalQIP file format**

| Field                    | Value Type                           | Description                                                                                                                                                                                                                                                                                               |
|--------------------------|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Object IP Address        | [M] numeric, with decimals (15 char) | Enter the IP address of the object (for example, 100.12.128.28).                                                                                                                                                                                                                                          |
| Object(Device) Name      | [M] text (63 char)                   | Enter the object device name for this object. The policy setting ALLOW_DOTTED_HOSTNAMES must be turned on if dotted names are used.                                                                                                                                                                       |
| Subnet Address           | [O] numeric, with decimals (15 char) | Enter the subnet address for this object (for example, 100.12.128.0).                                                                                                                                                                                                                                     |
| DHCP Server Name         | [O] text (253 char)                  | Enter the fully qualified name of the DHCP Server assigned for this object (for example, <i>dhcp1.quadritek.com</i> ).                                                                                                                                                                                    |
| Domain Name              | [O] text (190 char)                  | Enter the fully qualified domain name for this object (for example, <i>qtek.quadritek.com</i> ).                                                                                                                                                                                                          |
| Object Class Description | [O]                                  | refer to the following options:<br>- Workstation<br>- X-terminal<br>- PC<br>- Printer<br>- Server<br>- Wiring_HUB<br>- Router<br>- Bridge<br>- Terminal_Server<br>- Switch<br>- Legacy_System<br>- Gateway<br>- Test_Equipment<br>- Undefined<br>- Others<br>- External<br>-Any user-defined object class |
| Comment                  | [O] text (32 char)                   | Enter a description for this object.                                                                                                                                                                                                                                                                      |
| MAC Address              | [O] 12 or 16, in hexadecimal         | Enter the MAC Address for this object.                                                                                                                                                                                                                                                                    |

| Field                 | Value Type         | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------------------|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Contact Last Name     | [O] text (20 char) | Enter the last name of the contact for this object. If none exists, the Contact Last Name is created.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Contact First Name    | [O] text (20 char) | Enter the first name of the contact for this object. If none exists, the Contact First Name is created.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Contact Phone No.     | [O] text (20 char) | Enter the phone number of the contact for this object. If none exists, the Contact Phone Number is created.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Dynamic Configuration | [O] numeric        | Enter the dynamic configuration for this object. If you do not specify a value, the object is allocated as a static object. refer to the following options:<br>0=None - Allocates this address without using the Bootp or DHCP Protocol.<br>1=M-Bootp - Allocates this address using the Bootp protocol where the MAC address is defined.<br>2=M-DHCP - Allocates this address using the DHCP protocol where the MAC address is defined.<br>3=A-Bootp - Allocates this address using the Bootp protocol where the MAC address is not known.<br>4=A-DHCP - Allocates this address using the DHCP protocol to a DHCP template with an infinite lease.<br>5=D-DHCP - Defines this address using the DHCP protocol to a DHCP template for a specific lease time.<br>-1=Static object or static planned move.<br>-2=Reserves an object. |
| Hardware Type         | [O] numeric        | Enter the hardware type that corresponds to this object. Refer to the following options:<br>0=No hardware type defined<br>1=Ethernet<br>2=Token Ring<br>3=AX.25<br>4=Pronet<br>5=Chaos<br>6=IEEE802<br>7=Arcnet                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Manufacturer          | [O] text (31 char) | Enter the manufacturer of this object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Model Type            | [O] text (15 char) | Enter the manufacturer's model number of this object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Time to live          | [O] numeric        | Enter the amount of time (in seconds) that a DNS server is allowed to cache Resource Record data. If this field is not numeric, it is defaulted to -1.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Alias                 | [O] text (92 char) | Enter one or more aliases for this object (for example, "Alias1, Alias2, Alias3").                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Field                                                        | Value Type             | Description                                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Allow DHCP Clients to Modify Dynamic Object Resource Records | [O] numeric            | Specifies the subnet level setting to keep control and make the DNS updates for DHCP clients. The options are:<br>0 = False<br>1 = True<br>2 = Same as in Global Policies. Defaults to this value if not specified.                                                                       |
| Users List                                                   | [O] text (20 char)     | Enter one or more user's login ID for this object. Enclose a list of users in double quotes (for example, "user1,user2,user3").                                                                                                                                                           |
| User Class                                                   | [O] text (256 char)    | Enter the user class. <b>Only applicable for QDHCP server objects.</b>                                                                                                                                                                                                                    |
| UsageBillServices                                            | alphanumeric           | Determines whether the Usage Billing Service is on or off. The value can be 1=on, Y=on, 0=off, or N=off, depending on the Policy.<br><b>Important!</b> This field may be optional or required, depending on the policy. For more information, refer to the <i>VitalQIP User's Guide</i> . |
| UsageBillLocation                                            | alphanumeric (16 char) | Enter the location name of the Usage Billing Service. The location must exist within the database.<br><b>Important!</b> This field may be optional or required, depending on the policy. For more information, refer to the <i>VitalQIP User's Guide</i> .                                |
| UsageBillUserGroup                                           | alphanumeric (16 char) | Enter the user group of the Usage Billing Service. The User group must exist within the database.<br><b>Important!</b> This field may be optional or required, depending on the policy. For more information, refer to the <i>VitalQIP User's Guide</i> .                                 |
| UsageBillObjectClass                                         | alphanumeric (32 char) | Enter the object class for the Usage Billing Service. The object class must exist within the database.<br><b>Important!</b> This field may be optional or required, depending on the policy. For more information, refer to the <i>VitalQIP User's Guide</i> .                            |

**Important!** All data lines must end with a carriage return, or they are not imported.

**Input file examples**

- To allocate an address for a work station:

```
198.200.234.2,tst2,198.200.234.0,,usa.world.com,Workstation,,,,,3,,,,work-ny
```

- To reserve the address “144.144.144.2”:

```
144.144.144.2,tst2,,qtek.com,Workstation,,,,,-2,,,,work2
```



## entersubnet

---

**entersubnet** is the CLI command for importing Subnet Profile information to an existing network.

**Important!** The Contact, Location, TFTP Server Name, Application, and Subnet are optional information that must already exist in the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
entersubnet -f input_file [-g loginserver] [-s servername]
[-o organization] [-u username] [-p password] [-r reject_file]
[-e errmsg_file] [-w(overwrite)] [-b]
```

### Parameters

**entersubnet** recognizes the following parameters:

- f *input\_file* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.
- w(overwrite) Overwrites warning messages.
- b Not supported.

**Table 68 Subnet profile data file format**

| <b>Field</b>       | <b>Value type</b>                    | <b>Description</b>                                                                                                                                            |
|--------------------|--------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Subnet Address     | [M] numeric, with decimals (15 char) | Enter the subnet address for this Subnet (for example, 100.12.128.0).                                                                                         |
| Subnet Mask        | [M] numeric, with decimals (15 char) | Enter the Subnet Mask for this subnet (for example, 255.255.255.0).                                                                                           |
| Network Address    | [M] numeric, with decimals (15 char) | Enter the Network Address for this subnet (for example, 100.12.128.0).                                                                                        |
| Subnet Name        | [O] text (32 char)                   | Enter the Subnet Name for this Subnet Profile.                                                                                                                |
| Application        | [O] text (30 char)                   | Enter the primary application for this subnet (for example, Engineering).                                                                                     |
| Contact Last Name  | [O] text (20 char)                   | Enter the last name of the contact for this subnet.                                                                                                           |
| Contact First Name | [O] text (20 char)                   | Enter the first name of the contact for this subnet.                                                                                                          |
| Street 1           | [O] text (40 char)                   | Enter the Street 1 information of the location of the objects in this subnet.                                                                                 |
| Street 2           | [O] text (40 char)                   | Enter the Street 2 information of the location of the objects in this subnet.                                                                                 |
| City               | [O] text (20 char)                   | Enter the city of the location of the objects in this subnet.                                                                                                 |
| State              | [O] text (10 char)                   | Enter the state of the location of the objects in this subnet.                                                                                                |
| Zip                | [O] numeric                          | Enter the zip code of the location of the objects in this subnet.                                                                                             |
| Country            | [O] text (30 char)                   | Enter the country of the location of the objects in this subnet.                                                                                              |
| Hardware Type      | [O] numeric                          | Enter the hardware type for this subnet. Refer to the options below:<br>1=Ethernet<br>2=Token Ring<br>3=AX.25<br>4=Pronet<br>5=Chaos<br>6=IEEE802<br>7=Arcnet |
| TFTP Server Name   | [O] text (253 char)                  | Enter the server name that will act as the TFTP server.                                                                                                       |

| Field                                                        | Value type          | Description                                                                                                                                                                                                                                  |
|--------------------------------------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UsageBillLocation                                            | alphanumeric        | Enter the location of the Usage Billing Service. The location must exist within the database.<br><b>Note:</b> This field may be optional or required, depending on the policy. For more information, refer to the VitalQIP GUI help screens. |
| UsageBillUserGroup                                           | alphanumeric        | Enter the user group of the Usage Billing Service. The User Group must exist within the database.<br><b>Note:</b> This field may be optional or required, depending on the policy. For more information, refer to the VitalQIP help screens. |
| Allow DHCP Clients to Modify Dynamic Object Resource Records | [O] numeric         | Specifies the subnet level setting to keep control and make the DNS updates for DHCP clients. The options are:<br>0 = False<br>1 = True<br>2 = Same as in Global Policies. Defaults to this value if not specified.                          |
| Domain                                                       | [O] text (190 char) | Enter a fully qualified domain name for this subnet (for example, qtek.quadritek.com). May have multiple entries. The first domain in a list becomes the default domain for the subnet.                                                      |

**Important!** All data lines must end with a carriage return, or they cannot be imported.

**Input file example**

```

10.1.1.0,255.255.255.0,10.0.0.0,Subnet-1,,,,,,,,,,,,,0,qip.com,
192.168.0.0,255.255.192.0,192.168.0.0,Subnet-2,,,,,,,,,,,,,1,qip.com,north.qip.com,
192.168.64.0,255.255.192.0,192.168.0.0,Subnet-3,,,,,,,,,,,,,2,qip.com,south.qip.com,
192.168.128.0,255.255.192.0,192.168.0.0,Subnet-4,,,,,,,,,,,,,1,qip.com,east.qip.com,
192.168.192.0,255.255.192.0,192.168.0.0,Subnet-5,,,,,,,,,,,,,2,qip.com,west.qip.com,
172.17.1.0,255.255.255.224,172.17.0.0,Subnet-6,,,,,,,,,,,,,1,qip.com,north.qip.com,
south.qip.com,east.qip.com,west.qip.com,
172.17.1.32,255.255.255.224,172.17.0.0,Subnet-7,,,,,,,,,,,,,1,qip.com,
172.17.1.64,255.255.255.224,172.17.0.0,Subnet-8,,,,,,,,,,,,,1,qip.com,
172.17.1.96,255.255.255.224,172.17.0.0,Subnet-9,,,,,,,,,,,,,0,qip.com,
172.17.1.128,255.255.255.224,172.17.0.0,Subnet-10,,,,,,,,,,,,,0,qip.com,
172.17.1.160,255.255.255.224,172.17.0.0,Subnet-11,,,,,,,,,,,,,0,qip.com,
172.17.1.192,255.255.255.224,172.17.0.0,Subnet-12,,,,,,,,,,,,,0,qip.com,
172.17.1.224,255.255.255.224,172.17.0.0,Subnet-13,,,,,,,,,,,,,0,qip.com,

```



## entersubnetorg

---

**entersubnetorg** is the CLI command for importing Subnet Organization information. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
entersubnetorg -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file] [-w(overwrite)]
```

### Parameters

**entersubnetorg** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *rejectfile* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.
- w Specifies that this command overwrites existing data.

**Table 69 Subnet organization data file format**

| Field                    | Value type         | Description                                                |
|--------------------------|--------------------|------------------------------------------------------------|
| Subnet Organization Name | [M] text (32 char) | Specify the name you want to give the Subnet Organization. |

| Field                          | Value type                           | Description                                                                                                                                                                                                                                                                                                                                                                                |
|--------------------------------|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Subnet Address                 | [O] numeric, with decimals (15 char) | Enter the subnet address for this Subnet Organizer (for example, 100.12.128.0). May have multiples of Subnet Address. Multiple subnet addresses must be separated by a comma (,) and enclosed within a set of double quotations (“ ”).                                                                                                                                                     |
| Address Number (GAP)           | [O] numeric                          | If you assign the number 1, for example, with “Router” in the Object Class field below, the Router Object Class will occupy the first IP address in all subnets in this Subnet Group relative to the subnet. (-1 indicates 1 from the end). Multiple sets of Address Numbers/Object Classes may be used. Pairs must be enclosed within a set of (“ ”), for example, “1, Workstation,2,PC”. |
| Object Class (GAP)             | [O] text                             | The Object Class you want to associate with the Address Number. The options are shown below:<br>Workstation<br>X-terminal<br>PC<br>Printer<br>Server<br>Wiring_HUB<br>Router<br>Bridge<br>Terminal_Server<br>Switch<br>Legacy_System<br>Gateway<br>Test_Equipment<br>Undefined<br>Others<br>User-defined                                                                                   |
| Site Name                      | [M] text (32 char)                   | Specify the name you want to give the Windows 2000 site.                                                                                                                                                                                                                                                                                                                                   |
| Site Name Same as Subnetorg    | [O] numeric                          | Determines if the site has the same name as the subnet organization.<br>0=false<br>1=true                                                                                                                                                                                                                                                                                                  |
| Domain Controller Server Names | [M] text (255 char)                  | Specify the Domain Controllers that are to be associated with the subnet organization and managed as a Windows 2000 site within VitalQIP.                                                                                                                                                                                                                                                  |
| DHCP Server Name               | [O] text (255 char)                  | Specify the DHCP server to be associated with the subnet organization.                                                                                                                                                                                                                                                                                                                     |

| Field              | Value type         | Description                                                                                                                                                                                                                                                                                               |
|--------------------|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DHCP Template Name | [O] text (32 char) | Specify the DHCP Option Template to be associated with the subnet organization.                                                                                                                                                                                                                           |
| Warning Type       | [O] numeric        | The warning type you want to use with this subnet organization. The options are shown below:<br>0=The warning type is not being sent.<br>1=The “Visual” check box is being selected.<br>2=The “Email” check box is being selected.<br>3=Both the “Visual” and the “Email” check boxes are being selected. |
| Warning threshold  | [O] numeric        | The threshold this subnet organization will use to send out the warning message. When the percentage of the managed subnets addressed in the associated subnet organization reach this threshold, a warning message will be sent using the method specified in the Warning Type entry.                    |

**Important!** All data lines must end with a carriage return, or they are not imported.

The Address Number and Object Class must be entered in pairs, as shown below.

```
1,pc,
2,workstation,
```

The Address Number and Object Class can be repeated as often as necessary.

**Important!** If using a spreadsheet (for example, Excel) to store **entersubnetorg** data, store each Address Number and Object Class value in its own cell.

If you use the - sign, it indicates one from the end. In the example below, you are assigning the Gateway to the 9<sup>th</sup> subnet address from the end.

#### Input file example

```
Snorg1,"144.144.1.0,144.144.20", "1,pc,2,workstation",site1,0,"dc1.lu.com,dc2.lu.com,
dncpl.lu.com",general,1,95
```



## enterudf

---

The **enterudf** CLI command imports user-defined fields to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterudf -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**enterudf** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 70** UDF data file format

| Field      | Value type          | Description                                                                                                                                                                                                                           |
|------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Owner Type | [M] text (190 char) | The type of user-defined field. Options are: 1=Object, 2=Subnet, 6=Domain, 13=Organization, 16=User, 18=Reverse Zone, 27=Prov_Block*, 26=Prov_Pool*.<br>*Note – These owner types apply only to the Provisioning Manager application. |

| Field              | Value type          | Description                                                                                                       |
|--------------------|---------------------|-------------------------------------------------------------------------------------------------------------------|
| Field name/value   | [M] text (1 char)   | Specify whether this is a field name ( <b>N</b> ) or the value of a field ( <b>V</b> ).                           |
| Field name         | [M] text (30 char)  | If the field name/value is <b>N</b> (for example, field name), specify the field name.                            |
| Owner(username IP) | [M] text (32 char)  | If the field name/value is <b>V</b> (for example, field value), specify the IP address or user name of the field. |
| Field value        | [M] text (128 char) | If the field name/value is <b>V</b> (for example, field value), specify the actual value for the field.           |

**Important!** All data lines must end with a carriage return, or they are not imported.

#### Input file examples

1,n,psl-udf2

1,v,psl-udf2,150.1.6.14,This is the field value of 150-1-6-14



## enteruser

---

**enteruser** is the CLI command for importing user information to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enteruser -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file] [-c] [-w]
```

### Parameters

**enteruser** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- r *reject\_file* Specifies the directory and filename for the rejected records.
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.
- w Overwrites warning messages.
- c This parameter should only be used if the password is not encrypted. The password will be encrypted with the use of the -c parameter.

**Table 71 User data file format**

| <b>Field</b>       | <b>Value type</b>       | <b>Description</b>                                                                                                              |
|--------------------|-------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Log in Name        | [M] text (30 char)      | The name the user enters to login.                                                                                              |
| Last Name          | [M] text (20 char)      | The last name of the user.                                                                                                      |
| First Name         | [M] text (20 char)      | The first name of the user.                                                                                                     |
| User Group         | [M] text (30 char)      | User Group names.                                                                                                               |
| Phone Number       | [O] numeric (20 digits) | The user's phone number.                                                                                                        |
| E-mail Address     | [O] text (63 char)      | The e-mail address of the user.                                                                                                 |
| Password           | [O] text (10 char)      | The password the user uses to login.                                                                                            |
| Pin                | [O] text (30 char)      | The user's Pin number                                                                                                           |
| Description        | [O] text (30 char)      | A description for this user.                                                                                                    |
| Street1            | [O] text (30 char)      | The street address of the user.                                                                                                 |
| Street2            | [O] text (30 char)      | Additional street address information.                                                                                          |
| City               | [O] text (20 char)      | The city the user lives in.                                                                                                     |
| State              | [O] text (10 char)      | The state the user lives in.                                                                                                    |
| Zip                | [O] text (15 char)      | The user's zip code.                                                                                                            |
| Country            | [O] text (30 char)      | The country the user lives in.                                                                                                  |
| Activation Type    | [O] numeric (0-2)       | Activation type, as follows:<br>Active=1<br>Inactive=0<br>Pending=2<br>The default is 0 (inactive).                             |
| Default Subnet     | [O]                     | List of default subnet IP addresses, which are separated by a comma, and placed in double quotes.                               |
| Managed Range      | [O]                     | Groups of MAC addresses, Hostnames or IP addresses. You may have multiple types and sets. Place multiples within double quotes. |
| UsageBillLocation  | alphanumeric            | Enter a Usage Billing Service location.                                                                                         |
| UsageBillUserGroup | alphanumeric            | Enter a Usage Billing Service user group.                                                                                       |
| User Field Name    | [O]                     | The field name of the user field. This must exist if you are going to supply the next field, the User Field Value.              |

| Field            | Value type | Description                                                                                                                          |
|------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------|
| User Field Value | [O]        | The value of the preceding User Field Name. If there is a comma in this value, the whole string must be surrounded by double quotes. |

**Important!** All data lines must end with a carriage return, or they cannot be imported.

**Input file examples**

```
usr5,Jones,"pal,jr","isp_qtek,isp2_qtk",292-1212,plo@qtek.com,password,6666,"adding
managed ranges of the
user",,,,,,1,"144.144.144.0,161.251.6.0","112233445566,test.qtek.com,11.22.33.44",U
DF-1,"the text pertaining to the user field UDF-1"
```



## enterusergrp

---

**enterusergrp** is the CLI command for importing User Group information to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterusergrp -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password][-r reject_file]
 [-e errmsg_file]
```

### Parameters

**enterusergrp** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
  
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- r *reject\_file* Specifies the directory and filename for the rejected records.
  
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 72** User group data file format

| Field         | Value type         | Description                 |
|---------------|--------------------|-----------------------------|
| UserGroupName | [M] text (32 char) | The name of the user group. |

| Field                | Value type         | Description                        |
|----------------------|--------------------|------------------------------------|
| ContactLastName      | [M] text (20 char) | The last name of the contact.      |
| ContactFirstName     | [M] text (20 char) | The first name of the contact.     |
| UserGroupDescription | [O] text (32 char) | The description of the User Group. |

**Important!** All data lines must end with a carriage return, or they are not imported.

**Input file examples**

isp\_qtek,qipman,qipman,big time isp



## enterzoneext

---

**enterzoneext** is the CLI command for importing zone extension information to the VitalQIP database. The input must be in VitalQIP CSV (comma delimited) format.

### Synopsis

```
enterzoneext -f input_filename [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-r reject_file]
 [-e errmsg_file]
```

### Parameters

**enterzoneext** recognizes the following parameters:

- f *input\_filename* Specifies the directory and filename of the input data.
  
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- r *reject\_file* Specifies the directory and filename for the rejected records.
  
- e *errmsg\_file* Specifies the filename to which this CLI command writes error messages if they occur. The default is *STDERR*.

**Table 73 Domain data file format**

| Field     | Value type                | Description                                                                                                             |
|-----------|---------------------------|-------------------------------------------------------------------------------------------------------------------------|
| Zone name | [M] text (up to 256 char) | The fully qualified name of the domain (for example, mydomain.qtek.com) or the IP address and mask of the reverse zone. |

| Field              | Value type | Description                                                                                                                                                                                                                                       |
|--------------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Extension Location | [O] text   | The location of the extension. Valid types are:<br><b>prefix</b> - prefix extensions<br><b>postfix</b> - postfix extensions<br><b>Win2K</b> - Windows 2000 zone extension<br>If no extension location is specified, the postfix location is used. |
| Zone Extension     | [M] text   | Additional text for the extension.                                                                                                                                                                                                                |

**Important!** All data lines must end with a carriage return, or they are not imported.

**Input file example**

```
bank.com,localhost IN A 127.0.0.1, prefix
bank.com,bank.com. IN NS server01.bank.com., prefix
bank.com,bank.com. IN MX 10 email.bank.com., prefix
bank.com,Service-Center IN CNAME Support.Bank.Com., prefix
.
```



## qipbulkload

---

**qipbulkload** pre-loads objects from a comma delimited file that you create. It reads a file in the format shown below and loads the data by using the **qsi-import** routine (refer to the *VitalQIP Installation Guide*). It is, essentially, a condensed version of the **entersimpleobj** import utility. **qipbulkload** is installed in `$QIPHOME/usr/bin` directory on UNIX platforms and in the `%QIPHOME%` directory on Windows platforms.

### Before you begin

- If you build your infrastructure from scratch, **qipbulkload** cannot be used until forward zones have been bound to Subnet Profiles.
- No other VitalQIP process except for the VitalQIP Login Service (GUI, daemon, CLI, etc.), VitalQIPMessage Service and VitalQIP SSL Tunnel Service should be active when **qipbulkload** is run to avoid objects being added from other sources and causing corruption.
- **qsi-import** must be in the same path as **qipbulkload**.
- If records are rejected by the **qipbulkload** CLI command, they are sent to a file (called **bulkload.rej**) that is created automatically, or the record's associated error message is sent to the log file you specified in the `-l <logfile_name>` option.
- **qipbulkload** should only be used for adding objects. Use **qip-setobject** if you wish to modify them.
- Duplicate errors can occur. Do not have duplicates in data as input.

### Synopsis

```
qipbulkload [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] -f input_filename [-t 0|S]
 [-e errmsgfile] [-z] [-r reject_file]
```

### Parameters

**qipbulkload** recognizes the following parameters:

- |                              |                                                                                                                   |
|------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-g loginserver</code>  | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s servername</code>   | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-o organization</code> | Specifies the VitalQIP organization (corporation) name.                                                           |
| <code>-u username</code>     | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |

- p *password*                Specifies the password for the associated administrator account.
  
- f *input\_filename*        Specifies the filename from which the information is read. The default is *STDIN*.
  
- t 0|S                        Specifies either O=Oracle or S=Sybase.
  
- e *errmsgfile*              Specifies the name of the file to contain information concerning the records that could not be processed. The default is *STDERR*.
  
- z                              Specifies that qipbulkload runs in test mode. Specifically, generates the import files but prints the import command to the screen instead of executing the import command. Also, leaves the import files in place upon exit, instead of deleting them. This gives you a chance to look at the import files before actually loading them.
  
- r *reject\_file*              Specifies the file where records that are not added or modified are placed.

**Table 74     Input file format**

| <b>Field</b>        | <b>Value Type</b>                    | <b>Description</b>                                                                                             |
|---------------------|--------------------------------------|----------------------------------------------------------------------------------------------------------------|
| Object IP Address   | [M] numeric, with decimals (15 char) | Enter the IP address of the object (for example, 100.12.128.28).                                               |
| Object(Device) Name | [M] text (63 char)                   | Enter the object device name for this object.                                                                  |
| Subnet Address      | [O] numeric, with decimals (15 char) | Enter the subnet address for this object (for example, 100.12.128.0).                                          |
| DHCP Server Name    | [O] text (253 char)                  | Enter the fully qualified name of the DHCP Server assigned for this object (for example, dhcp1.quadritek.com). |
| Domain Name         | [O] text (190 char)                  | Enter the fully qualified domain name for this object (for example, qtek.quadritek.com).                       |

| <b>Field</b>                | <b>Value Type</b>                   | <b>Description</b>                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----------------------------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Object Class<br>Description | [O]                                 | Refer to the following options:<br><ul style="list-style-type: none"> <li>- Workstation</li> <li>- X-terminal</li> <li>- PC</li> <li>- Printer</li> <li>- Server</li> <li>- Wiring_HUB</li> <li>- Router</li> <li>- Bridge</li> <li>- Terminal_Server</li> <li>- Switch</li> <li>- Legacy_System</li> <li>- Gateway</li> <li>- Test_Equipment</li> <li>- Undefined</li> <li>- Others</li> <li>- External</li> </ul> |
| Comment                     | [O] text<br>(32 char)               | Enter a description for this object.                                                                                                                                                                                                                                                                                                                                                                                |
| MAC Address                 | [O] 12 or 16<br>(in<br>hexadecimal) | Enter the MAC Address for this object.                                                                                                                                                                                                                                                                                                                                                                              |
| Contact Last<br>Name        | [O] text<br>(20 char)               | Enter the last name of the contact for this object.                                                                                                                                                                                                                                                                                                                                                                 |
| Contact First<br>Name       | [O] text<br>(20 char)               | Enter the first name of the contact for this object.                                                                                                                                                                                                                                                                                                                                                                |
| Contact Phone<br>No.        | [O] text<br>(20 char)               | Enter the phone number of the contact for this object.                                                                                                                                                                                                                                                                                                                                                              |

| Field                                                     | Value Type         | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Dynamic Configuration                                     | [O] numeric        | Enter the dynamic configuration for this object. Refer to the following options:<br>0=None - Allocates this address without using the Bootp or DHCP Protocol.<br>1=M-Bootp - Allocates this address by using the Bootp protocol where the MAC address is defined.<br>2=M-DHCP - Allocates this address by using the DHCP protocol where the MAC address is defined.<br>3=A-Bootp - Allocates this address by using the Bootp protocol where the MAC address is not known.<br>4=A-DHCP - Allocates this address by using the DHCP protocol to a DHCP template with an infinite lease.<br>5=D-DHCP - Defines this address by using the DHCP protocol to a DHCP template for a specific lease time. |
| Hardware Type                                             | [O] numeric        | Enter the hardware type that corresponds to this object. Refer to the following options:<br>1=Ethernet<br>2=Token Ring<br>3=AX.25<br>4=Pronet<br>5=Chaos<br>6=IEEE802<br>7=Arcnet                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Manufacturer                                              | [O] text (31 char) | Enter the Manufacturer of this object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Model Type                                                | [O] text (15 char) | Enter the manufacturer's model number of this object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Time to live                                              | [O] numeric        | Specify the amount of time (in seconds) that a DNS server is allowed to cache Resource Record data.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Alias                                                     | [O] text (92 char) | Enter one or more aliases for the object (for example, "Alias1, Alias2, Alias3").                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Allow DHCP Clients Modify Dynamic Object Resource Records | [O] numeric        | Specifies the subnet level setting to keep control and make the DNS updates for DHCP clients. The options are:<br>0 = False<br>1 = True<br>2 = Same as in Global Policies. Defaults to this value if not specified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

**Important!** All data lines must end with a carriage return, or they are not imported.

### Input file examples

```
10.200.90.3,wks1,10.200.90.0,,seg4.qa.quadritek.com,workstation,test
workstation,484453000000,qipman,qipman,,1,,1231,,
```

10.200.90.4,lpt2,10.200.90.0,,seg4.qa.quadritek.com,switch,test  
switch,490003111111,qipman,qipman,,,2,,ACC,1231,,  
10.200.90.5,pc3,10.200.90.0,,seg4.qa.quadritek.com,pc,test  
pc,0000E2222222,qipman,qipman,,,3,,ACER,1231,,  
10.200.90.6,ptr4,10.200.90.0,,seg4.qa.quadritek.com,printer,test  
printer,900010333333,edwards,cindy,555-1212,.4,,AT&T,1231,,  
10.200.90.7,gw5,10.200.90.0,,seg4.qa.quadritek.com,gateway,test  
gateway,eiei0eeeeeee,murphy,paula,,,5,,,1231,,  
10.200.90.8,bdg6,10.200.90.0,,seg4.qa.quadritek.com,bridge,brooklyn  
bridge,eac4daeeffff,schmidt,alexander,,,6,,ALGORITHMS SOFTWARE PVT. LTD.,1231,,  
10.200.90.9,dns1,10.200.90.0,,seg4.qa.quadritek.com,server,the  
dns\_server,,schmidt,alexander,,,7,,,1231,,  
10.200.90.10,hub01,10.200.90.0,,seg4.qa.quadritek.com,wiring\_hub,the  
server,712377fffe77,qipman,qipman,,,1,,,1231,,  
10.200.90.11,hub02,10.200.90.0,,seg4.qa.quadritek.com,wiring\_hub,the  
wiring\_hub,87eeddaadd33,murphy,paula,,,2,,,1231,,  
10.200.90.12,rtr01,10.200.90.0,,seg4.qa.quadritek.com,router,the  
router,,newman,alfred,,,3,,,1231,,  
10.200.90.13,tserver01,10.200.90.0,,seg4.qa.quadritek.com,terminal\_server,the  
terminal\_server,98700aaaaadd,qipman,qipman,,,4,,,1231,,  
10.200.90.14,usa1,10.200.90.0,,seg4.qa.quadritek.com,legacy\_system,the  
legacy\_system,acel234deac7,edwards,cindy,,,5,,,1231,,  
10.200.90.15,testpack01,10.200.90.0,,seg4.qa.quadritek.com,test\_equipment,the  
test\_equipment,ccaeeedd1122,schmidt,alexander,,,6,,,1231,,



## qiploadsndomn

---

**qiploadsndomn** loads subnet domains from a comma delimited file that you create. It reads a file in the format shown below and loads the data using the **qsi-import** routine (refer to the *VitalQIP Installation Guide* for more information on this routine). **qiploadsndomn** is installed in `$QIPHOME/usr/bin` directory on UNIX platforms, or `%QIPHOME%` directory on Windows platforms. It must be in the same directory as **qsi-import**.

### Before you begin

- No other VitalQIP process (for example, GUI, daemon, CLI) should be active when **qiploadsndomn** is run.
- **qsi-import** must be in the same path as **qiploadsndomn**.
- The subnets that are included in the input file must have been defined in VitalQIP.

### Synopsis

```
qiploadsndomn [-g loginserver] [-s servername] [-o organization]
[-u username] [-p password] -f input_filename [-t O|S]
[-e errmsgfile] -z
```

### Parameters

**qiploadsndomn** recognizes the following parameters:

- |                                       |                                                                                                                   |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>-g <i>loginserver</i></code>    | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| <code>-s <i>servername</i></code>     | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| <code>-o <i>organization</i></code>   | Specifies the VitalQIP organization (corporation) name.                                                           |
| <code>-u <i>username</i></code>       | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| <code>-p <i>password</i></code>       | Specifies the password for the associated administrator account.                                                  |
| <code>-f <i>input_filename</i></code> | Specifies the file from which the information is read. The default is <i>STDIN</i> .                              |
| <code>-t O S</code>                   | Specifies either <b>O</b> =Oracle or <b>S</b> =Sybase.                                                            |

- `-e errmsgfile` Specifies the name of the file to contain information concerning the records that could not be processed. The default is *STDERR*.
- `-z` If this parameter is specified, **qjloadsndomn** runs in test mode. It generates the import file, but prints the import command to the screen instead of executing the import command. It also leaves the import files in place on exit instead of deleting them. This gives you the chance to look at the import files before actually loading them.

**Table 75** Input file format

| Field                  | Value type                   | Description                                                           |
|------------------------|------------------------------|-----------------------------------------------------------------------|
| Record Type Identifier | [M] subnet                   | Specifies the record type as a subnet.                                |
| Subnet IP Address      | [M] dotted decimal (20 char) | Specifies the IP address of the subnet where the domains are located. |
| Domain Name            | [M] text (up to 253)         | The list of domains associated with the subnet (Comma separated).     |

**Important!** All data lines must end with a carriage return, or they cannot be imported.



## Exporting files with CLIs

This section describes the use of CLI modules for exporting VitalQIP objects to the VitalQIP comma separated file format (CSV). These files can be modified, if desired, and used for importing. The files are automatically exported in the VitalQIP import file formats, as described in the corresponding Import section.

## exportaddrange

---

**exportaddrange** is the CLI command for exporting managed address ranges for networks and managed object ranges within subnets into VitalQIP CSV format.

### Synopsis

```
exportaddrange [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename] -t range_type
 [-a ip_address]
```

### Parameters

**exportaddrange** recognizes the following parameters:

- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization*   Specifies the VitalQIP organization (corporation) name.
  
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*       Specifies the password for the associated administrator account.
  
- f *filename*       Specifies the directory and filename of the output data. The default is *STDOUT*.
  
- t *range\_type*     Specifies a network or subnet (object) range, as follows: **network** or **subnet**.
  
- a *ip\_address*     Specifies a subnet or network address for exporting only the ranges within a single subnet or network.

### Command line input example 1

- To export a network address range:  

```
exportaddrange -t network
```

### Output example 1

Refer to Table 55, “Address range data in VitalQIP CSV format”, on page 340 for an explanation of the field layout.

148.94.0.0,148.94.0.1,148.94.0.20

### **Command line input example 2**

- To export a subnet object range:  
exportaddrange -t subnet

### **Output example 2**

Refer to Table 55, “Address range data in VitalQIP CSV format”, on page 340 for an explanation of the field layout.

198.200.234.0,198.200.234.20,198.200.234.50



## exportcontact

---

**exportcontact** is the CLI command for exporting contact information into VitalQIP CSV format.

### Synopsis

```
exportcontact [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportcontact** recognizes the following parameters:

- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization*   Specifies the VitalQIP organization (corporation) name.
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*       Specifies the password for the associated administrator account.
- f *filename*       Specifies the directory and filename of the output data. The default is *STDOUT*.

### Output example

Refer to Table 59, "Domain and DNS Zone data file format", on page 351 for an explanation of the field layout.

```
Day,"Betty",bday@quadritek.com,44-1256-346-344,
Day,"Betty",,44-1256-346-344,44-1256-346-355
Drescher,"Alex",,610-725-8535,800-sky-Page
Klein,"Dave",,,
Reiley,"Jim",,,
Scott,"Erin",escott@world.com,610-725-8535,
Scott,"Erin",,610-725-8535,800-Sky-Page
Swiss,"Bob",,,
qipman,"qipman",,,
```



## exportdnsrr

---

**exportdnsrr** is the CLI command for exporting all DNS resource record information associated with objects, domains, and reverse zones in VitalQIP CSV format.

### Synopsis

```
exportdnsrr -t object|domain|reverse_zone [-g loginserver]
[-s servername] [-o organization] [-u username] [-p password]
[-f filename]
```

### Parameters

**exportdnsrr** recognizes the following parameters:

|                                      |                                                                                                                   |
|--------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| -t <i>object domain reverse_zone</i> | The owner type of the resource records (for example, objects, domains, or reverse zones).                         |
| -g <i>loginserver</i>                | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable. |
| -s <i>servername</i>                 | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.        |
| -o <i>organization</i>               | Specifies the VitalQIP organization (corporation) name.                                                           |
| -u <i>username</i>                   | Specifies the VitalQIP administrator account to be used in establishing the database connection.                  |
| -p <i>password</i>                   | Specifies the password for the associated administrator account.                                                  |
| -f <i>filename</i>                   | Specifies the directory and filename of the output data.                                                          |

### Output examples

Refer to Table 58, “DNS resource record file input file format”, on page 348 for an explanation of the field layout.

- Sample output for object resource record.

```
150.1.0.11,<HOST_NAME>,IN,HINFO,-1,"""QIP DDNS"""" ""BIND 8.x""",F,0,0,,
150.1.0.11,<HOST_NAME>,IN,WKS,-1,150.1.0.11 53 11 22 33 44,F
150.1.4.8,<FULL_NAME>,IN,HINFO,-1,"""QIP DDNS"""" ""BIND 4.x""",F
```

- Sample output for domain resource record:

```
usa.world.com,www.usa.world.com,IN,CNAME,-1,webserver.usa.world.com
```

- Sample output for reverse zone resource record:

```
198.200.138.0/24,138.200.198.in-addr.arpa,CH,AAAA,0,321:0:1:2:3:4:567:89AB,R,,0,0,"",""
```



## exportdomain

---

**exportdomain** is the CLI command for exporting domain and DNS zone option information into VitalQIP CSV format.

### Synopsis

```
exportdomain [-f output_filename] [-sh] -nh [-g loginserver]
 [-s servername] [-o organization] [-u username] [-p password]
```

### Parameters

**exportdomain** recognizes the following parameters:

|                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -f<br><i>output_filename</i> | Specifies the directory and file name of the output data. Each field in the output file is described in Table 59.                                                                                                                                                                                                                                                                                                                                                                                                                            |
| -sh                          | If -sh is not specified, a common header is used for all the domains in the output file, and for all the domains, only those fields shown in the domain profile window in the VitalQIP GUI will be included in the output file. If -Sh is specified, a separate header is used for each domain and each domain is separated by a space line in the output file. The zone option fields are also included in the output file. If any DNS server is associated with a domain, that DNS server information is also included in the output file. |
| -nh                          | Specifies that no header appears in the output file. This data file format is specifically required for importing domains from GUI, and it only contains those fields shown in the domain profile GUI.                                                                                                                                                                                                                                                                                                                                       |
| -g <i>loginserver</i>        | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                            |
| -s <i>servername</i>         | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| -o <i>organization</i>       | Specifies the VitalQIP organization (corporation) name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| -u <i>username</i>           | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| -p <i>password</i>           | Specifies the password for the associated administrator account.                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

## Output examples

Refer to Table 59, “Domain and DNS Zone data file format”, on page 351 for an explanation of the field layout.

```
bank.com,21600,604800,3600,86400,bank@qtek.com,0,0,"any","any","any", ,""... .
```

**Important!** The output files from **exportdomain** can be used directly as input files for the **enterdomain** CLI. Users should run **exportdomain** to get the correct input file format and modify the input file data as needed before running the **enterdomain** CLI. You can get both input file formats from **exportdomain** whether or not you use the `-sh` parameter in the command line.



## exportlocation

---

**exportlocation** is the CLI command for exporting location information into VitalQIP CSV format.

### Synopsis

```
exportlocation [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportlocation** recognizes the following parameters:

- g *loginserver*      Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*      Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization*    Specifies the VitalQIP organization (corporation) name.
  
- u *username*        Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*        Specifies the password for the associated administrator account.
  
- f *filename*        Specifies the directory and filename of the output data.

### Output example

Refer to Table 61, "Location file format", on page 358 for an explanation of the field layout.

```
,,Berne,,Switzerland
,,Basingstoke,,UK
,,New York,NY,,USA
,,Philadelphia,PA,,USA
,,Malvern,PA,,USA
```



## exportmanufacturer

---

**exportmanufacturer** is the CLI command for exporting manufacturer information into VitalQIP CSV format.

### Synopsis

```
exportmanufacturer [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportmanufacturer** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.

### Output example

Refer to Table 62, "Manufacturer data file format", on page 360 for a description of the field layout.



## exportnetwork

---

**exportnetwork** is the CLI command for exporting network information into VitalQIP CSV format.

### Synopsis

```
exportnetwork [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportnetwork** recognizes the following parameters:

- g *loginserver*      Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*      Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization*    Specifies the VitalQIP organization (corporation) name.
  
- u *username*        Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*        Specifies the password for the associated administrator account.
  
- f *filename*        Specifies the directory and filename of the output data.

### Ouput example

Refer to Table 63, "Network data file format", on page 362 for a description of the field layout.

```
148.94.0.0,16,root@dns1.world.com,Network1,N,0,0,dns1.world.com
150.1.0.0,16,root@dns1.world.com,Network2,N,0,0,dns1.world.com
150.150.0.0,16,root@dns1.world.com,Network3,N,50,2,dns4.uk.world.com
155.55.0.0,16,root@dns1.world.com,Network4,N,0,0,
198.200.234.0,24,root@dns1.world.com,Network5,N,0,0,dns1.world.com
```



## exportorganization

---

**exportorganization** is the CLI command for exporting organization information into VitalQIP CSV format. This command exports the maximum object count.

### Synopsis

```
exportorganization [-g loginserver] [-s servername] [-u username]
 [-p password] [-f filename]
```

### Parameters

**exportorganization** recognizes the following parameters:

- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- u *username*       Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*       Specifies the password for the associated administrator account.
- f *filename*       Specifies the directory and filename of the output data.

### Output example

Refer to Table 64, "Organization data file format", on page 365 for a description of the field layout.

```
1,"VitalQIP Organization","Default Organization",0
```



# exportospf

---

**exportospf** is the CLI command for exporting OSPF information into VitalQIP CSV format.

## Synopsis

```
exportospf [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

## Parameters

**exportospf** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.

## Output example

Refer to Table 65, "OSPF area data file format", on page 366 for a description of the field layout.

```
England,000.000.000.001,1,50,2,"150.150.1.0,150.150.1.64",150.150.000.000,
150.150.031.255
```



## exportreversezone

---

**exportreversezone** is the CLI command for exporting reverse zone information into VitalQIP CSV format.

### Synopsis

```
exportreversezone [-f output_filename] [-sh] [-g loginserver]
 [-s servername] [-o organization] [-u username] [-p password]
```

### Parameters

**exportreversezone** recognizes the following parameters:

- |                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -f<br><i>output_filename</i> | Specifies the directory and file name of the output data. Each field in the output file is described in Table 66.                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| -sh                          | If -sh is not specified, a common header is used for all the reverse zones in the output file, and for all the reverse zones, only those fields shown in the reverse zone profile window in the VitalQIP GUI will be included in the output file. If -sh is specified, a separate header is used for each reverse zone and each reverse zone record is separated by a space line in the output file. All of the zone option fields are included in the output file. If any DNS server is associated with the reverse zones, that DNS server information is also included in the output file. |
| -g <i>loginserver</i>        | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| -s <i>servername</i>         | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| -o <i>organization</i>       | Specifies the VitalQIP organization (corporation) name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| -u <i>username</i>           | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| -p <i>password</i>           | Specifies the password for the associated administrator account.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

□

## exportserver

---

**exportserver** is the CLI command for exporting server information into VitalQIP CSV format.

### Synopsis

```
exportserver [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]|[-c class] [-t type]
```

### Parameters

**exportserver** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.
- c *serverClass* Specifies the class of the server .
- t *serverType* Specifies the type of server from the following list: Bootp, Lucent DNS 4.x, Lucent DNS3.x, DHCP, NIS, Local Host, Windows 2000  
-t always overrides anything entered in for the -c parameter. If neither is included, all servers are returned.

### Output example

Refer to the **enterserver** CLI for a description of the field layout.

```
Type=<Server_Class><CR>
<Format><CR>
<Data_Values><CR>
```



## exportsimpleobj

---

**exportsimpleobj** is the CLI command for exporting object information in VitalQIP format into VitalQIP CSV format.

### Synopsis

```
exportsimpleobj [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportsimpleobj** recognizes the following parameters:

- g *loginserver*    Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername*    Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization*   Specifies the VitalQIP organization (corporation) name.
  
- u *username*        Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password*        Specifies the password for the associated administrator account.
  
- f *filename*        Specifies the directory and filename of the output data.

### Output Example

Refer to Table 67, "Object data in VitalQIP file format", on page 380 for a description of the field layout.

```
212.0.0.1,wsp000078wss,,qtek.com,Workstation,,,,-1,0,,0,,,1,,Workstation,yes,1
212.0.0.2,pcp000012pcs,,qtek.com,PC,,,,-1,0,,-1,,,1,,PC,1
212.0.0.3,wsp000108wss,,qtek.com,Workstation,,,,-1,0,,-1,"asdf.qtek.com.
,bdaf.qtek.com.,gfz.qtek.com.,qwerqew.qtek.com.",,,1,,Workstation,2
212.0.0.4,worksta,,qtek.com,Workstation,,,,-1,0,,-1,,,1,,Workstation,0,1
212.0.0.5,wsp000112wss,,qtek.com,Workstation,,,,-1,0,,
-1,"asdfsafs.qtek.com.,asdfsafs.qtek.com.",,,1,,Workstation,1,1
198.102.15.1,itsawonderfullife,,northpole.com,Server,,,,-1,0,,-1,,,1,,Server,1,1
198.102.15.2,santalaptop,,northpole.com,Workstation,santa's
laptop,,,,-1,0,,-1,,,1,,Workstation,1,1
198.102.15.3,santadesktop,,northpole.com,Workstation,santa's,1
desktop,,,,-1,0,,-1,"stella.q.com.",,,1,,Workstation,1
198.102.15.4,wsp000113wss,,northpole.com,Workstation,,,,-1,0,,-1,,,1,,
```

```

Workstation,1
211.111.110.1,w1,,qtek.com,Undefined,,,,,-1,0,,,-1,,,1
211.111.110.2,w2,,qtek.com,Undefined,,,,,-1,0,,,-1,,,1
211.111.110.3,w3,,qtek.com,Undefined,,,,,-1,0,,,-1,,,0
211.111.110.4,w4,,qtek.com,Undefined,,,,,-1,0,,,-1,,,1
211.111.110.5,wsp000115wss,,test-dhcp.qtek.com,qtek.com,Workstation,,,,,5,0
,,,-1,,,,"user_class1,user_class2",1,,,Workstation,1
211.111.110.6,wsp000116wss,,test-dhcp.qtek.com,qtek.com,Workstation,,,,,5,0
,,,-1,,,,"user_class1,user_class2",2
211.111.110.7,wsp000117wss,,test-dhcp.qtek.com,qtek.com,Workstation,,,,,5,0
,,,-1,,,,"user_class1,user_class2",1
211.111.110.8,wsp000118wss,,test-dhcp.qtek.com,qtek.com,Workstation,,,,,5,0
,,,-1,,,,"user_class1,user_class2",1
211.111.110.9,udp000085uds,,first.one.com,qtek.com,Undefined,,,Test,John,123
-1234,3,0,,,,-1,,,1,,,Undefined,1
211.111.110.10,whp000033whs,,qtek.com,Wiring_HUB,,,Test,John,123-1234,-1,0,,,
-1,"a1.qtek.com.,a2.qtek.com.",,,1,,,Wiring_HUB,1
211.111.110.11,router1,,qtek.com,Router,,,,,-1,0,,,,-1,,,1,,,Router,1
211.111.110.12,wsp000122wss,,qtek.com,Workstation,,,,,-1,0,,,,-1,,,1,,,Workstation,0

```



## exportsubnet

---

**exportsubnet** is the CLI command for exporting Subnet Profile information into VitalQIP CSV format.

### Synopsis

```
exportsubnet [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportsubnet** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.

### Output example

Refer to Table 68, “Subnet profile data file format”, on page 385 for a description of the field layout.

```
148.94.1.0,255.255.255.0,148.94.0.0,,,,,,,,,1,,,1,world.com
148.94.0.0,255.255.255.0,148.94.0.0,,,,,,,,,1,,,1,world.com
150.1.40.0,255.255.252.0,150.1.0.0,,,,,,,,,1,,,2,usa.world.com,uk.world.com
150.1.6.128,255.255.255.192,150.1.0.0,,,,,New York,NY,,USA,1,
 Bootp Server,,0,usa.world.com
150.1.6.64,255.255.255.192,150.1.0.0,New York,Sales,Scott,Erin,,,
 New York,NY,,USA,1,Bootp Server,,1,usa.world.com
150.1.6.0,255.255.255.192,150.1.0.0,Philadelphia,Sales,Scott,Erin,,,
 Philadelphia,PA,,USA,2,Bootp Server,,1,usa.world.com
150.1.0.0,255.255.255.0,150.1.0.0,Data Center,Network,Scott,Erin,,,Malvern,
 PA,,USA,1,Bootp Server,,2,usa.world.com,world.com
150.1.1.20,255.255.255.252,150.1.0.0,SL-New York,Network,Scott,Erin,,,
 Malvern,PA,,USA,1,Bootp Server,,1,usa.world.com,world.com
```

150.1.1.16,255.255.255.252,150.1.0.0,SL-Alps,Network,Scott,Erin,,Malvern,  
PA,,USA,1,Bootp Server,,0,usa.world.com,world.com  
150.1.1.12,255.255.255.252,150.1.0.0,SL-  
Geneva,Network,Scott,Erin,,Malvern,PA,,USA,1,Bootp  
Server,,1,usa.world.com,world.com  
150.1.1.8,255.255.255.252,150.1.0.0,SL-Winchester,Network,Scott,Erin,,  
Malvern,PA,,USA,1,Bootp Server,,1,usa.world.com,world.com



## exportsubnetorg

---

**exportsubnetorg** is the CLI command for exporting Subnet Organization information into VitalQIP CSV format.

### Synopsis

```
exportsubnetorg [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportsubnetorg** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.

### Output example

Refer to Table 69, "Subnet organization data file format", on page 387 for a description of the field layout.

```
SubTest1,"10.200.60.0","10,Server",,0,,microsoft_clients,2,85
SubOrg,"10.12.12.0,198.200.138.0,198.200.138.64,198.200.138.128,198.200.138.160,198.20
0.138.192",-1,X-
terminal,1,Terminal_Server,2,Switch,3,Legacy_System,4,Gateway,5,Test_Equipment,6,Und
efined,7,Partially_Managed",SubOrgC,0,"3ibm.seg2.qa.quadritek.com",vqaw2ka01.seg1.qa
.quadritek.com,general,3,24
SubTestMoreSubnets,"198.200.138.32",,6,Bridge",,0,"3ibm.seg2.qa.quadritek.com",,,1,40
test12,"10.58.208.0",-
5,Workstation,3,Workstation",,0,,srt08.seg5.qa.quadritek.com,general,2,25
Subnetorgtest,,-9,Server",,0,,vqaw2ka01.seg1.qa.quadritek.com,general,0,33
```



## exportudf

---

**exportudf** is the CLI command for exporting user-defined field names or values into the VitalQIP CSV format.

### Synopsis

```
exportudf -t field_name|value [-g loginserver] [-s servername]
 [-o organization] [-u username] [-p password] [-f filename]
```

### Parameters

**exportudf** recognizes the following parameters:

- t *field\_name|value* Exports the user-defined field name or value.
  
- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
  
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
  
- o *organization* Specifies the VitalQIP organization (corporation) name.
  
- u *username* Specifies the VitalQIP administrator account to be used in establishing the database connection.
  
- p *password* Specifies the password for the associated administrator account.
  
- f *filename* Specifies the directory and filename of the output data. Output is in CSV format.

### Output Example

Refer to Table 70, “UDF data file format”, on page 390 for a description of the field layout.

```
object,n,Field_1
object,n,Field_2
object,n,Field_3
object,n,Field_4
object,n,Field_5
organization,n,AUTHORITY
domain,n,AUTHORITY
revzone,n,AUTHORITY
#Export UDF Values - Example: export -t value
```

```
organization,v,AUTHORITY,150.1.0.19,James Mitchell
domain,v,AUTHORITY,150.1.0.19,William Gates
domain,v,AUTHORITY,150.1.0.19,Tom Cruise
domain,v,AUTHORITY,150.1.0.19,Pierce Brosnan
```



## exportuser

---

**exportuser** is the CLI command for exporting user information to the VitalQIP CSV format.

### Synopsis

```
exportuser [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportuser** recognizes the following parameters:

- g *loginserver* Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername* Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization* Specifies the VitalQIP organization (corporation) name.
- u *username* Specifies the VitalQIP administrator account to be used. in establishing the database connection.
- p *password* Specifies the password for the associated administrator account.
- f *filename* Specifies the directory and filename of the output data.

### Output example

Refer to Table 71, "User data file format", on page 393 for a description of the field layout. Exported files contain encrypted passwords. The record will contain one or more records similar to this:

```
johndoe,"Doe","John","Group1",,,01228529f529bd8ae71006ce2857b0a121eb,,
,,,,,1,,,"200.200.200.38",,
```



## exportusergrp

---

**exportusergrp** is the CLI command for exporting user group information into the VitalQIP CSV format.

### Synopsis

```
exportusergrp [-g loginserver] [-s servername] [-o organization]
 [-u username] [-p password] [-f filename]
```

### Parameters

**exportusergrp** recognizes the following parameters:

- g *loginserver*      Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.
- s *servername*        Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.
- o *organization*      Specifies the VitalQIP organization (corporation) name.
- u *username*            Specifies the VitalQIP administrator account to be used in establishing the database connection.
- p *password*            Specifies the password for the associated administrator account.
- f *filename*            Specifies the directory and filename of the output data.

### Output example

Refer to Table 72, “User group data file format”, on page 395 for a description of the field layout.

```
VitalQIP Admins,,,Administrators of VitalQIP
user_group_1,,,Default User Group
```



## exportzoneext

---

**exportzoneext** is the CLI command for exporting zone extension information from domains, reverse zones, and root zones. The output is in VitalQIP CSV (comma delimited) format.

### Synopsis

```
exportzoneext -t domain|reverseZone|rootZone -f input_filename
 [-g loginserver] [-s servername] [-o organization] [-u username]
 [-p password]
```

### Parameters

**exportzoneext** recognizes the following parameters:

- |                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -t <i>domain</i>  <br><i>reverseZone</i>  <br><i>rootZone</i> | Specifies the type of export source (domain, reverseZone, or rootZone). The default is <i>all</i> . <b>domain</b> exports data comes from the <b>Domain Extension</b> tab of the Domain Profile. <b>reverseZone</b> exports data comes from the <b>Reverse Zone Extension</b> tab of the Reverse Zone Profile. <b>rootZone</b> exports data comes from the <b>Root Zone Extension</b> field in the DNS Server Profile. |
| -f <i>input_filename</i>                                      | Specifies the directory and filename of the input data.                                                                                                                                                                                                                                                                                                                                                                |
| -g <i>loginserver</i>                                         | Specifies the VitalQIP login server's IP address. This value is the equivalent of the LOGIN environment variable.                                                                                                                                                                                                                                                                                                      |
| -s <i>servername</i>                                          | Specifies the VitalQIP database server. This value is the value of the QIPDATASERVER environment variable.                                                                                                                                                                                                                                                                                                             |
| -o <i>organization</i>                                        | Specifies the VitalQIP organization (corporation) name.                                                                                                                                                                                                                                                                                                                                                                |
| -u <i>username</i>                                            | Specifies the VitalQIP administrator account to be used in establishing the database connection.                                                                                                                                                                                                                                                                                                                       |
| -p <i>password</i>                                            | Specifies the password for the associated administrator account.                                                                                                                                                                                                                                                                                                                                                       |

### Output examples

Refer to Table 73, “Domain data file format”, on page 397 for a description of the field layout.

- Sample export of domain extensions:

```
world.com,us.personnel IN A 192.249.249.44, postfix
world.com, IN MX 10 us.personnel.pa.com., postfix
world.com,employeeedb.personnel IN CNAME us.personnel.pa.com., postfix
```

- **Sample export of reverse zone extensions:**

```
148.94.0.0/16,samplesystem.tstsvr IN A 192.253.252.10, postfix
148.94.0.0/16, IN MX 10 samplesystem.tstsvr.uk.com., postfix
148.94.0.0/16, IN MX 100 backupsystem.tstsvr.uk.com., postfix
148.94.0.0/16,qip.tstsvr IN CNAME samplesystem.tstsvr.uk.com., postfix
```





# Index

---

## A A record

- host name, 314
- A records, 286
- active lease file, 161
  - DHCP, 161
- active leases, 1, 8, 40
  - converting, 173
  - database, 43
  - DHCP, 1, 8
  - retrieving, 8
- address fields, 211
  - searching, 211
- address ranges, 58
  - exporting, 407
  - importing, 339
  - Network Profile, 58
- administrative roles, 60
  - adding or modifying, 219
  - deleting, 36
  - managed list report, 203
  - report, 184
  - reports, 184, 203
- administrator
  - managed ranges report, 197
- Administrator Audit Report, 182
- administrator passwords, 25
- administrators
  - changing the password, 25
  - copying and deleting, 11
  - domain folders, 84
  - domain folders policy option, 84
  - list from the database, 197
  - managed ranges, 197
- alias name, 27
- aliases, 27
- applications, 32
  - list from the database, 62
- Audit Manager, 30, 298, 306, 308

- audit table
    - clear, 28
  - audit tables
    - clearing, 28
  - Automatic Reclaim Report, 253
- 

- ## B
- billing information, 63
    - adding to an organization, 221
    - object class, 63
    - organizations, 221
    - user groups, 63
  - billing locations, 63, 221
  - BIND 4.x, 410
  - BIND 8.x, xiii, 311, 315
  - Bootp server, 23
    - list from the database, 65
  - Bootp servers
    - list from the database, 65
  - Bootptab configuration files
    - generating, 23
  - Bootptab file, 23, 332
    - generating, 23
  - Bootptab files, 23
    - exporting, 330
    - generating, 23
- 

- ## C
- clear audit data, 28
  - client class
    - vendor and user classes, 66, 223
  - configuration and data files, 45
    - generating, 47
    - SOA record, 47
  - configuration files
    - DNS, 47
  - contact information, 67
    - adding or modifying, 225, 239, 257
    - exporting, 409
    - importing, 343
  - contacts
    - contact ID, 33, 67, 259
-

- exporting, 409
- importing, 343
- contract information, 225, 259

---

**D** database, 28, 30

- DHCP server, 8, 43, 166
- DNS files, 312
- dropping tables and indexes, 298
- exporting, 312
- importing, xiii, 308, 312
- initializing, 30
- searching, 211
- utility, 298

database tables

- clearing, 28

DECNet

- addresses, 71, 294
- dual protocol, 246

DECNet address

- releasing, 294

DECNet addresses, 71, 112, 294, 297

- searching for, 212

DHCP

- active lease file, 161
- active leases, 8, 43, 161
- adding and deleting MAC addresses, 157
- dumping active lease files, 40
- dumping the active lease file, 40
- generating configuration files, 41, 166
- host name IP address synchronization, 43
- MAC pools, 157
- scope, 207
- scopes, 207
- subnet list, 74
- templates, 139, 141, 223, 289
- upgrading from 3.1 to 4.0, 173

DHCP Policy Template, 138, 223, 246, 261

DHCP scopes, 166, 207

DHCP server subnets, 74

DHCP servers, 173

- database, 8, 43, 166
- exporting, 420
- importing, 312, 338, 350, 376
- list from the database, 75
- MAC address pools, 157
- MAC pools, 157
- Microsoft, 166
- reports, 188

DHCP templates

- client class, 66, 223
- creating, 289

DNS

- configuration and data files, 47
- domain extensions, 430
- exporting files, 312
- exporting resource records, 410
- exporting zone options, 412
- generating configuration and data files, 45
- importing, 312, 329
- importing resource records, 327, 347
- importing zone options, 350
- non-managed servers, 104, 236
- resource records, 318, 327, 347
- reverse zones, 123
- Update Service, 51
- updating the SOA, 47
- zone extensions, 151, 271
- zone options, 266, 350
- zones, 266

DNS files

- exporting, 312
- importing, 312

DNS servers, 80

- exporting, 420
- importing, 312, 324, 329, 338, 350, 376
- list from the database, 80
- non-managed, 236
- non-managed profile, 104
- primary, 123, 276

- resource records, 286
- secondary, 128, 276
- DNS Update Service, 51
- domain
  - adding, modifying, or deleting, 49
  - domain extensions, 286
  - folders, 49
  - grouping into folders, 49
- Domain Controller, 277
- domain extensions, 430
- domain folders, 49, 84
- Domain Profile, 154, 274, 430
- domains, 82
  - domain folders, 49, 84, 230
  - Domain Profile, 154, 274, 430
  - exporting, xiii, 410, 412
  - folders, 84, 230
  - importing, 312, 324, 338, 404
  - resource records, 213, 227, 321
  - user-defined fields, 213
  - zone extensions, 271

---

**E**

- encryption, 181
- enteraddrange, 339
- enteralias, 312, 328, 329, 341
- entercontact, 343
- enterdnsobj, 345
- enterdnsrr, 312, 327, 329, 347
- enterdomain, 312, 324, 329, 338, 350
- enterlocalobj, 356
- enterlocation, 358
- entermanufacturer, 360
- enternetwork, 312, 336
- enterorganization, 364
- enterospf, 366
- enterreversezone, 368
- enterserver, 312, 338, 350
- entersimpleobj, 312
- entersubnet, 312
- entersubnetorg “
  - “D2HBentersubneto49, 387

- enterudf, 390
- enteruser, 392
- enterusergrp, 395
- enterzoneext, 397
- exportaddrange, 407
- exportcontact, 409
- exportdnsrr, 410
- exportdomain, 412
- exporting
  - Audit Manager data, 306
  - Bootptab files, 330
  - contacts, 409
  - DNS files, 312
  - DNS zone options, 412
  - domains, xiii, 410, 412
  - locations, 414
  - managed address ranges, 407
  - manufacturers, 415
  - networks, 416
  - objects, 330, 421
  - organizations, 417, 425
  - OSPF areas, 418
  - OSPFs, 418
  - resource records, 410
  - reverse zones, 419
  - servers, 420
  - subnet organizations, 425
  - Subnet Profile, 423
  - subnets, 423
  - user groups, 429
  - user-defined fields, 426
  - users, 428
  - VitalQIP data, 306
  - zone extensions, 430
- exporting Bootptab files, 330
- exporting contacts, 409
- exportlocation, 414
- exportmanufacturer, 415
- exportnetwork, 416
- exportorganization, 417
- exportospf, 418

- exportreversezone, 419
- exportserver, 420
- exportsimpleobj, 421
- exportsimpleobj, 421
- exportsubnet, 423
- exportsubnetorg, 425
- exportudf, 426
- exportuser, 428
- exportusergrp, 429
- exportzoneext, 430

---

**F** folders, 49, 84  
free subnets, 86, 192  
report, 192

---

**G** GAP objects, 132, 256  
Generating Bootptab configuration files, 23  
generating Bootptab files, 23  
Global Policies, 37  
deleting, 37  
objects, 357  
user-defined fields, 37

---

**H** HINFO records, 213, 314  
importing, 345  
host name, 27, 43  
A record, 314  
host names  
synchronization with IP addresses, 43  
hub ports, 89, 297  
releasing, 295

---

**I** importing, 311  
Bootptab files, 336  
contact information, 343  
database information, 308  
DNS files, 312, 329  
DNS servers, 312, 324, 329, 338, 376  
domains, 312, 324, 329, 338, 350, 404

- locations, 358
- MAC addresses, 338
- managed address ranges, 339
- manufacturers, 360
- networks, 312, 336
- object alias information, 312, 328, 329, 341
- object information in local format, 356
- objects, 312, 318, 333, 341, 399
- organizations, 279, 364, 387
- OSPF areas, 366
- OSPFs, 338, 366
- resource records for objects, 345
- resource records, 312, 318, 327, 347
- resource records for objects, 312, 327, 329, 347
- reverse zones, 368
- rules file, 43
- ser-defined fields, 390
- servers, 312, 338, 350
- subnet domains, 404
- subnet organizations, 387
- Subnet Profile, 384
- subnets, 312
- user groups, 395
- users, 392
- using the rules file, 318
- zone extensions, 397
- zone options, 350

importing contacts, 343  
importing DHCP servers, 376  
importing servers, 324, 375, 420  
Inquiry Report, 194  
IP addresses, 8, 296  
aliases, 328  
and host names, 27  
deleting objects, 32  
DHCP scope, 166, 207  
free, 92, 296  
Planned Use addresses, 2  
reclaiming, 174  
reserved, 26, 32

- retrieving available, 296
    - subnet addresses, 92, 129
    - subnets, 92, 129
    - synchronization with host names, 43
    - unallocated, 32, 296
    - unallocating, 32
    - unlocking, 297
    - unused, 26, 174
    - users, 150
  - IP addresses and aliases, 27
  - IP addresses and host names, 27
- 
- L**
    - Local Host configuration file
      - generating, 160
    - Local Host Server
      - configuration files, 160
    - Local Host server, 160
    - Local host server, 93
    - local host servers
      - list from the database, 93
    - local objects
      - importing, 356
    - locations
      - adding and modifying, 232, 239, 257
      - adding or modifying, 232
      - billing, 221
      - exporting, 414
      - importing, 358
      - list from the database, 94
      - location ID, 264
- 
- M**
    - MAC address, 338
      - adding or modifying, 157
      - querying the MAC pool, 157
    - MAC address pools, 157
      - importing, 338
    - MAC addresses
      - adding and deleting from MAC pools, 157
    - MAC pools, 284
- 
- managed ranges
    - report, 197
  - manufacturers
    - exporting, 415
    - importing, 360
    - list from the database, 96
    - model lists, 98
  - Microsoft DHCP, 166
- 
- N**
    - network address ranges, 58
    - networks
      - exporting, 416
      - free subnets, 192
      - importing, 312
      - importing managed address ranges, 339
      - list from the database, 102
      - shared networks, 137
    - NIS
      - generating files, 159
    - NIS servers, 103, 159
      - generating files, 159
      - importing, 376
      - list by organization, 103
      - list from the database, 103
    - non-managed DNS servers, 236
- 
- O**
    - object
      - object class, 115
    - object aliases
      - importing, 312, 328, 341
    - object class
      - billing, 63
    - Object History Report, 185
    - Object Profile, 108, 207, 239
    - objects
      - adding and modifying, 239
      - adding or modifying, 239, 330
      - aliases, 27
      - billing, 63, 221
      - by manufacturer, 96
-

- by user, 150
- deleting, 32, 293
- exporting, 330, 421
- free, 115
- GAP objects, 132, 256
- host name, 27, 43
- importing, 312, 318, 333
- importing local objects, 356
- list from the database, 106
- moves, 162, 164
- moving, 162
- object classes, 63, 96, 98, 132, 256, 318
- object names, 115, 166
- Object Profile, 108, 207
- object range, 32
- policies, 357
- reclaiming, 174
- report, 194, 199
- reports, 186, 194, 199
- resource records, 77, 227, 287, 318
- scheduled moves, 134
- searching, 211
- statistics, 174
- status, 26, 43
- subnet address, 129
- tombstoned, 78
- unused, 26
- user-defined fields, 262

Oracle, 30, 33, 298, 306, 338, 404

organizations, 3, 5, 65

- billing information, 221
- Bootp servers by organization, 65
- domain folders, 84
- exporting, 417, 425
- generating DNS files, 51
- importing, 279, 364, 387
- list by user name, 117
- NIS servers by organization, 103
- reverse zones, 125
- subnet organizations, 33, 87, 130, 255, 279, 387, 425

- user-defined fields, 145, 262

OSPF, 119, 248

- exporting, 418
- importing, 338, 366
- list from the database, 118, 119
- objects by OSPF, 199
- OSPF Profile, 119, 248

OSPF areas, 418

OSPF Profile, 119, 248

---

**P** passwords, 3, 5

- administrators, 25
- changing, 25
- encrypting, 29
- encryption, 29
- qip.pcy file, 3
- qipman, 29

Planned Use addresses, 2, 297

policies

- adding or modifying, 250
- deleting, 37
- deleting records, 293

policies file, 3, 5, 308

policy, 121

- adding or modifying, 250
- administrators, 84
- billing, 382, 386
- deleting, 33, 37
- hostname, 357, 380
- qip.pcy file, 5
- reclaim policy, 175
- tombstone, 293

policy information, 121

- adding or modifying, 250

---

**Q** qip.pcy, 3, 5, 308

qip.pcy file

- passwords, 3

qip-active, 8

qip-admin, 11

qip-altersubnet, 21  
qip-bootpgen, 23  
qip-bootptabcsv, 331  
qipbulkload, 312, 341, 399  
qip-changepassword, 25  
qip-check, 26  
qip-checkobjname, 27  
qip-clear, 28  
qip-crypt, 29  
qip-dbinit, 30  
qip-del, 32  
qip-delacldtemplate, 35  
qip-deladminrole, 36  
qip-delpolicy, 37  
qip-dhcpdebuglog, 38  
qip-dhcpdump, 40  
qip-dhcpgen, 41  
qip-dhcpsync, 43  
qip-dnscsv, 312  
qip-dnsngen, 45  
qip-dnsupdate, 47  
qip-export, 306  
qip-folder, 49  
qip-genddnsconfs, 51  
qip-getacldtemplate, 52  
qip-getactiveobjectlst, 54  
qip-getaddrangelst, 58  
qip-getadminrole, 60  
qip-getapplst, 62  
qip-getbillinfo, 63  
qip-getbtpsvrlst, 65  
qip-getclientclass, 66  
qip-getcontactlst, 33, 67, 225, 259  
qip-getdebuglevel, 69  
qip-getdecnetaddr, 71, 294  
qip-getdhcpscopes, 72  
qip-getdhcpsubnetlst, 74  
qip-getdhcpsvrlst, 75  
qip-getdnsrr, 77  
qip-getdnssvrlst, 80  
qip-getdomnlst, 82  
qip-getfolderlst, 84  
qip-getfreesubnetlst, 86  
qip-gethublst, 88  
qip-gethubport, 89  
qip-gethubslotlst, 91  
qip-getipaddr, 92  
qip-getlocalsvrlst, 93  
qip-getloclst, 94, 259  
qip-getmaclst, 96  
qip-getmacmodellst, 98  
qip-getmacpools, 100  
qip-getnetlst, 102  
qip-getmissvrlst, 103  
qip-getnmdnssserver, 104  
qip-getobjectlst, 106  
qip-getobjectprof, 108  
qip-getobjname, 115  
qip-getorganization, 117  
qip-getospflst, 118  
qip-getospfprof, 119  
qip-getpolicy, 121  
qip-getprimdnssvrlst, 123  
qip-getrevzonelst, 125  
qip-gettrrlst, 127  
qip-getsecdnssvrlst, 128  
qip-getsnaaddr, 129  
qip-getsnorglst, 130  
qip-getsnorgprof, 131  
qip-getsubnetlst, 133  
qip-getsubnetprof, 135  
qip-gettemplate, 139  
qip-gettemplst, 141  
qip-gettimesvrlst, 143  
qip-getudfnelst, 147  
qip-getuser, 148  
qip-getuseraddrlst, 150  
qip-getzoneext, 151  
qip-getzoneprof, 154  
qip-globalmacpool, 157  
qip-hndbgen, 159  
qip-hostgen, 160

- qip-import, 308
- qip-leasefilegen, 161
- qiploadsndomn, 404
- qip-mcancel, 162
- qip-move, 164
- qip-msextract, 166
- qip-namingpolicy, 167
- qip-objectclass, 170
- qip-qdhcplease, 173
- qip-reclaim, 174
- qip-report, 177
- qip-rot13, 181
- qip-rptadminaudit, 182
- qip-rptadminrole, 184
- qip-rptaudithistory, 186
- qip-rptdhcp, 188
- qip-rptfreesubnet, 192
- qip-rptinquire, 194
- qip-rptmanaged, 197
- qip-rptobjectlst, 199
- qip-rptrole, 203
- qip-rptzonerr, 205
- qip-scope, 207
- qip-search, 211
- qip-searchacltemplates, 216
- qip-setadminrole, 219
- qip-setbillinfo, 221
- qip-setclientclass, 66, 223
- qip-setcontact, 225
- qip-setdnsrr, 78, 227
- qip-setdomainfolder, 230
- qip-setlocation, 232
- qip-setmacpools, 234
- qip-setnmdnsserver, 236
- qip-setobject, 239, 330
- qip-setospfprof, 248
- qip-setpolicy, 250
- qip-setreclaimschedule, 252
- qip-setsnorgprof, 255
- qip-setsubnet, 257
- qip-setudf, 262

- qip-setuser, 264
- qip-setzonednsoptions, 266
- qip-setzoneext, 271
- qip-setzoneprof, 274
- qip-sitegen, 277
- qip-siteimport, 279
- qip-splitmergeenum, 281
- qip-splitrevzone, 283
- qip-subnetmacpool, 284
- qip-syncexternal, 286
- qip-template, 289
- qip-tombstonepurge, 293
- qip-ungetdecnetaddr, 294
- qip-ungethubport, 295
- qip-ungetipaddr, 296
- qip-unlock, 297
- qip-util, 298
- qsi-import, 399, 404

---

**R** reclaim

- IP addresses, 174

reource records

- domains, 213, 321
- importing, 312, 347

reousrce records

- reports, 205

report

- object list, 199

Reports

- administrator audit, 182
- Automatic Reclaim, 253

reports, 177

- Adminisitrative Role, 184
- administrative roles, 184, 203
- administrative roles and managed lists, 203
- Administrator Audit, 182
- administrator audit, 182
- administrators' managed lists, 197, 203
- DHCP Server Profile, 188
- DHCP server profile, 188

- free subnet, 192
  - free subnets, 192
  - Inquiry Report, 194
  - managed range, 197
  - Object History, 186
  - object history, 185, 186
  - object inquiry, 194
  - object list by owner, 199
  - objects, 194, 199
  - qip-report, 8, 177
  - resource records, 205
  - scheduled reclaims, 174
  - zone resource records, 205
  - reserved addresses, 26
  - resource record
    - CNAME, 113, 205, 246, 314, 318, 345, 348
    - SRV, 213
  - resource records, 77, 194, 205, 212, 213, 286, 312, 347, 410
    - adding and deleting, 227
    - deleting, 293
    - DNS, 318, 327, 347
    - domain, 77
    - exporting, 410
    - HINFO, 213, 314, 345
    - importing, 312, 327, 329, 345, 347
    - objects, 77
    - reports, 205
    - reverse zones, 77
    - rules file, 43
    - searching, 194, 212, 213
  - reverse zone
    - resource records, 227
    - zone extension, 271
  - Reverse Zone Profile, 154, 274
  - reverse zones, 123
    - DNS resource records, 77, 227
    - DNS servers, 123
    - exporting, 419
    - importing, 368
    - list from the database, 125
    - profile, 154, 274
    - PTR record, 314
    - Reverse Zone Profile, 154, 274
    - splitting, 283
    - user-defined fields, 33, 146, 263, 269
    - zone extensions, 151, 267, 271
  - routers, 42, 113, 127
    - default routers, 113, 138, 245, 260
  - rules file, 318
- 
- S**
    - scheduled moves, 134
      - canceling, 162
      - cancelling, 162
    - scheduled reclaims, 174
    - searching
      - for contacts, 67
      - for locations, 94
      - for objects, 194
      - qip-search, 211
    - searching VitalQIP, 211
    - secondary servers
      - refresh time, 268
    - serarching
      - searching the search\_token list, 322
    - serial number
      - in SOA record, 47
      - of object, 111, 244
    - servers
      - DHCP, 173
      - DNS servers, 123, 324
      - exporting, 420
      - importing, 312, 324, 338, 350, 375, 420
      - local host servers, 93
      - NIS, 103, 159
      - non-managed DNS, 104
      - Oracle database servers, 30
      - primary DNS servers, 123, 128
      - time servers, 143
      - upgrading, 173
    - site files, 279
    - sites

- creating, 277
- subnet organizations, 277
- SOA record, 47, 324, 326
- SOA records
  - updating, 47
- subnet
  - subnet addresses, 21, 44, 75, 129, 200, 257
- Subnet Organization Profile, 131
- subnet organizations
  - deleting, 33
  - exporting, 425
  - importing, 387
  - list from the database, 130
  - profile, 131, 255
  - profiles, 131, 255
  - Windows 2000, 277, 279
  - Windows 2000 sites, 279
- Subnet Profile, 135, 257
  - exporting, 423
  - importing, 384
- subnet profile
  - user-defined fields, 145, 263
- subnets
  - adding or modifying, 257
  - associated domains, 82
  - DHCP servers, 74
  - exporting, 423
  - free subnets, 86, 192
  - free subnets report, 192
  - importing, 312
  - importing domains, 404
  - join, 21
  - MAC pools, 284
  - moves, 162, 164
  - organization profiles, 255
  - profile, 135
  - profiles, 257
  - reclaiming, 174
  - router list, 127
  - routers, 42, 127

- scheduled moves, 134
- split, 21
- splitting and joining, 21
- Subnet Profile, 135, 257
- Windows 2000, 279
- wiring hubs, 88
- Sybase, 30, 298, 306, 308, 338, 400, 404
- synchronizing IP addresses with host names, 43

---

**T** templates

- DHCP Policy Templates, 138, 223, 246, 261

- time servers, 113, 138, 143, 245, 260
  - list from the database, 143
- tombstoned objects, 78

---

**U** user classes, 66, 223, 263

- user groups, 63
  - billing information, 63
  - billing user groups, 63, 113, 382, 386, 393
  - exporting, 429
  - importing, 395
- User Profile, 262, 263
- user-defined fields, 194, 269
  - class, 32, 262
  - deleting, 33
  - domains, 213
  - exporting, 426
  - importing, 390
  - names, 144, 147
  - policies, 37
  - policy, 32
  - reverse zones, 33
  - setting the value, 262
  - Subnet Profile, 145, 263
  - User Profile, 263
  - values, 144
- users, 66, 148, 223, 263

- billing user groups, 63, 138, 221, 246, 261
- deleting, 33
- exporting, 428
- importing, 392
- IP addresses, 150
- searching for, 195
- User Profile, 264

---

**V** vercheck, 303  
VitalQIP DHCP Report, 189

---

**W** Windows 2000 sites

- creating, 277

wiring hub slots, 91wiring hubs

- slot list, 91

Wiring\_HUB, 91, 96, 98, 107, 110, 112, 115, 200, 243, 245, 322, 380, 388, 401Wiring\_HUB slots, 91, 245

---

**Z** zone extension

- DNS server, 271

zone extensions, 151

- DNS servers, 151
- exporting, 430
- importing, 397
- setting, 271

zone options

- DNS, 266, 350, 412
- exporting, 412
- importing, 350
- Windows 2000, 269

zone resource records, 205zones

- setting the profiles, 274
- setting zone options, 266

---

